

Министерство образования Республики Беларусь  
Белорусский национальный технический университет  
Автотракторный факультет  
Кафедра «Экономика и логистика»

**ЭЛЕКТРОННЫЙ УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ПО  
УЧЕБНОЙ ДИСЦИПЛИНЕ**

**«СЕТЕВЫЕ ТЕХНОЛОГИИ И БАЗЫ ДАННЫХ»**

для специальности

1-27 01 01 «Экономика и организация производства (по направлениям)»,

направления специальности

1-27 01 01-02 «Экономика и организация производства (автомобильный транспорт)»

Составитель: Стефанович Н.В.

Минск ◇ БНТУ ◇ 2022

## Перечень материалов

### Теоретический раздел

- тематический конспект лекций

### Практический раздел

- материалы лабораторных занятий

### Контроль знаний

- экзаменационные вопросы;
- задание для выполнения курсовой работы

### Вспомогательный раздел

- выдержки из учебной программы;
- рекомендуемая литература

## Пояснительная записка

Цели данного ЭУМК – повышение эффективности организации учебного процесса с использованием дистанционных технологий; представление возможности студентам заниматься самообразованием, пользуясь комплектом учебно-методических материалов по курсу «Сетевые технологии и базы данных».

ЭУМК содержит четыре раздела: теоретический, практический, контроля знаний и вспомогательный раздел. В теоретическом разделе представлен лекционный материал в соответствии с основными разделами и темами учебной программы. Практический раздел включает лабораторный практикум. Раздел контроля знаний включает экзаменационные вопросы, а также задания для выполнения курсовой работы, предусмотренной программой дисциплины. Вспомогательный раздел содержит выдержки из учебной программы дисциплины, список рекомендуемой литературы.

Материалы учебно-методического комплекса представлены в формате *PDF*. Учебные материалы структурированы по разделам. ЭУМК содержит гиперссылки, позволяющие оперативно найти необходимый материал, перейти к нужной теме или дополнительному источнику информации. Предусматривается навигация по разделам через закладки *PDF* формата, обеспечивающая возможность быстрого поиска требуемой информации и быстрый возврат к предыдущей информации.

Открытие ЭУМК производится посредством запуска файла *STiBD.pdf*.

# ТЕОРЕТИЧЕСКИЙ РАЗДЕЛ

## Содержание

### Раздел I. БАЗЫ ДАННЫХ

[Тема 1.1. Основные концепции организации данных и реляционная модель данных](#)

[Тема 1.2. Проектирование реляционных баз данных](#)

[Тема 1.3. Разработка приложений для работы с базой данных](#)

[Тема 1.4. Введение в язык SQL](#)

### Раздел II. СЕТЕВЫЕ ТЕХНОЛОГИИ

[Тема 2.1. Введение в компьютерные сети. Аппаратное и программное обеспечение компьютерных сетей. Протоколы компьютерных сетей](#)

[Тема 2.2. Глобальные компьютерные сети](#)

[Тема 2.3. Основы бизнеса в Internet](#)

[Тема 2.4. Вопросы безопасности информации в Internet](#)

[Тема 2.5. Общие сведения о проектировании web-узлов. Язык HTML](#)

[Тема 2.6. Использование программ, выполняющихся в контексте Web-браузера, при разработке Web-документов. Динамическая генерация содержимого Web-документа](#)

## РАЗДЕЛ I. БАЗЫ ДАННЫХ

### *Тема 1.1 Основные концепции организации данных и реляционная модель данных*

[Понятие и классификация СУБД. Иерархические, сетевые, реляционные базы данных](#)

Впервые термин «реляционная модель данных» ввел сотрудник фирмы IBM доктор Кодд (математик по образованию), который предложил использовать для обработки данных аппарат теории множеств (объединение, пересечение, разность, декартово произведение) и показал, что любое представление данных сводится к совокупности двумерных таблиц особого вида, известного в математике как отношение (relation, англ.).

Реляционной является БД, в которой все данные организованы в виде набора двумерных таблиц и все операции над данными сводятся к операциям над этими таблицами.

*Система управления базами данных (СУБД) с централизованной архитектурой*

При использовании этой технологии база данных, СУБД и прикладная программа (приложение) располагаются на одном компьютере. Для такого

способа организации не требуется поддержки сети и все сводится к автономной работе.

#### *СУБД (архитектура «файл-сервер»)*

База данных в виде набора файлов находится на жестком диске специально выделенного компьютера (файлового сервера). При этом существует локальная сеть, состоящая из клиентских компьютеров, на каждом из которых установлены СУБД и приложение для работы с БД.

СУБД инициирует обращения к данным, находящимся на файловом сервере, в результате которых часть файлов БД копируется на клиентский компьютер и обрабатывается. При необходимости (в случае изменения данных) данные отправляются назад на файловый сервер с целью обновления БД. Однако недостаточно развитый аппарат транзакций служит потенциальным источником ошибок в плане нарушения смысловой и ссылочной целостности информации при одновременном внесении изменений в одну и ту же запись.

#### *СУБД с архитектурой «клиент-сервер»*

Использование технологии «клиент – сервер» предполагает наличие некоторого количества компьютеров, объединенных в сеть, один из которых выполняет особые управляющие функции (является сервером сети) и разделяет функции приложения пользователя (называемого клиентом). При этом по сети "путешествуют" только те данные, которые необходимы клиенту, за счет чего снижается нагрузка на сеть и существенно уменьшается сетевой трафик.

Трехзвенная (*клиент - сервер приложений - сервер базы данных*) представляет собой дальнейшее совершенствование технологии «клиент – сервер», в которой приложения разделены на три логических и физических уровня: уровень представления (пользовательский интерфейс), уровень приложения, на котором осуществляется обработка данных, и уровень данных, предназначенный для хранения и управления данными, относящимися к приложению.

По функциональному назначению СУБД делят на:

- системы оперативной обработки транзакций (OLTP-системы, Online Transaction Processing);
- системы делового анализа / хранилища данных (OLAP-системы, Online Analysis Processing).

Если первые это основа функционирования предприятия: принятие заказов клиентов, учет сырья, складской учет, учет оплаты продукции, или ведение в электронном виде небольших по объему транзакций (операций с денежными средствами: перевод, вывод или зачисление на счет). Транзакция - это логически завершенная банковская операция, заключающаяся в переводе определенной суммы денег со счета одного лица на счет другого.

СУБД делового анализа используются для принятия решений на основе сбора и анализа большого объема информации.

Распределенная система управления базой данных (распределенная СУБД) состоит из единой логической базы данных, разделенной на некоторое

количество фрагментов. Каждый фрагмент базы данных сохраняется на одном или нескольких компьютерах, работающих под управлением отдельных СУБД и соединенных между собой сетью связи.

Приведем список современных СУБД (рис. 1.1).



Рис. 1.1. Современные СУБД

### *Тема 1.2 Проектирование реляционных баз данных*

Организация типичной СУБД и состав ее компонентов соответствует набору функций. Логически в современной СУБД можно выделить внутреннюю часть — ядро СУБД (Data Base Engine), компилятор языка баз

данных (обычно SQL - Structured Query Language), подсистему поддержки времени выполнения, набор утилит.

Ядро СУБД отвечает за журнализацию и управление:

- данными во внешней памяти;
- буферами оперативной памяти;
- транзакциями.

Все функции взаимосвязаны, поэтому компоненты должны взаимодействовать по протоколам или определенным правилам.

Ядро СУБД является основной резидентной частью СУБД и обладает собственным интерфейсом, не доступным пользователю напрямую и используемым в программах, производимых компилятором SQL, и в утилитах БД, реализующих такие процедуры, которые накладно выполнять с использованием языка БД, например, загрузка БД, сбор статистики, глобальная проверка целостности.

Почти все продукты баз данных, созданные с 1970 года, основаны на подходе, который называют реляционным.

Основными понятиями реляционных баз данных являются тип данных, домен, атрибут, кортеж, первичный ключ и отношение.

В таблице 1.1 приведены термины, используемые на практике, и показан общий смысл этих понятий.

Таблица 1.1 Терминология БД

Теория БД	Практика	SQL Server
Отношение (Relation)	Таблица (Table)	Таблица (Table)
Кортеж (Tuple)	Запись (Record)	Строка (Row)
Атрибут (Attribute)	Поле (Field)	Столбец (Column)
Домен (Domain)	Общая совокупность допустимых значений	Количество столбцов
Степень отношения	Кардинальное число отношения	Количество строк

Понятие тип данных в реляционной модели данных полностью адекватно понятию типа данных в языках программирования. Обычно в современных реляционных БД допускается хранение символьных, числовых данных, битовых строк, специализированных числовых данных (таких как "деньги"), а также специальных "темпоральных" данных (дата, время, временной интервал).

При работе с таблицами в реляционных базах данных, желательно (необходимо), чтобы каждая таблица имела так называемый первичный ключ.

Первичный ключ – это поле, которое используется для обеспечения уникальности данных в таблице. Уникальность необходима во избежание неоднозначности, когда неизвестно к какой записи таблицы можно обратиться, если в таблице есть повторяющиеся записи (две записи имеют одинаковые значения во всех полях таблицы).

[Базисные средства манипулирования реляционными данными](#)  
[Проектирование реляционных баз данных с использованием нормализации.](#)  
[Внутренняя организация реляционных СУБД](#) (стр. 12-23).

Среди важных характеристик любой базы данных следует назвать производительность, надежность и простоту администрирования. Знание того, как большинство СУБД физически хранят данные во внешней памяти, представление о параметрах этого хранения и соответствующих методах доступа может помочь при проектировании баз данных, обладающих заданной производительностью.

Любая СУБД основывается на конкретном комплексном решении задач, связанных с организацией хранения и управления данными.

[Физическая организация баз данных](#) (стр. 116-131).

### ***Тема 1.3 Разработка приложений для работы с базой данных***

Пользователи и система по-разному воспринимают работу базы данных и ее предназначение. Пользователь рассматривает ее в качестве таблицы, в которой содержится необходимая ему информация для выполнения тех или иных задач. Как системный компонент база данных представляет собой комплекс файлов, внутри которых расположены всевозможные связи и таблицы, к которым должен получить доступ пользователь.

На сегодня большинство компаний использует централизованную технологию обеспечения доступа к базам данных, которая предусматривает прямое взаимодействие сервера и клиента. В качестве последних выступают компьютеры пользователей, которых обслуживают базы данных, в то время как сервер представляет собой высокомогущный компьютер, с помощью которого пользователи могут одновременно получить доступ к одной и той же информации, даже если одинаковые запросы будут отправлены с более чем тысячи компьютеров.

Пользователь или приложение создает запрос SQL-серверу, который выполняет его обработку и возврат вместе с запрошенными данными. Реализация этого запроса осуществляется на специальном языке, который одинаково понятен обеим сторонам.

Сервера могут отличаться друг от друга по следующим критериям:

- технология предоставления доступа к данным;
- каким образом осуществляется хранение информации;
- каким образом пользователь получает сведения с сервера при отправке соответствующего запроса.

Каждая из задач предусматривает использование определенного программного компонента:

- прикладные программы, с помощью которых пользователи могут получить доступ и визуализацию базы данных;

- интерфейс, через который предоставляется вся необходимая информация;
- специальное программное обеспечение для взаимодействия с файловой системой, с помощью которого обеспечивается надежное хранение данных на сервере.

Система управления базой данных представляет собой своеобразную программную прослойку, которая создается между взаимодействующими сторонами (клиентом и сервером), чтобы пользователи могли абстрагироваться от системного видения базы данных, но при этом не потеряли возможности нормально с ней взаимодействовать.

Грамотно построенная система должна обеспечивать:

- возможность сохранения всей необходимых данных на носители и ее извлечения по необходимости;
- возможность взаимодействия с данными, расположенными в серверной оперативной памяти;
- запись истории внесения корректировок в базу данных;
- поддержание запросов пользователей.

В зависимости от того, как обеспечена реализация указанных компонентов, системы управления базами данных разделяются на несколько типов. Помимо этого, они различаются по тому, каким образом внутри них обеспечивается связь данных.

База данных (БД) представляет собой совокупность взаимосвязанных, хранящихся вместе данных при наличии такой минимальной избыточности, которая допускает их эффективное использование для одного или нескольких приложений (задач). Данные и их описания (словарь данных) хранятся так, что они в значительной степени становятся независимыми от использующих их программ. Для добавления новых или модификации существующих данных, а также для поиска данных в базе данных применяется общий управляемый способ.

Современные системы идентификации и аутентификации пользователей позволяют с высокой степенью вероятности определить подлинность пользователя, либо удаленного узла. Данные системы предназначены для однозначного определения субъекта доступа и его полномочий по отношению к конкретному ресурсу.

**Идентификация** — это процедура распознавания субъекта по его идентификатору. В процессе регистрации субъект предъявляет системе свой идентификатор, и она проверяет его наличие в своей *базе данных*. Субъекты с идентификаторами известными системе считаются легальными (законными), остальные субъекты относятся к нелегальным.

**Аутентификация пользователей** — процедура проверки подлинности субъекта, позволяющая достоверно убедиться в том, что субъект, предъявивший свой идентификатор, на самом деле является именно тем субъектом, идентификатор которого он использует. Для этого он должен



подтвердить факт обладания некоторой информацией, которая может быть доступна только ему одному (пароль, ключ).

**Авторизация** — процедура предоставления субъекту определенных прав доступа к ресурсам системы после прохождения им процедуры аутентификации. Для каждого субъекта в системе определяется набор прав, которые он может использовать при обращении к ее ресурсам.

Существующие системы аутентификации пользователей:

- 1) **парольные системы** (самый простой и распространенный способ);
- 2) системы РКІ (криптографические сертификаты);
- 3) системы одноразовых паролей;
- 4) биометрические системы.

1) В основе механизмов аутентификации пользователей «лежат» ПАРОЛИ, поэтому данный способ наиболее распространенный. В силу своей «открытости», а также ужесточении требований к длине пароля, большим количеством программного обеспечения для взлома паролей, данная система является наиболее уязвимой.

Основными «проблемными» местами систем паролей являются:

- «сложность» для запоминания пароля конечным пользователем, как следствие — нарушение Политики безопасности организации;
- хранение паролей в «открытом» виде; «беззащитность» пароля при вводе;
- применение «нестойких» алгоритмов аутентификации и открытых каналов передачи данных;
- легкость взлома с помощью специального программного обеспечения — «взломщиков паролей», «клавиатурных шпионов», «кейлогеров»;
- канал передачи пароля — открытый, нешифрованный канал передачи данных.

2) Технология инфраструктуры открытых ключей позволяет проверять и удостоверять подлинность пользователя. Инфраструктура открытых ключей или РКІ обеспечивает единую идентификацию, аутентификацию и авторизацию пользователей системы, приложений и процессов и вместе с этим гарантирует доступность, целостность и конфиденциальность информации. Инфраструктура РКІ представляет собой систему цифровых сертификатов, носителями которых являются USB-ключи или смарт-карты.

При использовании индивидуального секретного пароля и средств криптографической защиты, цифровые сертификаты получают роль электронных паспортов. Использование в корпоративной сети технологии инфраструктуры открытых ключей значительно повышает безопасность всей сети в целом, так как позволяет отказаться от использования парольной аутентификации пользователей внутри, и обеспечивает безопасный доступ удаленных пользователей в систему.

Основные носители информации: USB-ключи; Смарт-карты.

Использование в корпоративной сети технологии РКІ значительно повышает безопасность всей сети в целом, так как позволяет отказаться от

использования парольной аутентификации пользователей внутри, а также обеспечивает безопасный доступ удаленных пользователей в систему. Пользователям не надо запоминать сложные пароли и периодически их менять — достаточно подключить электронный ключ или смарт-карту и ввести PIN-код.

3) Системы многофакторной аутентификации, основанные на технологии одноразовых паролей ОТР является платформенно-независимым решением для аутентификации мобильных пользователей, которое отличается крайней простотой в использовании, установке и администрировании.

Данная технология основана на том, что пароль пользователя не постоянен и изменяется с течением времени специальным устройством (аппаратным или программным) — токеном. Данное решение широко используется в системах удаленного доступа, в том числе системах клиент-банк, для аутентификации пользователей при доступе из недоверенной среды (Интернет-кафе, бизнес-центр).

ОТР-токен — мобильное персональное устройство, принадлежащее определенному пользователю, генерирующее одноразовые пароли, используемые для аутентификации данного пользователя. ОТР-токены имеют небольшой размер и выпускаются в виде: карманного калькулятора; брелока; смарт-карты; устройства, комбинированного с USB-ключом; специального программного обеспечения для карманных компьютеров.

3) Биометрические системы — это измеримые физиологические или поведенческие данные живого человека.

Некоторые биометрические данные уникальны для данного человека, и их можно использовать для установления личности или проверки декларируемых личных данных:

- для идентификации пользователя (вместо ввода имени пользователя);
- для однофакторной аутентификации пользователя;
- совместно с паролем или аутентификационным токеном (таким, как смарт-карта) для обеспечения двухфакторной аутентификации.

Биометрические данные делятся на группы:

1. Физиологические биометрические характеристики — основанные на данных, полученных путем измерения анатомических характеристик человека, таких, как, например, отпечаток пальца, форма лица или кисти, сетчатка глаза.

2. Поведенческие биометрические характеристики (динамические) — основанные на данных, полученных путем измерения действий человека. Характерной чертой для поведенческих характеристик является их протяженность во времени — измеряемое действие имеет начало, середину и конец. Например, голос.

### ***Криптографическая защита данных***

Криптография (иногда употребляют термин криптология) — область знаний, изучающая тайнопись (криптография) и методы ее раскрытия (криптоанализ). Криптография считается разделом математики.

Цель криптографической системы заключается в том, чтобы зашифровать осмысленный исходный текст (также называемый открытым текстом), получив в результате совершенно бессмысленный на взгляд зашифрованный текст (шифртекст, криптограмма). Получатель, которому он предназначен, должен быть способен расшифровать («дешифровать») этот шифртекст, восстановив, таким образом, соответствующий ему открытый текст. При этом противник (называемый также криптоаналитиком) должен быть неспособен раскрыть исходный текст.

Существует важное отличие между расшифрованием (дешифрованием) и раскрытием шифртекста. Пример простого алгоритма шифрования: Широко известным историческим примером криптосистемы является так называемый шифр Цезаря, который представляет из себя простую замену каждой буквы открытого текста третьей следующей за ней буквой алфавита (с циклическим переносом, когда это необходимо). Например, «А» заменялась на «D», «В» на «Е», «Z» на «С».

Все методы шифрования можно разделить на две группы:

- шифры с секретным ключом (симметричная схема);
- шифры с открытым ключом (асимметричная схема).

Первый тип шифров подразумевает наличие некоей информации (ключа), обладание которой позволяет как зашифровать, так и расшифровать сообщение. Шифры с открытым ключом подразумевают наличие двух ключей — открытого и закрытого; один используется для шифровки, другой для расшифровки сообщений.

### ***Электронная подпись (ЭП)***

ЭП — последовательность символов, полученная в результате криптографического преобразования исходной информации с использованием закрытого ключа ЭЦП, которая позволяет подтверждать целостность и неизменность этой информации, а также ее авторство при условии использования открытого ключа ЭП и его сертификата.

Цифровая подпись обеспечивает:

- Удостоверение источника документа. В зависимости от деталей определения «документа» могут быть подписаны такие поля как автор, внесённые изменения, метка времени.
- Защиту от изменений документа. При любом случайном или преднамеренном изменении документа (или подписи) изменится хэш, следовательно, подпись станет недействительной.
- Невозможность отказа от авторства. Так как создать корректную подпись можно лишь зная закрытый ключ, а он известен только владельцу, то владелец не может отказаться от своей подписи под документом.

Более мощным средством защиты данных от просмотра является их шифрование. Шифрование – это преобразование читаемого текста в

нечитаемый текст, при помощи некоторого алгоритма; применяется для защиты уязвимых данных.

Процесс дешифрования восстанавливает данные в исходное состояние.

В целях контроля использования основных ресурсов СУБД во многих системах имеются средства установления прав доступа к объектам БД. Права доступа определяют возможные действия над объектами. Владелец объекта (пользователь, создавший объект), а также администратор БД имеют все права. Остальные пользователи к разным объектам могут иметь различные уровни доступа. Разрешение на доступ к конкретным объектам базы данных сохраняется в файле рабочей группы.

Файл рабочей группы содержит данные о пользователях группы и считывается во время запуска. Файл содержит следующую информацию: имена учетных записей пользователей, пароли пользователей, имена групп, в которые входят пользователи.

По отношению к таблицам могут предусматриваться следующие права доступа:

- просмотр (чтение) данных;
- изменение (редактирование) данных;
- добавление новых записей;
- добавление и удаление данных;
- изменение структуры таблицы.

К данным, имеющимся в таблице, могут применяться меры защиты по отношению к отдельным полям и отдельным записям.

Защита данных в полях таблиц предусматривает следующие уровни прав доступа:

- полный запрет доступа;
- только чтение;
- разрешение всех операций (просмотр, ввод новых значений, удаление и изменение).

По отношению к формам могут предусматриваться две основные операции:

- вызов для работы и проектирование (режим Конструктора). Запрет вызова Конструктора целесообразно выполнять для экранных форм готовых приложений, чтобы конечный пользователь случайно не изменил приложение;
- защита отдельных элементов. Например, некоторые поля исходной таблицы вообще могут отсутствовать или скрыты от пользователя, а некоторые поля - доступны для просмотра.

Отчеты во многом похожи на экранные формы. На отчеты аналогично может накладываться запрет на вызов средств их разработки.

К дополнительным средствам защиты БД можно отнести такие, которые нельзя прямо отнести к средствам защиты, но которые непосредственно влияют на безопасность данных. Их составляют следующие средства:

- встроенные средства контроля значений данных в соответствии с типами;

- повышения достоверности вводимых данных;
- обеспечения целостности связей таблиц;
- организации совместного использования объектов БД в сети.

Восстановление базы данных с помощью резервного копирования базы данных, с помощью журнала транзакций.

Поскольку данные, хранимые компьютерными средствами подвержены потерям и повреждениям, вызываемым разными событиями, важно обеспечить средства восстановления данных. Приведение базы данных точно в то состояние, которое существовало перед отказом не всегда возможно, но процедуры восстановления базы данных могут привести ее в состояние, существовавшее незадолго до отказа.

Восстановление базы данных применяется при повреждениях, не позволяющих пользователю открыть базу данных или работать с ней. Одной из причин повреждения базы данных может быть воздействие компьютерных вирусов или наличие дефектов (физических или логических) на диске.

Повреждение базы данных может проявляться при попытке пользователя открыть, сжать, зашифровать или дешифровать БД.

Следующим способом обслуживания базы данных является резервное копирование. Основным назначением резервного копирования базы данных является предотвращение потери информации и реализуется путем одноразового или периодического копирования и архивирования наиболее ценной информации. Резервное копирование заключается в создании резервной копии базы данных и размещении на вспомогательных носителях информации.

Резервная копия может быть точной копией исходной БД или сжатой (архивной) копией.

Резервное копирование может осуществляться во время работы с БД (режим online) или в другое время. Копия может создаваться по инициативе оператора, либо автоматически в заданное время путем запуска соответствующей утилиты.

При организации резервного копирования администратор решает такие вопросы как:

- какие устройства выбрать для резервного копирования;
- когда и с какой частотой выполнять резервное копирование.

Важно периодически проверять корректность выполненного резервирования информации путем пробного восстановления.

Репликация (replication) – создание специальных копий (реплик) базы данных, с которыми пользователи могут работать одновременно на разных рабочих станциях.

Журнал транзакций БД — это особая часть БД, недоступная пользователям СУБД, в которую поступают записи обо всех изменениях основной части БД. Для эффективной реализации функции ведения журнала изменений БД необходимо обеспечить повышенную надежность хранения и поддержания в рабочем состоянии самого журнала. Иногда для этого в системе хранят

несколько копий журнала. В разных СУБД изменения базы данных фиксируются в журнале на разных уровнях. Иногда запись в журнале соответствует какой-то операции изменения БД (например, операции удаления строки из таблицы реляционной БД), а иногда — минимальной внутренней операции модификации страницы внешней памяти. В некоторых схемах используются оба подхода одновременно.

Небольшую и несложную базу данных или приложение Access можно создать в СУБД Access без использования языков программирования SQL и Visual Basic. В СУБД Access имеется достаточно средств (различных мастеров и конструкторов) для визуального проектирования объектов базы данных. Для решения некоторых задач автоматизации приложений Access можно использовать макросы вместо языка программирования Visual Basic (например, при создании кнопочной формы). Но создание коммерческих баз данных в СУБД Access невозможно без применения визуального языка программирования Visual Basic и языка запросов SQL.

Для автоматизации действий над объектами в Microsoft Access и в других приложениях Microsoft Office применяются макросы и модули.

Макросы - это небольшие программы на языке макрокоманд (языке сценариев). Модули - это объекты, содержащие программы на языке Visual Basic.

Основное назначение макросов и модулей — это создание удобного интерфейса приложения.

Стандарты DAO и ADO обеспечивают программиста похожим набором инструментов управления. DAO – это более ранний стандарт средств управления базой данных. Стандарт ADO представляет более мощные инструменты и его применение приводит к сокращению программного кода.

### ***Тема 1.4. Введение в язык SQL***

Для извлечения данных из базы данных используется язык SQL.

SQL (Structured Query Language) — это язык, предназначенный для программ управления базами данных. В 1986 г. ANSI и ISO официально приняли стандартное определение языка «Язык баз данных SQL». Новые версии стандарта были опубликованы в 1989, 1992, 1996, 1999, 2003, 2006, 2008, 2011 и в 2016 году. Последняя действующая редакция стандартов языка SQL – ISO/IEC 9075:2016.

В Ms Access SQL используется в каждом запросе.

Структурированный язык запросов SQL реализуется в следующих формах:

- интерактивный SQL;
- статический SQL;
- динамический SQL;
- встроенный SQL.

Выделяют следующие виды SQL запросов:

- 1) DDL (Data Definition Language) – язык определения данных. Задачей DDL запросов является создание БД и описание ее структуры. Запросами такого вида устанавливаются правила того, в каком виде различные данные будут размещаться в БД.
- 2) DML (Data Manipulation Language) – язык манипулирования данными. В число запросов этого типа входят различные команды, используя которые непосредственно производятся некоторые манипуляции с данными. DML-запросы необходимы для добавления изменений в уже внесенные данные, для получения данных из БД, для их сохранения, для обновления различных записей и для их удаления из БД. В число элементов DML-обращений входит основная часть SQL операторов.
- 3) DCL (Data Control Language) – язык управления данными. Включает в себя запросы и команды, касающиеся разрешений, прав и других настроек СУБД.
- 4) TCL (Transaction Control Language) – язык управления транзакциями. Конструкции такого типа применяют чтобы управлять изменениями, которые производятся с использованием DML запросов. Конструкции TCL позволяют производить объединение DML запросов в наборы транзакций.

Например, простая инструкция SQL, извлекающая список фамилий контактов с именем Mary, может выглядеть следующим образом:

```
SELECT Last_Name  
FROM Contacts  
WHERE First_Name ="Mary";
```

Общий формат инструкции SQL:

```
SELECT <что выводится>  
FROM <откуда>  
WHERE <каким условиям должно отвечать>  
GROUP BY <какие поля группируются>  
HAVING <условие для сгруппированных данных>  
ORDER BY <в каком порядке выводить данные>.
```

Операторы (команды), написанные на языке SQL, лишь указывают СУБД, какой результат должен быть получен, но не описывают процедуру получения этого результата. СУБД сама определяет способ выполнения команды пользователя.

Операторы языка SQL строятся с применением:

- зарезервированных ключевых слов;

- идентификаторов (имен) таблиц и столбцов таблиц;
- логических, арифметических и строковых выражений, используемых для формирования критериев поиска информации в БД и для вычисления значений ячеек результирующих таблиц;
- идентификаторов (имен) операций и функций, используемых в выражениях.

SQL — это следующие функциональные возможности:

- определение данных — дает возможность разработчику определять структуру хранения данных и отношения между элементами данных;
- выборка данных — дает возможность разработчику использовать любые данные из БД для своих целей;
- обработка информации — дает возможность разработчику делать с данными что угодно: добавлять, изменять, удалять и не только;
- управление доступом — дает возможность разработчику обезопасить информацию в БД от непреднамеренного использования;
- совместное применение данных — дает возможность нескольким разработчикам одновременно работать с информацией в одной БД, при этом организует работу таким образом, чтобы действия одного разработчика не навредили действиям другого разработчика;
- целостность данных — дает возможность разработчику обезопасить данные от случайного разрушения при отказе системы или случайных изменений.

Основные преимущества языка структурированных запросов SQL приведены на рисунке 1.2.

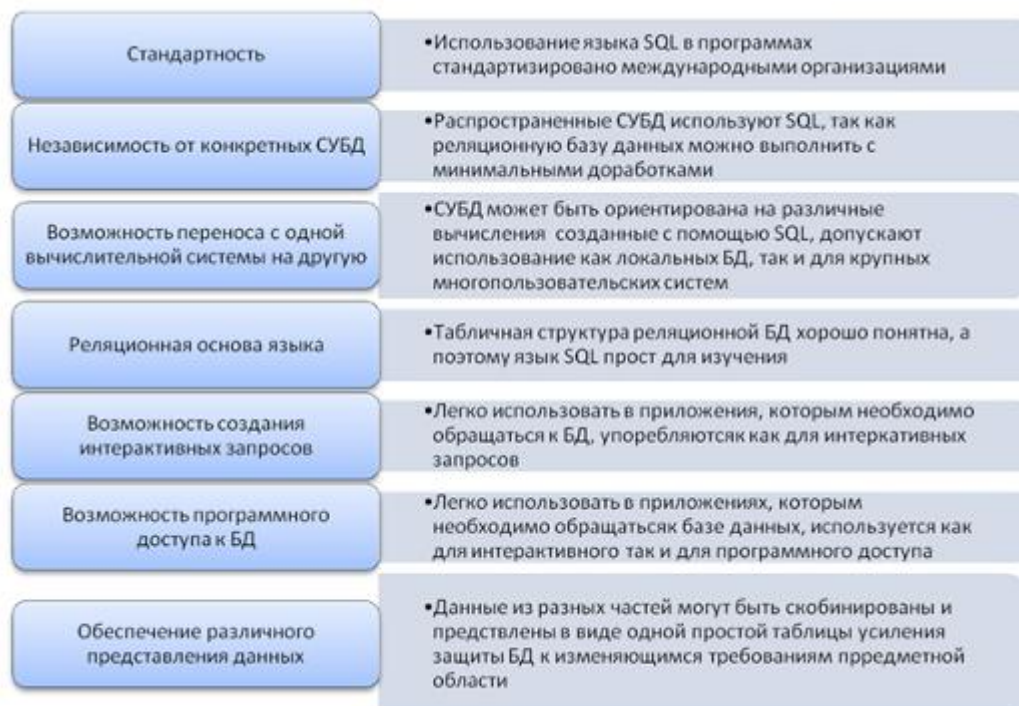


Рис. 1.2. Преимущества структурированного языка запросов



Язык универсален и обладает чётко определённой структурой за счёт устоявшихся стандартов. Взаимодействие с базами данных происходит быстро даже в ситуациях, когда объёмы данных велики (Big Data). Кроме того, эффективное управление возможно даже без особых познаний кода.

## РАЗДЕЛ II. СЕТЕВЫЕ ТЕХНОЛОГИИ

### *Тема 2.1. Введение в компьютерные сети. Аппаратное и программное обеспечение компьютерных сетей. Протоколы компьютерных сетей*

Компьютерная сеть - совокупность взаимосвязанных через каналы передачи данных компьютеров, обеспечивающих пользователей средствами обмена информацией и коллективного использования ресурсов сети:

1) аппаратных, 2) программных, 3) информационных.

Компьютерная сеть (вычислительная сеть, сеть передачи данных) - система связи двух или более компьютеров и/или компьютерного оборудования (серверы, маршрутизаторы и другое оборудование). Для передачи информации могут быть использованы различные физические явления, как правило - различные виды электрических сигналов или электромагнитного излучения.

Компьютерные сети классифицируются:

1) по размеру охваченной территории: персональная, локальная, городская, национальная, глобальная вычислительная сеть;

2) по типу функционального взаимодействия: клиент-сервер, смешанная сеть, точка-точка, одноранговая сеть, многоранговые сети;

3) по типу сетевой топологии: шина, звезда, кольцо, решётка, смешанная топология, полносвязная топология;

4) по функциональному назначению: сети хранения данных, серверные фермы, сети управления процессом;

5) по сетевым операционным системам;

6) по необходимости поддержания постоянного соединения: пакетная или онлайн-сеть, например, Интернет и GSM.

Передача данных может осуществляться по проводной связи (телефонная сеть (модем и коммутированный доступ), выделенные линии, коммутация пакетов, передача по оптоволокну), или по беспроводной связи: ближнего, среднего или дальнего радиуса действия (спутниковая связь, мобильные телефоны).

Локальная вычислительная сеть (ЛВС) - группа ЭВМ, а также периферийное оборудование, объединенные в пределах одного или нескольких близлежащих зданий автономными (не арендуемыми) высокоскоростными каналами передачи цифровых данных (проводными или беспроводными: медными, волоконно-оптическими, СВЧ или ИК-диапазона). Служит для решения информационных задач (например, в рамках какой-либо организации),

а также совместного использования объединенных информационных и вычислительных ресурсов. ЛВС могут иметь в своем составе средства для выхода в Интернет.

Каждый компьютер, включаемый в локальную сеть должен иметь сетевую плату, в разъем которой и подключается связующий кабель. Кабели, выходящие из различных компьютеров, объединяются в устройстве, называемом сетевой концентратор (switch, HUB). Сетевые концентраторы могут иметь связь друг с другом, объединяя вместе подсети различных участков здания. Таким образом, обеспечивается прохождение сигналов между всеми устройствами, включенными в сеть.

Преимущества работы в локальной сети:

- 1) возможность хранения данных персонального и общего использования (программного обеспечения, необходимого многим пользователям) в единственном экземпляре на дисках файлового сервера;
- 2) обмен информацией между всеми компьютерами сети;
- 3) одновременная печать всеми пользователями сети на общесетевых принтерах;
- 4) обеспечение доступа с любого компьютера локальной сети к ресурсам Интернет, при наличии единственного коммуникационного узла глобальной сети.

В зависимости от принципов построения ЛВС подразделяются на следующие основные типы:

Одноранговая ЛВС (peer-to-peer, p2p LAN) – «безсерверная» организация построения сети. Термин «одноранговая» означает, что все компьютеры, объединенные в сеть, имеют в ней одинаковые права. Каждый пользователь одноранговой сети может определить состав файлов, которые он предоставляет для общего использования. Таким образом, пользователи одноранговой сети могут работать как со всеми своими файлами, так и с файлами, предоставляемыми другими ее пользователями. Создание одноранговой сети обеспечивает совместную эксплуатацию периферийных устройств. Серверные функции при этом распределены по сети. Компьютер, к которому подключен принтер, может являться для всех остальных сервером печати, компьютер, к которому присоединен модем – сервером Интернет.

ЛВС с выделенным сервером (Dedicated Server Network: file-server или client-server architecture) – означает, что специально выделенный самый мощный компьютер (сервер) в сети берет на себя основные функции по её обслуживанию: управляет созданием, поддержкой и использованием общих информационных ресурсов, включая доступ к её базам данных и отдельным файлам, а также их защиту и аудит.

К нему же присоединяются основные периферийные устройства: модем, принтер. Следовательно, серверные функции не рассредоточены по сети, а централизованы. Один и тот же компьютер-сервер может являться и файловым

сервером, и Интернет-сервером, и сервером печати. Остальные компьютеры сети тогда именуют «клиентами» или «рабочими станциями».

Интернет (Internet, от Interconnected Networks - объединённые сети) - глобальная телекоммуникационная сеть информационных и вычислительных ресурсов. Служит физической основой для Всемирной паутины. Часто упоминается как Всемирная сеть, Глобальная сеть, либо просто Сеть.

Обращаясь в Интернет, пользуются услугами Интернет-провайдера или ISP (Internet Service Provider - поставщик услуг Интернета). ISP - это организация, которая имеет собственную высокоскоростную сеть, объединенную с другими сетями по всему земному шару. Провайдер подключает к своей сети клиентов, которые становятся частью сети данного провайдера и одновременно частью всех объединенных сетей, которые и составляют Интернет.

В более широком смысле Интернет - это информационное пространство, распределенное среди миллионов компьютеров во всем мире, которые постоянно обмениваются данными. Основная задача Интернета - это связь. Связь круглосуточная, высоконадежная. Для того чтобы осуществлять такую высоконадежную связь, была разработана специальная Интернет-технология доставки данных.

Существует достаточно много толкований термина Интернет, однако он имеет два основных качественных значения:

- глобальное сообщество произвольно объединяемых мировых сетей, которые используются для свободного обмена данными, информацией и знаниями;
- совокупность технологий, которые реализуют обмен данными на основе использования семейства протоколов TCP/IP (Transmission Control Protocol/Internet Protocol), называемых Интернет-технологиями.

Всемирная паутина (World Wide Web) — распределенная система, предоставляющая доступ к связанным между собой документам, расположенным на различных компьютерах, подключенных к Интернету. Всемирную паутину образуют более миллиона web-серверов. Большинство ресурсов всемирной паутины представляет собой гипертекст. Гипертекстовые документы, размещаемые во всемирной паутине, называются web-страницами. Несколько web-страниц, объединенных общей темой, дизайном, а также связанных между собой ссылками и обычно находящихся на одном и том же web-сервере, называются web-сайтом. Для загрузки и просмотра web-страниц используются специальные программы браузеры.

Основными элементами технологии WWW являются:

- язык гипертекстовой разметки документов (Hyper Text Markup Language - HTML);
- протокол обмена гипертекстовой информацией (Hyper Text Transfer Protocol - HTTP);

- универсальный способ адресации ресурсов в сети (Universal Resource Identifier - URI, и Universal Resource Locator - URL);
- система доменных имен (Domain Name System - DNS);
- универсальный интерфейс шлюзов (Common Gateway Interface - CGI), добавленный позже сотрудниками Национального Центра Суперкомпьютерных Приложений (National Center for Supercomputing Applications - NCSA).
- расширяемый язык разметки (eXtensible Markup Language - XML), рекомендованный Консорциумом Всемирной паутины.

Всемирную паутину образуют миллионы веб-серверов сети Интернет, расположенные по всему миру. Веб-сервер является программой, запускаемой на подключённом к сети компьютере и использующей протокол HTTP для передачи данных. В простейшем виде такая программа получает по сети HTTP-запрос на определённый ресурс, находит соответствующий файл на локальном жёстком диске и отправляет его по сети запросившему компьютеру. Более сложные веб-серверы способны динамически распределять ресурсы в ответ на HTTP-запрос. Для идентификации ресурсов (зачастую файлов или их частей) во Всемирной паутине используются единообразные идентификаторы ресурсов URI (Uniform Resource Identifier). Для определения местонахождения ресурсов в сети используются единообразные локаторы ресурсов URL (Uniform Resource Locator). Такие URL-локаторы сочетают в себе технологию идентификации URI и систему доменных имён DNS (Domain Name System) - доменное имя (или непосредственно IP-адрес в числовой записи) входит в состав URL для обозначения компьютера (точнее - одного из его сетевых интерфейсов), который исполняет код нужного веб-сервера.

Для просмотра информации, полученной от веб-сервера, на клиентском компьютере применяется специальная программа – веб-браузер. Основная функция веб-браузера - отображение гипертекста. Всемирная паутина неразрывно связана с понятиями гипертекста и гиперссылки.

Работа по разметке гипертекста называется вёрсткой, мастера по разметке называют веб-мастером или вебмастером. После HTML-разметки получившийся гипертекст помещается в файл, такой HTML-файл является самым распространённым ресурсом Всемирной паутины. После того, как HTML-файл становится доступен веб-серверу, его начинают называть «веб-страницей». Набор веб-страниц образует веб-сайт. В гипертекст веб-страниц добавляются гиперссылки. Гиперссылки помогают пользователям Всемирной паутины легко перемещаться между ресурсами (файлами) вне зависимости от того, находятся ресурсы на локальном компьютере или на удалённом сервере. Гиперссылки основаны на технологии URL.

Для улучшения визуального восприятия веб-страниц стала широко применяться технология CSS, которая позволяет задавать единые стили оформления для множества веб-страниц. Ещё одно нововведение, на которое

стоит обратить внимание, - система обозначения ресурсов URN (Uniform Resource Name).

Информация в Интернете может отображаться как пассивно (то есть пользователь может только считывать её), так и активно - тогда пользователь может добавлять информацию и редактировать её. К способам активного отображения информации во Всемирной паутине относятся: гостевые книги, форумы, чаты, блоги, wiki-проекты, системы управления контентом. Следует отметить, что это деление весьма условно.

Интернет состоит из многих тысяч корпоративных, научных, правительственных и домашних компьютерных сетей. Объединение сетей разной архитектуры и топологии стало возможно благодаря протоколу IP (Internet Protocol) и принципу маршрутизации пакетов данных.

Чтобы компьютеры, объединенные в сеть, могли обмениваться сообщениями, каждый из них должен иметь уникальный адрес. В сети Интернет это 32-х разрядный (32-х битный = 4-х байтный) адрес, называемый IP-адрес.

В точечно-десятичной нотации IP-адрес выглядит, например, так: 220.215.14.22. Каждая часть, разделенная точкой, представляет собой один байт, и, следовательно, максимальное десятичное число, которое может быть представлено одним байтом 255. Но, для человека такая система адресации сложна, так же как нам сложно помнить, набирать и диктовать одиннадцатизначные телефонные номера, поэтому в 1984 году Полом Мокапетрисом была разработана надстройка над IP-адресацией, называемая системой DNS (domain name system, система доменных имен).

На стыках сетей специальные маршрутизаторы (программные или аппаратные) занимаются автоматической сортировкой и перенаправлением пакетов данных, исходя из IP-адресов получателей этих пакетов.

Для того чтобы при обмене данными компьютеры, объединенные в сеть, действовали согласованно, разработан ряд стандартов и правил, называемых протоколами. Весь набор сетевых протоколов, на которых базируется Интернет называется TCP/IP.

Протокол - это правила передачи данных между узлами компьютерной сети. Систему протоколов Интернет называют «стеком протоколов TCP/IP».

Протоколы, входящие в семейство TCP/IP разделяются на уровни.

Физический уровень описывает среду передачи данных, физические характеристики такой среды и принцип передачи данных (разделение каналов, модуляцию, амплитуду сигналов, частоту сигналов, способ синхронизации передачи, время ожидания ответа и максимальное расстояние).

Канальный уровень описывает, каким образом передаются пакеты данных через физический уровень, включая кодирование (т.е. специальные последовательности битов, определяющих начало и конец пакета данных).

Сетевой уровень изначально разработан для передачи данных из одной сети (подсети) в другую.

Протоколы транспортного уровня могут решать проблему гарантированной доставки сообщений, гарантировать правильную последовательность прихода данных. Транспортные протоколы определяют, для какого именно приложения предназначены эти данные.

На прикладном уровне работает большинство сетевых приложений. Эти программы имеют свои собственные протоколы обмена информацией, например, HTTP для WWW, FTP (передача файлов), SMTP (электронная почта), SSH (безопасное соединение с удалённой машиной), DNS (преобразование символьных имён в IP-адреса) и многие другие.

Доменные имена системы DNS – синонимы IP-адреса, так же, как имена в адресной книжке вашего телефона – синонимы телефонных номеров. Они символьные, а не числовые; они удобнее для запоминания и ориентации; они несут смысловую нагрузку.

Доменные имена также уникальны, нет в мире двух одинаковых доменных имён. Доменные имена, в отличие от IP-адресов необязательны, они приобретаются дополнительно.

Чтобы найти в Интернете какой-либо документ, достаточно знать ссылку на него - так называемый универсальный указатель ресурса (URL - Uniform Resource Locator), который определяет местонахождение каждого файла, хранящегося на компьютере, подключенном к Интернету. Адрес URL является сетевым расширением понятия полного имени ресурса в операционной системе. В URL, кроме имени файла и директории, где он находится, указывается сетевое имя компьютера, на котором этот ресурс расположен, и протокол доступа к ресурсу, который можно использовать для обращения к нему. Система адресации URL и адресация почтовой службы имеют сходную структуру.

ICMP (Internet Control Message Protocol) — это протокол третьего уровня модели OSI, который используется для диагностики проблем со связностью в сети. ICMP помогает определить может ли достичь пакет адреса назначения в установленные временные рамки.

Протокол определения адреса (ARP) — это протокол или процедура, которая соединяет постоянно изменяющийся адрес интернет-протокола (IP) с фиксированным физическим адресом оборудования, известным как адрес контроля доступа к медиаинформации (MAC), в локальной сети (LAN).

Протокол транспортного уровня UDP (User Datagram Protocol) позволяет посылать сообщения (в данном случае называемые датаграммами) другим хостам по IP-сети без необходимости предварительного сообщения для установки специальных каналов передачи или путей данных.

Порт — это числовое значение, назначаемое каждому приложению или процессу. При связи данные передаются от отправляющего устройства к устройству назначения (приемнику). Если на целевом устройстве запущено три приложения, номер порта помогает определить приложение, для которого требуются полученные данные. Компьютер имеет в общей сложности 65536

портов. Номера портов от 0 до 1023 зарезервированы для использования системой. Порт помогает идентифицировать процесс, который требует полученных данных.

Комбинация IP-адреса и порта называется сокетом.

Разница между сокетом и портом заключается в том, что сокет является интерфейсом отправки и получения данных на конкретном порту, а порт - это числовое значение, назначенное конкретному процессу или приложению на устройстве.

Рабочее предложение (англ. Request for Comments, RFC) — документ из серии пронумерованных информационных документов Интернета, содержащих технические спецификации и стандарты, широко применяемые во всемирной сети.

Сетевое программное обеспечение предназначено для организации совместной работы группы пользователей на разных компьютерах; позволяет организовать общую файловую структуру, общие базы данных, доступные каждому члену группы; обеспечивает возможность передачи сообщений и работы над общими проектами, возможность разделения ресурсов.

Сетевые операционные системы (Network Operating System – NOS) – это комплекс программ, обеспечивающих обработку, хранение и передачу данных в сети. Сегодня практически все современные ОС имеют встроенные сетевые функции.

Сетевая операционная система выполняет функции прикладной платформы, предоставляет разнообразные виды сетевых служб и поддерживает работу прикладных процессов, выполняемых в абонентских системах. Сетевые операционные системы используют клиент-серверную, либо одноранговую архитектуру. Компоненты NOS располагаются на всех рабочих станциях, включенных в сеть.

NOS обеспечивает выполнение основных функций сети. К ним, в первую очередь, относятся:

1. адресация объектов сети;
2. функционирование сетевых служб;
3. обеспечение безопасности данных;
4. управление сетью.

Приведем примеры сетевых операционных систем: Novell NetWare, LANtastic, Microsoft Windows (NT, XP, Vista, 7, 8, 8.1, 10, 11), различные GNU/Linux и UNIX системы (Solaris, FreeBSD).

Корпоративная сеть – сложная система, включающая множество самых разнообразных компонентов: серверные комплексы, рабочие станции, системное и сетевое программное обеспечение, базы данных, активное и телекоммуникационное оборудование, структурированную кабельную систему. Основная задача состоит в том, чтобы система как можно лучше справлялась с теми функциями, которые на нее возлагаются.

Для поддержания сети в состоянии, соответствующем требованиям времени, необходимо внедрение разнообразных современных технических, технологических и организационных новшеств.

Сетевые технологии, разработанные за последнее время, подготовили фундамент для выполнения корпоративными сетями ряда задач, которые ранее решались при помощи отдельных систем. При помощи интеграции функций корпоративной сети с функциями таких подсистем, как корпоративная телефония, обработка факсимильных сообщений, видеоконференцсвязи, можно достигнуть значительной экономии накладных расходов, сконцентрировавшись на использовании сети в качестве единой информационной системы предприятия. Как следствие, на корпоративную сеть возлагается ответственность за выполнение всех информационных функций, которые составляют основу нормальной работы современного предприятия. Поэтому требования, предъявляемые к надежности функционирования сети, безопасности и целостности корпоративных данных, ужесточаются.

С ростом сложности и увеличением размера корпоративных систем обостряются проблемы их поддержки и сохранения средств, инвестированных в сетевую инфраструктуру. Главный вопрос состоит в том, позволит ли существующая инфраструктура корпоративной сети внедрять в дальнейшем новые технологии, и насколько существенные затраты потребуются для этого. В том случае, если при проектировании сети было уделено достаточное внимание гибкости ее архитектуры и заложен необходимый резерв для будущих применений, при внедрении новых приложений не будет возникать значительных проблем. Необходимая для этого гибкость закладывается на этапе планирования корпоративной сети, предшествующем проектированию.

Планирование сети состоит в нахождении компромисса между потребностями предприятия, его финансовыми возможностями и возможностями сетевых и информационных технологий сегодня и в будущем.

## ***Тема 2.2. Глобальные компьютерные сети***

Глобальные сети – это распределённые сети всемирного масштаба. Самой известной и доступной глобальной сетью является Интернет (Internet) - самая большая в мире совокупность разнотипных компьютерных сетей.

Рассмотрим наиболее распространенные телекоммуникационные системы, или территориальные сети связи. К ним относятся X.25, Frame Relay (FR), АТМ и Internet.

Основу X.25 сетей составляют Центры Коммутации Пакетов (ЦКП), расположенные во многих городах и обеспечивающие доступ к сети. Обычно абонент получает доступ к сети, соединяясь с ближайшим ЦКП. Следовательно, можно получить доступ к сети из любого места, где есть телефонная связь, без привязки к конкретному ЦКП. Абоненты сети



подключаются к ней для того, чтобы передавать информацию или принимать ее от других абонентов или хост-машин. Для этого в сети устанавливается временная логическая связь между этими абонентами, называемая виртуальным соединением. После установления виртуального соединения между абонентами может происходить обмен данными одновременно в двух направлениях (дуплекс), причем задержка передачи пакетов данных не превышает долей или нескольких секунд в зависимости от загруженности сети.

Ретрансляция кадров FR (Frame Relay) – это технология сети передачи данных с пакетной коммутацией, является усовершенствованной сетью стандарта X.25 и использует технику виртуальных соединений. Сети FR используют на физическом уровне цифровые каналы связи, которые лучше по сравнению с аналоговыми каналами по показателю коэффициента ошибок.

Технология асинхронного режима передачи ATM - Asynchronous Transfer Mode разработана как единый универсальный транспорт для нового поколения сетей с интеграцией услуг, которые называются широкополосными сетями ISDN - Integrated Service Digital Network - цифровая сеть с интеграцией услуг.

Технология ATM является дальнейшим развитием идей предварительного канала, реализованных в технологии FR.

Сеть Internet (INTERconnection NETwork) является синтезом многих локальных, корпоративных и национальных сетей, обладающих часто собственными внутренними разнообразными линиями связи и протоколами. Компьютеры, подключенные к сети Интернет, могут иметь любые аппаратные и программные платформы, но при этом они должны поддерживать стек протоколов (семейство протоколов) связи TCP/IP. Объединяет их в единую глобальную сеть система серверов различной аппаратной и программной конфигурации. Эти сервера объединены между собой спутниковыми и оптоволоконными линиями связи, реже коаксиальными и иными кабелями.

Каждое подключенное к сети Internet устройство (узел, host) однозначно адресуется 32-значным уникальным двоичным числом (разбиваемым точкой на 4 октета - например, 198.137.240.91). Адрес узла логически разделяется на две части, одна из которых называется идентификатором сети (Network ID), а другая - идентификатором узла (Host ID).

Компьютер в сети TCP/IP может иметь адреса трех уровней (но не менее двух):

- Локальный адрес компьютера. Для узлов, входящих в локальные сети - это MAC-адрес сетевого адаптера. Эти адреса назначаются производителями оборудования и являются уникальными адресами.
- IP-адрес, состоящий из 4 байт, например, 109.26.17.100. Этот адрес используется на сетевом уровне. Он назначается администратором во время конфигурирования компьютеров и маршрутизаторов.
- Символьный идентификатор-имя (DNS), например, [www.bntu.by](http://www.bntu.by).

Автономная система (AS) в Интернете — это система IP-сетей и маршрутизаторов, управляемых одним или несколькими операторами, имеющими единую политику маршрутизации с Интернетом.

WHOIS – это система, содержащая информацию о том, кто владеет тем или иным доменом либо IP адресом.

Сеть бывает проводной и беспроводной. В случае первого варианта должны быть любые два устройства, которые соединены проводом для взаимодействия друг с другом. В качестве соединения может быть использован волоконно-оптический или медный кабель. При таком типе соединения функциональные возможности передачи данных ограничены физическими свойствами кабеля. Проводная сеть регламентируется стандартом IEEE 802.3, в котором четко прописаны все основные требования.

Изначально для выхода в сеть использовалась телефонная линия, а также специальное устройство — модем. При этом скорость передачи данных была небольшой.

Сейчас скорость проводного интернета выросла, и нет необходимости приобретения дополнительного оборудования. Ведь во все материнские платы встроена сетевая карта.

Недостаток проводной сети в том, что она требует прокладки кабелей к каждому рабочему месту. Поскольку разводку обычно делают сразу при ремонте помещения, то любые изменения в организации работы также напрямую коснутся сетевой инфраструктуры.

В беспроводных технологиях сигнал интернета передается по радиоканалам высокочастотных диапазонов, или при помощи лазерного, оптического и инфракрасного излучения. Регламентируется работа беспроводного подключения Wi-Fi протоколом IEEE 802.11X, который постоянно совершенствуется.

Для раздачи беспроводного сигнала на компьютер, телефон, планшет, ноутбук используется роутер (он может называться точкой доступа или маршрутизатором).



К недостаткам Wi-Fi можно отнести то, что качество связи напрямую зависит от расстояния до точки доступа и наличия различных препятствий на пути радиоволн в виде стен, крупных объектов. Причем скорость передачи данных равномерно распределяется между всеми подключенными устройствами.

Виды технологий для подключения к сети Интернет

1. Подключение через модем, используя телефонный кабель

Данный вид интернета предполагает коммутированный доступ к всемирной паутине, который является «самым древним». Для настройки проводного подключения необходим стационарный телефон и оператор связи,

оборудование которого сможет обеспечить такую услугу. Для подключения к сети провайдера модем набирает номер пула и соединяется с идентичным устройством на станции.

Неудобство заключено в том, что связь проходит по той же линии, которую использует телефонный аппарат. Это исключает возможность одновременного подключения.

## 2. ADSL-подключение

В данном случае технология обеспечивает подключение к интернету благодаря асимметричному распределению пропускной способности и приоритету для передачи данных.

Для ADSL-подключения необходимо наличие телефонной линии, но в том случае работа в сети не исключает возможность дозвона в одно и то же время. Это было достигнуто за счет отделения используемых частот для передачи голоса и данных.

## 3. Подключение через кабельное телевидение

В этом случае для доступа к сети используется обыкновенный телевизионный кабель. Передача данных проходит по стандарту DOCSIS. Для успешного подключения требуется наличие кабельного телевидения и специального устройства — модема.

Кабельное подключение бывает локальным и виртуальным.

### 3.1. Доступ через Ethernet-кабель

Данный тип подключения предполагает выход в сеть напрямую с использованием выделенной линии Ethernet. Для подключения необходимо наличие широкополосного интернета пользователя.

### 3.2. Подключение через GPON по оптоволокну

Данная технология одновременно по одному каналу связи предоставляет несколько сервисов. GPON подключение предполагает использование специального оптоволокна, которое устанавливается у пользователя, и оптического модема — ONT. Такой модем уже оснащен модулем, что не требует установки проводов для подключения устройств.

## 4. Антенный вид доступа

Существует мобильный, радиоволновый и спутниковый тип доступа. В первом случае выход в сеть возможен везде в зоне покрытия выбранного провайдера.



Выделяют следующие виды доступных мобильных интернет соединений:

- GPRS;
- 3G;
- 4G/LTE.

В случае с радио доступом к сети подключение осуществляется через оборудование, предоставленное провайдером. Сигнал поступает через антенну,

установленную у пользователя. Затем модем перенаправляет его на компьютер. Для увеличения радиуса действия сигнала может быть использован дополнительно усилитель.

Спутниковый интернет предполагает использование уличных антенн, которые направлены в сторону спутника.

При выборе типа подключения необходимо обращать внимание на скорость интернета.

Интернет является совокупностью эффективных методов коммуникации (на базе современных стандартизированных протоколов связи) и работы с информацией, находящейся на удаленных носителях. Кроме непосредственных функций по транзиту данных любых типов технологии Интернет обеспечивают широкий спектр разнообразных информационных услуг, реализуемых различными службами:

- пересылки и приема сообщений (E-mail);
- гипертекстовой среды (WWW);
- передачи файлов (File Transfer Protocol - FTP);
- удаленного управления компьютером (Teletype Network - Telnet);
- имен доменов (Domain Name System);
- телеконференций (Users Network - Usenet) и чат-конференций (Интернет Relay Chat - IRC).

Программная индустрия для Web создают новые технологии и инструментальные средства для навигации, работы в Сети и разработки пользовательских приложений. К их числу можно отнести:

- программы просмотра и навигации (браузеры);
- средства поиска и доставки информации (поисковые машины);
- программное обеспечение Интернет и Web-серверов, серверные приложения и расширения;
- средства администрирования в сетях. Поставляются в составе программного обеспечения Web-сервера и служат для конфигурирования, активации и мониторинга Web-сервисов, для контроля актуальности гиперссылок и связности гипертекстовой структуры, для учета и протоколирования использования серверов, для настройки и сопровождения системы безопасности;
- клиентские приложения и расширения (Web-сервисы);
- инструментальные средства разработки;
- средства обеспечения безопасности, которые могут быть встроены в программное обеспечение Интернет-серверов или представлены в виде дополнительных компонентов: комплексов Firewall и Proxy-серверов, выполняющих фильтрацию данных на различных уровнях.

Инструментальные средства разработки Интернет-приложений разнообразны и включают:

- редакторы гипертекста и графические редакторы. Редакторы гипертекста формируют HTML-файлы в режимах программирования

или WYSIWYG (What You See Is What You Get). Можно использовать обычные текстовые редакторы и средства, встроенные в браузеры. К этой же группе относятся конверторы, сохраняющие офисные документы в гипертекст. Графические редакторы служат для создания изображений, включаемых в гипертекст;

- средства разметки карт изображений и конверторы изображений. Позволяют разбить изображение на участки и связать гиперссылки с каждым из них. Могут быть встроены в графический редактор. Конверторы изображений обеспечивают преобразование форматов, размеров и цветов, создание специальных эффектов;
- средства мультимедиа (аудио, анимация, видео). Предназначены для создания звукового и музыкального сопровождения, анимационных и видеороликов. Часто воспроизведение файлов мультимедиа осуществляется клиентскими расширениями или специальными Helper-программами;
- средства генерации виртуальной реальности и управление ими на языке VRML (Virtual Reality Modeling Language). Позволяют запрограммировать трехмерные сцены. Ввиду того, что процесс воспроизведения виртуальной реальности достаточно сложен, могут потребоваться дополнительные средства автоматизированного проектирования и анимации;
- средства и языки программирования серверных и клиентских приложений и расширений. Предназначены для разработки и отладки сценариев (на языках *VBScript* или *JavaScript*) и мобильных приложений (на языке *Java*), выполняемых на стороне клиента. В качестве средств программирования серверных приложений могут применяться как обычные системы программирования (*Visual Basic*, *C/C++*, *Java*), так и интерпретаторы команд (*UNIX-shell*, *REXX* и др.) и интерпретаторы и компиляторы сценариев на *JavaScript*, *VBScript* и *Perl*. Для создания клиентских и серверных расширений используются системы программирования, которые позволяют создавать компоненты с использованием механизмов *ActiveX* или *Plug-in*, представленных в виде встроенных или дополнительных библиотек интерфейсов.

*Сервисы обмена данными на базе сети Интернет:*

*E-mail.* Позволяет обмениваться текстовыми сообщениями, к которым присоединяются файлы любых типов.

*FTP (File Transfer Protocol)* - протокол передачи файлов. Служит для перемещения файлов между компьютерами сети Интернет.

*Telnet.* Позволяет подключаться по Интернет к удаленному компьютеру и работать с ним так, как будто пользователь находится за его терминалом.

*WWW.* Обеспечивает доступ к гигантскому объему информации: текст, графика, аудио- и видеопрограммы. В основе – технология гипертекста.

На ранних стадиях развития сеть Интернет была «улицей с односторонним движением», так как информация с *Web*-страниц поступала к пользователю от *Web*-сервера только при наличии запроса пользователя. С появлением в языке *HTML* диалоговых свойств пользователь получил обратную связь с *Web*-сервером. Обмен параметров при этом осуществляется через специальный графический интерфейс (*Computer Graphical Interface - CGI*).

В последнее время все большее распространение получает механизм согласования запускаемых программ через многоцелевые расширения почтовой службы Интернет (*Multipurpose Internet Mail Extensions - MIME*). Современные браузеры, помимо взаимодействия с *Web*-серверами через протокол *http*, могут работать с различными типами серверов и служб с использованием протоколов *FTP, File, Gopher, Mailto, NNTP, Telnet, WAIS*. В состав *URL* входит информация о методе доступа, требующаяся браузеру, чтобы использовать любой из этих протоколов.

*Intranet* - внутреннее информационное пространство организации, реализуемое либо в локальной сети *LAN (Local Area Network)*, либо в компьютерной сети *WAN (Wide Area Network)*, охватывающей несколько территорий и включающей в себя десятки и/или сотни тысяч компьютеров) и обладающее всеми возможностями Интернет.

Наряду с понятиями гипертекста и протокола *HTTP*, понятие *URL (Uniform Resource Locator)* является одной из основных концепций Всемирной паутины. Это механизм, используемый браузерами для получения ресурса опубликованного во Всемирной сети (*HTML*-страницы, *CSS*-файла, изображения).

Веб-сайт (*Website*, от *web* - паутина и *site* - «место») - совокупность документов частного лица или организации в компьютерной сети объединённая под одним адресом (доменным именем или *IP*-адресом). В настоящее время большинству из них свойственна динамичность и интерактивность - способность информационно-коммуникационной системы, без участия человека, активно и разнообразно реагировать на действия пользователя.

В большинстве случаев в Интернете одному веб-сайту соответствует одно доменное имя. Именно по доменным именам сайты идентифицируются в глобальной сети. Возможны иные варианты: один сайт на нескольких доменах или несколько сайтов под одним доменом. Обычно несколько доменов используют крупные сайты (веб-порталы) чтобы логически отделить разные виды предоставляемых услуг (*mail.google.com, news.google.com, maps.google.com*). Нередки и случаи выделения отдельных доменов для разных стран или языков. Например, *google.ru* и *google.fr* логически являются сайтом Google на разных языках, но технически это разные сайты. Объединение нескольких сайтов под одним доменом характерно для бесплатных хостингов. Иногда для идентификации сайтов в адресе после указания хоста стоит тильда и имя сайта: *example.com/~my-site-name/*, иногда используется доменное имя третьего уровня: *my-site-name.example.com*.

Аппаратные сервера для хранения веб-сайтов называются веб-серверами. Сама услуга хранения называется веб-хостингом. Изначально каждый сайт хранился на своём собственном сервере, но с ростом Интернета технологическим улучшением серверов на одном компьютере стало возможно размещение множества сайтов (виртуальный хостинг). Сейчас сервера для хранения только одного сайта называются выделенными. Один и тот же сайт может быть доступен по разным адресам и хранится на разных серверах. Копия оригинального сайта в таком случае называется зеркалом. Существует так же понятие оффлайновая версия сайта - это копия сайта, которая может быть просмотрена на любом компьютере без подключения к компьютерной сети и использования серверного программного обеспечения.

Веб-сайты классифицируются:

- по доступности сервисов: открытые - все сервисы полностью доступны для любых посетителей, полуоткрытые - для доступа необходимо зарегистрироваться (обычно бесплатно), закрытые – полностью закрытые служебные сайты организаций (в том числе корпоративные сайты), личные сайты частных лиц (такие сайты доступны для узкого круга людей, доступ новым людям даётся через приглашения);

- по природе содержимого (статические - всё содержимое заранее подготавливается, пользователю выдаются файлы в том виде, в котором они хранятся на сервере); динамические - содержимое генерируется специальными скриптами (программами) на основе других данных из любого источника;

- по физическому расположению (внешние сайты сети Интернет, локальные сайты - доступны только в пределах локальной сети (корпоративные сайты организаций; сайты частных лиц в локальной сети провайдера);

- по схеме представления информации, её объёму и категории решаемых задач.

Классификация средств поиска в сети Интернет:

1. Поисковые машины (search engines).

Поисковая машина состоит из баз данных (БД) и программного обеспечения (ПО) для их обработки. ПО, в свою очередь, делится на две части: анализатор содержимого Web-страниц (Spider — спайдер, или «паук») и классификатор — определитель рейтинга, осуществляющий сортировку ссылок, найденных по запросу пользователя.

2. Порталы.

Поисковые машины с интегрированным предложением собственного информационного контента, которые могут предоставлять дополнительные услуги: бесплатные адреса электронной почты, места для размещения домашних Web-страниц и другие.

Интернет-портал (portal «главный вход; ворота») – веб-сайт, предоставляющий пользователю Интернета различные интерактивные сервисы, работающие в рамках одного веб-сайта, такие как почта, поиск, погода, новости, форумы, обсуждения, голосования. Сегодня порталами часто

называют себя просто большие сайты с разветвленной внутренней структурой и большим количеством ссылок. Однако если большая часть этих ссылок – внутренние или отправляющие пользователя на другую страницу этого же сайта, то называть такой сайт Интернет-порталом неправомерно.

3. Метапоисковые системы. Надстройки над поисковыми машинами, которые не имеют собственной БД и при поиске по запросу пользователя формируют, в свою очередь, запросы для нескольких внешних поисковых машин, а затем анализируют полученные результаты и выдают список ссылок в порядке, определяемом соотношением рейтингов ответа сразу по нескольким поисковым машинам. Следовательно, машина ведет опрос нескольких поисковых систем, а затем отбирает ссылки, следуя собственному алгоритму.

4. Каталоги (directories). Тщательно подобранные (чаще вручную) ссылки на ресурсы, сопровождаемые описаниями. Некоторые каталоги содержат внутренние поисковые машины.

*Поиск по ключевым словам.* Поиск документа в базе данных поисковой системы осуществляется с помощью введения запросов в поле поиска. Запрос должен содержать одно или несколько ключевых слов, которые являются главными для этого документа.

Релевантность поиска — это степень соответствия ответа запросу пользователя. Охват поиска напрямую зависит от объема базы данных той или иной поисковой машины. Глубина поиска — это ограничение на количество страниц или же на глубину вложенности папок на одном сервере.

*Современные подходы к обработке и хранению информации в сети Интернет*

Big Data — это термин, которым описывают наборы данных большого объема, быстро растущие с течением времени, и инструменты для работы с ними.

В общем виде технология Big Data должна выполнять следующие функции:

- «очистение» массива данных от дополнительной (бесполезной) информации;
- обработка и структурирование массива данных;
- анализ массива данных;
- защита данных;
- обеспечение доступа ко всему объему постоянно изменяемых данных.

Из всех вышеперечисленных функций приоритетное значение имеет анализ постоянно обновляемых данных. В современных условиях результаты такого анализа будут иметь решающее значение для компаний и предприятий при создании новых персонализированных товаров и услуг, позволят спрогнозировать дальнейшее направление развития. Принцип работы анализа Big Data отличается от традиционных концепций систем хранения данных или бизнес-аналитики. Анализ «больших данных» является по своей сути новым подходом к информационному менеджменту: создание принципиально новой



комплексной структуры (архитектуры) анализа. Она предполагает распределение функций сбора, хранения и анализа данных между несколькими программами-исполнителями, функционирующими в соответствии с алгоритмами, заложенными контрольными модулями.

В настоящее время существует множество проектов Big Data, которые отличаются разными моделями, структурой, особенностями анализа, разнообразными программными комплексами и техническими требованиями.

В научной литературе принято определять Big Data по трем «V»:

1. «Volume» - «объем».

Обычные инструменты хранения и анализа не способны справляться с таким объемом данных.

2. «Velocity» - «скорость».

Объемы данных поступают в обработку в режиме реального времени, в отличие от традиционной обработки пакета данных. Это означает, что они накапливаются моментально, при этом не имеет значения продолжительность потока самих данных. Big Data не только фиксирует потоки данных, но и производит их запись и обработку в таком виде, чтобы не было потерь.

3. «Variety» - «разнообразие». Big Data формируется из различных источников и в виде множества разнообразных форматов данных (видеоданные, фотографии, звуковые записи, текстовые сообщения, файлы транзакций, комментарии, использование ссылок и фиксация просмотров страниц).

Big Data не относится исключительно к «большим данным» в понимании объема. Он значительно шире, поскольку включает в себя скорости поступления данных и разнообразие источников и форматов получаемой информации.

Data mining, также называемый *Обнаружение знаний в базе данных (KDD)*, представляет собой метод, часто используемый для анализа больших массивов данных с помощью статистических и математических методов для поиска скрытых закономерностей или тенденций и извлечения из них ценности.

Автоматизируя процесс, инструменты data mining могут просматривать базы данных и эффективно выявлять скрытые закономерности. Для предприятий data mining часто используется для выявления закономерностей и взаимосвязей в данных, помогающих принимать оптимальные решения в бизнесе.

Data mining — это целостный процесс сбора, отбора, очистки, преобразования и извлечения данных для оценки закономерностей и, в конечном итоге, для извлечения ценности.

Главной функцией программного обеспечения для анализа данных является выполнение трудоемкой работы и автоматизации процесса преобразования данных в аналитическую информацию. Программное обеспечение для обработки данных:

- предоставляет необходимые инструменты для проведения качественного и количественного анализа;
- применяет статистические и аналитические возможности для принятия решений;
- обрабатывает и преобразует информацию для анализа корреляций между наборами данных;
- визуализирует наборы данных и результаты анализа.

Аналитическая платформа - это специализированное программное решение (или набор решений), которое содержит в себе все инструменты для осуществления качественной обработки данных.

Примеры программных средств для анализа данных:

- IBM SPSS Statistics – статистический пакет;
- IBM SPSS Modeler – платформа Data mining;
- SAS Base – статистический пакет;
- SAS Enterprise Miner – платформа Data mining;
- Stata – статистический пакет;
- Statistica Data Miner – платформа Data mining;
- Matlab – пакет анализа, математического моделирования;
- R, Python – языки программирования с развитыми средствами и библиотекам анализа данных.

Популярным программным инструментом анализа данных является MS Excel, который предполагается самой конструкцией табличного процессора и разнообразием встроенных инструментальных средств.

### ***Тема 2.3. Основы бизнеса в Internet***

*Электронный бизнес* — форма ведения бизнеса, при которой значительная его часть выполняется с применением информационных технологий (локальные и глобальные сети, специализированное программное обеспечение).

Электронный бизнес включает: продажи, маркетинг, финансовый анализ, платежи, поиск сотрудников, поддержку пользователей и партнерских отношений.

Части электронного бизнеса, связанные с Интернетом, включают в себя интернет-провайдинг, хостинг, контент-провайдинг и сервис-провайдинг, поставку технических и программных средств, web-дизайн, программирование и сопутствующие услуги, интернет-рекламу, интернет-магазины, интернет-аукционы, интернет-расчеты, интернет-маркетинг, интернет-коммерцию, информационную подписку и СМИ.

[Цифровые услуги: понятийный аппарат, классификаторы](#)

[Исследование аудитории Internet](#)

*Организация обратной связи в Интернете*



Можно выделить 4 разновидности формы обратной связи:

- 1) Онлайн-звонок
- 2) Онлайн-консультант
- 3) Формы заказа обратного звонка и онлайн-чаты
- 4) Callback-виджеты, которые представляют собой всплывающее окно с

одним полем для ввода номера телефона.

Web-сайты иначе называют интернет-представительством человека или организации. Веб-представительство - это, иными словами, визитная карточка фирмы. Обычно оно содержит всю необходимую публичную информацию, контакты, новости компании, отражает ее идеи, корпоративный стиль, представительность, все знаковые черты организации, номенклатурный каталог с фотографиями и (-или) перечень услуг.

Обычно при создании сайта преследуются следующие цели:

1. Представительская. Создать простую визитную карточку, разместив все реквизиты, контактную информацию, немного иллюстраций, общую характеристику фирмы или предприятия.

2. Имиджевая. Сформировать имидж предприятия как современного, динамично развивающегося, с высоким потенциалом, использующего в своей деятельности передовые информационные технологии.

3. Информационная. Размещение полной информации о деятельности фирмы, ее потенциале, программах, товарах и услугах, характеристиках товаров, сопутствующей информации по товарам и услугам в помощь посетителям сайта и заказчикам. Максимальное размещение информации для уменьшения нагрузки на другие каналы связи и работников.

4. Маркетинговая. Организовать привлечение новых потенциальных заказчиков, оформление на сайте предварительных заказов, изучение рынка (популярность товаров, спрос), другие маркетинговые задачи.

5. Рекламная.

6. Корпоративно-управленческая. Решение некоторых задач управления: информационный центр для региональных (удаленных) партнеров, оформление заявок на сайте, двусторонний обмен информацией между центром и партнерами (филиалами), привлечение новых партнеров.

[Электронная коммерция: понятие и особенности электронной коммерции, категории электронной коммерции, проблемы и прогнозы развития электронной коммерции](#)

Применение интернет-технологий в торговой деятельности позволяет существенно усилить позиции предприятия, сделать его более конкурентоспособным. Самое простое приложение интернет-технологий - интернет-магазин. Как правило, при создании интернет-магазина реализуются функции поиска и заказа товаров, проводится интеграция с платежными

системами. Торговые интернет-системы расширяют понятие интернет-магазина. Необходимо выделить следующие составляющие торговой интернет-системы: модель системы, структура системы, форма организации взаимодействия с пользователями, форма организации взаимодействия с поставщиками, инструменты разработки и поддержки.

В соответствии с выделенными составляющими можно предложить ряд классификаций торговых интернет-систем.

1. Классификация торговых интернет-систем по используемой бизнес-модели:
  - 1.1. весь функционал торговой интернет-системы работает с применением сети Интернет;
  - 1.2. функционально торговая система работает как на основе традиционной модели, так и с применением сети Интернет.
2. Классификация торговой системы по структуре:
  - 2.1. системы класса бизнес-потребитель (B2C - business-to-customer);
  - 2.2. системы класса бизнес-бизнес (B2B - business-to-business).
3. Классификация торговой системы по форме взаимодействия с пользователями:
  - 3.1. представлена в форме интернет-магазина;
  - 3.2. является веб-витриной;
  - 3.3. система выполнена как торговая площадка;
  - 3.4. система представляет собой контентный проект, например, система интернет-заказов.
4. Классификация торговой системы по форме взаимодействия с поставщиками:
  - 4.1. торговая интернет-система не имеет складов и реальных торговых запасов.
  - 4.2. система имеет склады товаров.

Выделим базовые функции торговой интернет-системы:

1. размещение каталога товаров (услуг) в сети Интернет;
2. обработка запросов пользователей (поиск товара, выборка по товарам, формирование заказа, просмотр состояния заказа, отчет по выполненным заказам);
3. поддержка оплаты товара (ведение истории покупок авторизованных пользователей системы дает возможность индивидуального подхода к пользователям):
  - 3.1. оплата через интегрированные в систему электронные платежные системы;
  - 3.2. оплата через банк, для чего система предоставляет необходимые для организации денежного перевода реквизиты;
4. косвенная и прямая реклама;
5. маркетинговые исследования.

Отметим, что процессы хранения, оплаты и доставки товара могут быть исключены из бизнес-модели интернет-магазина.

Internet - аукционы (онлайн-аукцион) – это аукцион, в котором можно принимать участие дистанционно, делая ставки через сайт или компьютерную программу аукциона. Цены на товары, которые продаются на онлайн-аукционе, устанавливаются во время открытых торгов на основании спроса участников.

В отличие от досок объявлений, торговых площадок и интернет-магазинов, цену на аукционе устанавливает покупатель, а не продавец. Продавец устанавливает правила, по которым проводится интернет-аукцион.

Существует 4 основных вида интернет-аукционов: прямой (английский), обратный, голландский (оптовый) и янки (своей цены).

Прямой (английский) аукцион – самый простой и самый распространенный, где купить и продать можно практически любую вещь или услугу, которые не запрещены законодательством. Отличительные особенности данного вида аукциона – гласные торги и поднятие цены. Аукцион начинается с минимальной цены, затем покупатели по очереди выставляют более высокие цены. Заканчиваются торги тогда, когда истекает срок аукциона, установленный продавцом (обычно 1-14 дней). Торги на таком интернет-аукционе не всегда заканчиваются продажей. Если продавец установил резервную цену (минимальную цену, за которую он готов продать), то продажа будет только в том случае, если эта резервная цена достигнута. Товар достается покупателю, который дал за товар максимальную цену.

Обратный онлайн-аукцион является противоположностью прямому – покупатели объявляют о том, что они хотят приобрести, а продавцы в ответ выставляют свои предложения. Конкурируют теперь между собой не покупатели, а продавцы, и цены уменьшаются.

Голландский (оптовый) аукцион – аукцион, на котором продавец может выставлять много единиц одного товара одновременно. Покупатели могут претендовать на покупку многих единиц товара, выигравшие покупатели платят только минимальную из выигравших цен.

Аукцион янки называется еще дискриминационным. Торги на нем закрытые от других участников, победитель, давший наивысшую цену, получает товар за цену, которую он назвал. Обычно каждый участник такого аукциона подает только одну заявку, поэтому к нему нужно подготовиться. После открытия заявок определяется победитель. Если выставлено несколько единиц товара, то поданные заявки сортируются по убыванию предложенной цены, выигравшие платят ту сумму, которую они сами предложили.

Существуют еще приватные интернет-аукционы (VIP аукционы), в которых участвуют избранные покупатели со специальной аккредитацией. Участник может сделать только 1 ставку в течение строго ограниченного времени, он не знает размер и количество ставок других участников.

К финансовым услугам, предоставляемым через Интернет, относятся следующие виды деятельности:

- интернет-банкинг - предоставление банковских услуг через Интернет;
- интернет-трейдинг - услуги по оперированию на валютном и фондовом рынках через Интернет;
- интернет-страхование - предоставление услуг страхования через Интернет.

Платежные системы Internet или Электронные платежные системы отличаются от платежных систем банковских карт тем, что все транзакции осуществляются только через интернет. К крупным международным электронным платежным системам можно отнести OkPay, Perfect Money, PayPal, Payoneer, Google Wallet, e-gold, ePayService, Moneybookers, Liberty Reserve, Qiwi. В Республике Беларусь онлайн-платежи можно подключить через следующие сервисы:

EasyPay — первая официальная белорусская электронная платежная система, которая была запущена в 2004 году и являлась совместной разработкой ООО «Открытый контакт» с банком ОАО «Белгазпромбанк». С 2017 года в связи с изменением законодательства перестала работать с физическими лицами.

WebMoney — это одна из первых российских систем для приема интернет-платежей, работающая с начала 1998 года. В Республике Беларусь гарантом системы WebMoney и в частности WMB (электронный эквивалент белорусских рублей BYN) выступает Технобанк, который сопровождает и оказывает полный комплекс услуг;

iPay — белорусская мобильная платежная система от компании ООО «АЙ ПЭЙ», реализованная совместно с ОАО «Паритетбанк», сотовым оператором связи СООО «Мобильные ТелеСистемы» и разработчиком программного обеспечения ЗАО «БиСмарт» в 2009 году. Помимо стандартных функций, позволяет оплачивать услуги с помощью средств, находящихся на балансе телефонов белорусских мобильных операторов.

Яндекс Деньги (сейчас ЮMoney) — это платёжная система от компании «Yandex», пользоваться которой могут граждане Беларуси, совершая платежи в белорусских рублях в рамках сервиса.



Процесс подключения платежей на сайт состоит из 3 этапов:

- выбор специальных программ/сервисов (платежный шлюз или агрегатор), через которые будут приниматься платежи;
- юридическое оформление и редактирование сайта согласно требованиям платежных сервисов;
- техническая реализация и настройка интеграции.

Платежный шлюз — это отдельный сервис, основными задачами которой является авторизация и обработка платежей по кредитным и дебетовым карточкам для онлайн и оффлайн торговли. Шлюз позволяет безопасно

проводить финансовые операции с шифрованием конфиденциальных данных между торговой площадкой (сайт, мобильное приложение) и процессором платежей.

Наиболее известными платежными шлюзами в мире являются Amazon Payments, Authorize.net, 2Checkout, Dwolla. Принцип работы у них одинаковый, отличия заключаются в совместимости с платформами, возможными методами и валютами, используемыми при оплате, комиссиями за проведение транзакций.

Платежный агрегатор — аппаратно-программный сервис, главной задачей которого является максимальная автоматизация и контроль процедуры приема электронных платежей. Как и платежные шлюзы, они занимаются интеграцией различных способов оплаты и проведением платежа с максимальной безопасностью. Характерное отличие в том, что агрегаторы собирают у себя денежные средства их клиентов, а шлюзы только предоставляют услуги маршрутизации платежей. Агрегаторы платежей имеют статус небанковской кредитной организации, а шлюзы — это просто сервисы-посредники, которые не взаимодействуют напрямую с оплатами, а только помогают провести их через сайт, интернет-магазин или приложение.

Из наиболее известных платежных агрегаторов в мире можно отметить Wallet One, PayOnline, Яндекс.Касса (ЮKassa), ROBOKASSA, Stripe, PayPal, Payoneer.



*Известные белорусские платежные агрегаторы:*

- 1) самый распространенный платежный провайдер Единое расчетное информационное пространство (АИС «ЕРИП»);
- 2) WebPay — платежное решение для приема онлайн-платежей через сайт по международным/белорусским банковским карточкам, а также через АИС «Расчёт» ЕРИП;
- 3) bePaid;
- 4) ArtPay;



- 5) EasyPay — электронная платежная система и агрегатор, работает с сервисом E-POS;
- 6) «Хуткі Грош» — сервис, который специализируется только на работе с ЕРИП;
- 7) Assist BELARUS — дочерняя компания российского агрегатора Assist, позволяет принимать оплату карточками на сайтах и интернет-магазинах. Имеется стандартный механизм подключения к белорусской системе ЕРИП;
- 8) Expresspay — как и предыдущие агрегаторы позволяет подключать онлайн платежи на веб-сайте и принимать их посредством банковских карт, так и через ЕРИП;
- 9) Яндекс.Касса (ЮКassa) — российский сервис по приёму платежей через интернет, работающий как для компаний и индивидуальных предпринимателей, социальных и некоммерческих фирм, организаций. Работает официально в Беларуси с 2011 года, в том числе с белорусскими рублями.

#### ***Тема 2.4. Вопросы безопасности информации в Internet***

Согласно закону Республики Беларусь от 10 ноября 2008 г. № 455-З «Об информации, информатизации и защите информации» применяются следующие основные термины и их определения:

- база данных — совокупность структурированной и взаимосвязанной информации, организованной по определенным правилам на материальных носителях;
- банк данных — организационно-техническая система, включающая одну или несколько баз данных и систему управления ими;
- владелец программно-технических средств, информационных ресурсов, информационных систем и информационных сетей — субъект информационных отношений, реализующий права владения, пользования и распоряжения программно-техническими средствами, информационными ресурсами, информационными системами и информационными сетями в пределах и порядке, определенных их собственником в соответствии с законодательством;
- государственная информационная система — информационная система, создаваемая и (или) приобретаемая за счет средств республиканского или местных бюджетов, государственных внебюджетных фондов, а также средств государственных юридических лиц;
- государственный информационный ресурс — информационный ресурс, формируемый или приобретаемый за счет средств республиканского или местных бюджетов, государственных внебюджетных фондов, а также средств государственных юридических лиц;



- доступ к информации – возможность получения информации и пользования ею;
- доступ к информационной системе и (или) информационной сети – возможность использования информационной системы и (или) информационной сети;
- защита информации – комплекс правовых, организационных и технических мер, направленных на обеспечение конфиденциальности, целостности, подлинности, доступности и сохранности информации;
- информационная сеть – совокупность информационных систем либо комплексов программно-технических средств информационной системы, взаимодействующих посредством сетей электросвязи;
- информационная система – совокупность банков данных, информационных технологий и комплекса (комплексов) программно-технических средств;
- информационная технология – совокупность процессов, методов осуществления поиска, получения, передачи, сбора, обработки, накопления, хранения, распространения и (или) предоставления информации, а также пользования информацией и защиты информации;
- информационная услуга – деятельность по осуществлению поиска, получения, передачи, сбора, обработки, накопления, хранения, распространения и (или) предоставления информации, а также защиты информации;
- информационный ресурс – организованная совокупность документированной информации, включающая базы данных, другие совокупности взаимосвязанной информации в информационных системах;
- конфиденциальность информации – требование не допускать распространения и (или) предоставления информации без согласия ее обладателя или иного основания, предусмотренного законодательными актами;
- обладатель информации – субъект информационных отношений, получивший права обладателя информации по основаниям, установленным актами законодательства, или по договору;
- оператор информационной системы – субъект информационных отношений, осуществляющий эксплуатацию информационной системы и (или) оказывающий посредством ее информационные услуги;
- персональные данные – основные и дополнительные персональные данные физического лица, подлежащие в соответствии с законодательными актами внесению в регистр населения, а также иные данные, позволяющие идентифицировать такое лицо;
- пользователь информации – субъект информационных отношений, получающий, распространяющий и (или) предоставляющий информацию, реализующий право на пользование ею;

- пользователь информационной системы и (или) информационной сети – субъект информационных отношений, получивший доступ к информационной системе и (или) информационной сети и пользующийся ими;
- предоставление информации – действия, направленные на ознакомление с информацией определенного круга лиц;
- распространение информации – действия, направленные на ознакомление с информацией неопределенного круга лиц;
- собственник программно-технических средств, информационных ресурсов, информационных систем и информационных сетей – субъект информационных отношений, реализующий права владения, пользования и распоряжения программно-техническими средствами, информационными ресурсами, информационными системами и информационными сетями.

Информационная безопасность - защищенность информационной системы от случайного или преднамеренного вмешательства, наносящего ущерб владельцам или пользователям информации.

Проблемам международной информационной безопасности уделяется достаточно пристальное внимание со стороны многих международных организаций, включая ООН, Организацию Договора о коллективной безопасности (далее – ОДКБ), НАТО, Шанхайскую организацию сотрудничества. Координацию такого взаимодействия осуществляют как органы общей компетенции (Генеральная Ассамблея ООН, Североатлантический совет НАТО, Совет коллективной безопасности ОДКБ), органы специальной компетенции, для которых мандат в данной сфере является дополнительным (Межамериканский комитет против терроризма), специально созданные структурные единицы (Консультационный координационный центр ОДКБ по вопросам реагирования на компьютерные инциденты, Комитет по киберобороне НАТО, Управление НАТО в сфере киберобороны, Агентство по коммуникациям и информации НАТО и созданный в его рамках Центр по реагированию на киберугрозы, Агентство по сетевой и информационной безопасности Европейского союза, Европейский центр по киберпреступности).

Сетевой атакой называют намеренные действия третьих лиц, направленные на установление контроля над локальным или удаленным компьютером, вычислительной системой.

В результате атак злоумышленники могут нарушать работу сети, изменять права аккаунта, получать персональные данные пользователей и реализовывать другие цели.

Современная классификация угроз и основные виды сетевых атак:

1. По характеру воздействия на атакуемую сеть: активные и пассивные.

Активная атака проводится с непосредственным воздействием на сеть, и обычно предусматривает ограничение ее работоспособности, модификацию настроек.

Пассивная атака проводится без непосредственного влияния на работу сети. Однако в ее результате нарушается сетевая безопасность. Обнаружить пассивную атаку намного сложнее именно из-за отсутствия прямого воздействия.

2. В зависимости от цели различают виды сетевых атак, направленных на нарушение:
  - функционирования;
  - конфиденциальности;
  - целостности атакуемой сети.
3. Воздействие на атакуемую сеть может осуществляться на разных уровнях эталонной модели ISO/OSI. В том числе выделяются такие уровни:
  - физический;
  - сетевой;
  - транспортный;
  - канальный;
  - сеансовый;
  - прикладной;
  - представительный.

В связи с быстрым развитием информационных технологий и технических средств статичные механизмы защиты от сетевых угроз часто оказываются неэффективными. Обеспечить эффективную защиту информации позволяют динамические методы, способные оперативно выявлять и устранять угрозы. Работа динамических технологий строится на оценки уровня подозрительности действий в сети со стороны определенной службы или процесса.

Для защиты сетей от внешних угроз могут применяться следующие основные методы и технологии:

- применение портов высокой надежности, шифрование данных;
- использование эффективных антивирусов и сканеров;
- применение программного или аппаратного сетевого экрана;
- установка блокираторов руткитов (вредоносного программного обеспечения, активно использующего технологии маскировки, чтобы скрыть свое присутствие в зараженной системе) и снифферов (программное обеспечение, анализирующее входящий и исходящий трафик с компьютера, подключенного к интернету).

Разнообразие видов сетевых атак, которым могут подвергаться корпоративные и частные сети, требует выработки эффективных мер по их защите.

### [Основные аспекты профилактики киберпреступности в Республике Беларусь](#)

Базовые основы кибербезопасности для пользователей:

1. Сложные пароли.

Не рекомендуется хранить информацию о паролях на ПК, используемого для работы в сети. Обязательно нужно использовать двухэтапную аутентификацию, это обеспечит повышенную защиту аккаунта.

Нельзя для пароля использовать данные, которые преступники могут найти сами. Пароли должны быть сложными, содержать специальные значки, цифры. Запрещается использовать один пароль для разных сайтов. Они должны быть разными и менять их следует каждые несколько недель.

### 2. Проверка политики конфиденциальности.

Перед тем, как устанавливать расширения на браузеры, приложения, регистрироваться в них, пользователь обязан изучать политику конфиденциальности. Нужно убедиться в том, что приложения или расширения не имеют права использовать личные данные.

### 3. Настройки браузеров.

Пользователь не должен давать браузеру разрешение автоматически запоминать пароли к сайтам. Рекомендуется отключать опцию в настройках. Должна быть отключена синхронизация браузеров на ПК и в смартфонах.



### 4. Очистка cookies.

Эксперты рекомендуют регулярно очищать временные файлы, которые содержат сведения о сайтах, посещаемых пользователем. Так минимизируется риск взлома и доступа к личным данным.



### 5. Общие Wi-Fi сети.

Не рекомендуется использовать открытые общественные сети, поскольку через них можно быстро взломать ПК или гаджет.

*Таблица 2.1 Средства обеспечения сетевой безопасности*

 <p><i>Межсетевые экраны</i></p>	<p>Межсетевые экраны создают барьер между надежной внутренней сетью и ненадежными внешними сетями, такими как Интернет. Они используют набор определенных правил, которые разрешают или блокируют трафик. Межсетевой экран может быть аппаратным, программным или смешанного типа</p>
 <p><i>Защита электронной почты</i></p>	<p>Шлюзы электронной почты являются первоочередной целью для злоумышленников, которые хотят проникнуть в сеть. Хакеры анализируют личную информацию и прибегают к методам социальной инженерии для организации изощренных фишинговых кампаний, чтобы обмануть получателей и направить их на сайты с вредоносным ПО. Приложение для защиты электронной почты блокирует входящие атаки и контролирует исходящие сообщения для предотвращения потери конфиденциальных данных</p>

 <p>Программы для защиты от вредоносного ПО</p>	<p>На компьютере устанавливается программное обеспечение, которое сканирует его на наличие вредоносного ПО входящий трафик, и впоследствии непрерывно отслеживают файлы с целью поиска аномалий, удаления вредоносного ПО и устранения ущерба</p>
 <p>Сегментация сети</p>	<p>Программно-определяемая сегментация классифицирует сетевой трафик и упрощает применение политик безопасности. В идеале классификация должна быть основана на идентификации конечных устройств, а не просто на IP-адресах. Можно назначать права доступа на основе ролей, расположения и других факторов, чтобы предоставлять необходимый уровень доступа нужным пользователям, изолировать подозрительные устройства и устранять угрозы</p>
 <p>Управление доступом</p>	<p>Для защиты от потенциальных хакеров необходимо распознавать каждого пользователя и каждое устройство. Только в этом случае можно обеспечить соблюдение политик безопасности. Можно блокировать конечные устройства, не соответствующие установленным требованиям, или предоставлять им только ограниченный доступ. Этот процесс называется контролем доступа к сети (NAC)</p>
 <p>Безопасность приложений</p>	<p>Любое программное обеспечение, используемое в организации, необходимо защищать, независимо от того, создается оно ИТ-специалистами организации или приобретается на стороне. К сожалению, любое приложение может содержать уязвимости, которые хакеры могут использовать для проникновения в сеть</p>
 <p>Поведенческая аналитика</p>	<p>Для обнаружения аномального поведения сети необходимо знать, как выглядит нормальное. Средства поведенческой аналитики автоматически выявляют действия, отличающиеся от нормы. Это помогает отделу информационной безопасности эффективнее выявлять признаки компрометации, которые могут создавать проблемы, и быстро блокировать угрозы</p>

 <p><i>Предотвращение утечки данных</i></p>	<p>Организации должны быть уверены, что их сотрудники не передают конфиденциальную информацию за пределы сети. Технологии предотвращения утечки данных (DLP) могут препятствовать небезопасной загрузке, переадресации или даже печати важной информации пользователями</p>
 <p><i>Управление событиями и данными об информационной безопасности (SIEM)</i></p>	<p>Продукты SIEM группируют сведения, необходимые сотрудникам, занимающимся ИТ-безопасностью, чтобы выявлять угрозы и реагировать на них. Они могут иметь форму физических и виртуальных устройств и серверного ПО</p>
 <p><i>Сеть VPN</i></p>	<p>Виртуальная частная сеть шифрует подключение конечных устройств к сети, часто выполняемое через Интернет. Как правило, сети VPN для удаленного доступа используют протокол IPsec для аутентификации взаимодействия между устройством и сетью</p>
 <p><i>Безопасность мобильных устройств</i></p>	<p>Целью атак киберпреступников все чаще становятся мобильные устройства и приложения. В течение ближайших 3 лет 90 % ИТ-организаций могут организовать поддержку корпоративных приложений на личных мобильных устройствах. При этом необходимо контролировать доступ различных устройств к сети, требуется настраивать их подключения, чтобы сохранить конфиденциальность сетевого трафика</p>
 <p><i>Безопасность web-трафика</i></p>	<p>Решение для обеспечения безопасности веб-трафика контролирует использование Интернета сотрудниками, блокирует веб-угрозы и запрещает доступ к вредоносным веб-сайтам. Оно защищает веб-шлюз в локальной сети или в облаке</p>

 <p><i>Безопасность беспроводного доступа</i></p>	<p>Беспроводные сети менее безопасны, чем проводные. Если не соблюдать строгие меры безопасности, развертывание беспроводной локальной сети не отличается от бесконтрольного размещения Ethernet-портов в свободном доступе. Чтобы предотвратить появление эксплоитов (эксплоит - компьютерная программа, фрагмент программного кода или последовательность команд, использующие уязвимости в программном обеспечении и применяемые для проведения атаки на вычислительную систему), необходимы продукты, специально предназначенные для защиты беспроводных сетей</p>
 <p><i>Системы предотвращения вторжений (IPS) и системы обнаружения атак (IDS)</i></p>	<p>Система IPS сканирует сетевой трафик и активно блокирует атаки. С этой целью устройства сопоставляют огромные объемы информации, полученной с помощью глобальной аналитики угроз, что позволяет не только блокировать вредоносную активность, но и отслеживать путь подозрительных файлов и вредоносного ПО во всей сети для предотвращения распространения эпидемий и повторного заражения. Отличие IDS и IPS заключается лишь в дальнейших действиях, которые выполняет система после выявления угрозы: IDS может только сигнализировать о факте обнаружения, IPS — разрывать соединение, в котором была обнаружена атака.</p>

Network Address Translation (NAT) — это механизм в сетях TCP/IP, позволяющий изменять IP адрес в заголовке пакета, проходящего через устройство маршрутизации трафика.

Прокси-сервер — это промежуточный сервер между пользователем Интернета и серверами, откуда запрашивается информация, выполняющий функцию посредника или шлюза, фильтра, который стоит между человеком и огромными (и не всегда безопасными) данными в сети.

[Криптографическая защита информации \(симметричные и асимметричные алгоритмы шифрования\)](#)

Стеганография – наука о методах преобразования информации в вид, при котором факт её существования становится секретом.

[Электронная цифровая подпись](#)



## Тема 2.5. Общие сведения о проектировании web-узлов. Язык HTML

**Web-узел** — это набор файлов в формате HTML, расположенных в определенной папке и связанных друг с другом гиперссылками. Один из файлов web-узла назначается главным, он представляет собой домашнюю страницу и открывается в браузере пользователя при подключении к web-узлу. Остальные web-страницы выводятся в окно браузера по мере перехода к ним по гиперссылкам. Кроме файлов HTML в состав узла входит набор графических объектов формата GIF или JPG, предназначенных для оформления страниц.



Рис. 1.3. Этапы разработки web-узла или web-сайта



### Разработка Web-документов средствами программ пакета MS Office

Любой документ MS Office можно:

1) сохранить в виде веб-страницы

Варианты сохранения документа как веб-страницы: «Веб-страница в одном файле (\*.mht; \*.mhtml)»; «Веб-страница (\*.htm; \*.html)»; и «Веб-страница с фильтром (\*.htm; \*.html)».

2) создать на основе встроенного шаблона веб-страницы.

Для качественной разработки web-сайта используются различные инструменты, при этом все его веб-страницы будут содержать разметку, написанную на языке разметки гипертекста, что в переводе на английский язык означает «Hypertext Markup Language» или сокращённо «HTML».

```
<html>
<title>
<head>
<meta>
<script>
<body>
<link><head>
```

HTML

HTML - это язык разметки (не является языком программирования), который представляет простые правила оформления и компактный набор структурных и семантических элементов разметки (тегов). HTML позволяет описывать способ



представления логических частей документа (заголовки, абзацы, списки) и создавать веб-страницы разной сложности. При этом веб-страницы могут содержать встроенные или загружаемые программы на скриптовых языках (в первую очередь Javascript) и программы-апплеты на языке Java.

Тег (html-тег, тег разметки) — управляющая символьная последовательность, которая задает способ отображения гипертекстовой информации. HTML-тег состоит из имени, за которым может следовать необязательный список атрибутов. Весь тег (вместе с атрибутами) заключается в угловые скобки  $\langle \rangle$ :

`<имя_тега [атрибуты]>`

Как правило, теги являются парными и состоят из начального и конечного тегов, между которыми и помещается информация. Имя конечного тега совпадает с именем начального, но перед именем конечного тега ставится косая черта / (`<html>...</html>`). Конечные теги никогда не содержат атрибутов. Некоторые теги не имеют конечного элемента, например, тег `<img>`. Регистр символов для тегов не имеет значения.

Примеры часто используемых тегов HTML:

`<html>...</html>` — контейнер гипертекста

`<head>...</head>` — контейнер заголовка документа

`<title>...</title>` — название документа (то, что отображается в заголовке окна браузера)

`<body>...</body>` — контейнер тела документа

`<div>...</div>` — контейнер общего назначения (структурный блок)

`<hN>...</hN>` — заголовок N-ого уровня (N = 1...6)

`<p>...</p>` — основной текст

`<a>...</a>` — гиперссылка

`<ol>...</ol>` — нумерованный список

`<ul>...</ul>` — маркированный список

`<li>...</li>` — элемент списка

`<table>...</table>` — контейнер таблицы

`<tr>...</tr>` — строка таблицы

`<td>...</td>` — ячейка таблицы

`<img>` — изображение

`<form>...</form>` — форма

`<i>...</i>` — отображение текста курсивом

`<b>...</b>` — отображение текста полужирным шрифтом

`<em>...</em>` — выделение (курсивом)

`<strong>...</strong>` — усиление (полужирным шрифтом)

`<br>` — принудительный разрыв строки

[Язык HTML: структура HTML-документа, гиперссылки, закладки, тэги оформления текста, графика, создание таблиц, списки, создание и использование форм, фреймы, разделенные рисунки \(карты-изображения\), стили CSS](#)

Программы разработки интерактивных Web-документов позволяют в процессе создания визуально наблюдать добавление новых компонентов к формам документа. Такие средства программирования принято называть визуальными интегрированными средами.

### ***Тема 2.6 Использование программ, выполняющихся в контексте Web-браузера, при разработке Web-документов. Динамическая генерация содержимого Web-документа***

Современный JavaScript – это «безопасный» язык программирования. Он не предоставляет низкоуровневый доступ к памяти или процессору, потому что изначально был создан для браузеров, не требующих этого.

В браузере для JavaScript доступно всё, что связано с манипулированием веб-страницами, взаимодействием с пользователем и веб-сервером. Например, JavaScript может:

- добавлять новый HTML-код на страницу, изменять существующее содержимое, модифицировать стили;
- реагировать на действия пользователя, щелчки мыши, перемещения указателя, нажатия клавиш;
- отправлять сетевые запросы на удалённые сервера, скачивать и загружать файлы;
- получать и устанавливать куки (небольшой фрагмент данных, отправленный веб-сервером и хранимый на компьютере пользователя), задавать вопросы посетителю, показывать сообщения;
- запоминать данные на стороне клиента («local storage»).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений и находит применение в браузерах как язык сценариев для придания интерактивности веб-страницам.

Благодаря независимости от платформы, программы Java могут работать на различных устройствах, особенно на встроенных устройствах, таких как принтеры, сканеры и факсы. При этом выполнение программы не ограничено типом предоставляемой среды операционной системы.

В Java-технологиях используются апплеты (англ. applet от application — приложение и -let — уменьшительный суффикс) — это не самостоятельные компоненты программного обеспечения, работающие в контексте другого, полновесного приложения, предназначенные для одной узкой задачи и не имеющие ценности в отрыве от базового приложения.

ActiveX - это специальная технология создания программ, основными компонентами которой являются управляющие элементы или небольшие программы, используемые на веб-страницах и позволяющие расширить ее функционал. Например, при помощи ActiveX в браузере работают различные кнопки и появляются диалоговые окна на веб-страницах, воспроизводится анимация, загружается плеер, открываются файлы других форматов прямо в

окне браузера. Многие компании используют элементы управления ActiveX для установки своих программ прямо со своего интернет-сайта на компьютер пользователя. Но это необычные программы, которые можно запустить двойным щелчком левой кнопки мышки по файлу. Эти программы запускаются и выполняются операционной системой.

Если при создании сайта использовалась технология ActiveX, то при посещении этого сайта браузер выведет запрос на установку элемента управления ActiveX. Если пользователь соглашается с запросом, то управляющий элемент загружается на компьютер и запускается на нём.

Поскольку управляющие компоненты ActiveX – это программы, которые запускаются через веб-браузер, то есть вполне реальная угроза установки вместе с файлом вредоносной программы или вируса.

Macromedia Flash является одним из инструментов для создания анимированных объектов (изображений, схем навигации, динамических web-узлов, проигрывателей, рекламных баннеров, видеороликов, музыкального видео) на основе векторной графики со встроенной поддержкой интерактивности.

ActionScript — объектно-ориентированный язык программирования, который добавляет интерактивность и обработку данных в содержимое Flash-приложений.

CGI-скриптом называют программу, написанную на любом языке программирования или командном языке, которая осуществляет обмен данными с HTTP-сервером.

SSI (Server Side Includes - «включения на стороне сервера») — язык для динамической «сборки» веб-страниц на сервере из отдельных составных частей и выдачи пользователю полученного HTML-документа.

SSI — серверные включения, созданные для выполнения задач, направленных на облегчение администрирования сайта. SSI директивы вставляются в HTML-код и служат для передачи указаний Web-серверу.

Технология Active Server Pages (ASP) — это серверная технология для разработки и выполнения динамических интерактивных Web-приложений. Средства ASP позволяют объединить возможности HTML-страниц, команд сценариев и компонентов COM (технологический стандарт Microsoft, предназначенный для создания программного обеспечения на основе взаимодействующих компонентов, каждый из которых может использоваться во многих программах одновременно) в интерактивных Web-страницах и мощных Web-приложениях, делают удобным и легким процесс их создания и изменения.

PHP (Hypertext PreProcessor, «препроцессор гипертекста») — скриптовый язык программирования, основная область применения — разработка скриптов командной строки и GUI-приложения (приложения с графическим интерфейсом пользователя), которые работают на стороне сервера.

## ПРАКТИЧЕСКИЙ РАЗДЕЛ

### Лабораторная работа № 1

#### Создание таблиц

**Цель работы:** Изучить структуру объекта MS ACCESS “таблица”; научиться создавать таблицы с помощью конструктора, задавать полям различные типы данных и создавать связи между таблицами.

#### Ход работы

Необходимо создать таблицы Автосклад, Агенты, Договора и связать их между собой. Для этого необходимо выполнить следующие действия:

1. Запустить программу MS–ACCESS, подтвердить операцию «новая база данных», выбрать папку для хранения базы данных (D:\Users или E:\Users) и задать ей имя (Автосалон).

2. Для того чтобы создать 1-ую таблицу «Автосклад» с помощью конструктора необходимо:

1. На панели инструментов вкладке *Объекты* выбрать вариант *Создание таблицы в режиме Конструктора*.
2. Создать структуру будущей таблицы «Автосклад».

Имя поля	Тип данных	Описание (необязательно)
Код авто	Счетчик	Код автомобиля
Марка авто	Короткий текст	Полное название Марки автомобиля
Цена	Денежный	Цена за один автомобиль
Остаток	Числовой	Остаток на складе


Рис. 1.1. Режим конструктора для таблицы «Автосклад»


Для этого в столбце *Имя поля* вписать *КодАвто*, в столбце *Тип данных* выбрать вариант *Счетчик*, в столбце *Описание* ввести *Код автомобиля*.

Для поля *МаркаАвто* выбрать тип поля – *Короткий текст* или *Текстовый*, в области *Свойства поля* на вкладке *Общие* установить размер поля – 30, обязательное поле – *да*, описание – *Полное название автомобиля*.

Для поля *Цена* выбрать тип поля – *денежный*, число десятичных знаков – 0, обязательное поле – *да*, описание – *Цена за 1 автомобиль*.

Для поля *Остаток* выбрать тип поля – *числовой*, размер поля - *длинное целое*, условие на значение – «>=0», описание – *Остаток на складе*.

3. Определить ключевое поле, для этого необходимо перейти в поле *КодАвто*, нажать кнопку **Ключевое поле**  на панели инструментов **Конструктор таблиц**.

4. Перейти в режим таблицы, для этого нажать кнопку **Режим**  на панели инструментов **Конструктор таблиц**. В информационном окне MS Access нажать кнопку **Да**. В диалоговом окне **Сохранение** в поле *Имя таблицы* ввести *АвтоСклад*.

5. Заполнить таблицу подготовленными данными, для этого их следует ввести в соответствующие поля (см. рис. 1.2).

	Код авто	Марка авто	Цена	Остаток
+	1	Мерседес	60 000Bp	5
+	3	Ауди	24 000Bp	10
+	4	Рено	28 000Bp	6
+	5	Фольксваген	10 000Bp	15
+	6	Жигули	50 000Bp	14
+	7	Лексус	100 000Bp	1
+	8	Форд	360 000Bp	8
*	(№)			


Рис. 1.2. Таблица «АвтоСклад»

Аналогично с помощью конструктора создать 2-ую таблицу базы данных **Агенты**, которая будет иметь поля *КодАгента*, *ФИО*, *Телефон*. Сделать поле *КодАгента* ключевым. Перейти в режим таблицы и сохранить таблицу под именем *Агенты*. Таблица «Агенты» имеет следующий вид (см. рис. 1.3).

Затем с помощью конструктора создать 3-ю таблицу **Договора** базы данных, которая содержит следующие поля: *НомерДоговора*, *Дата*, *КодАгента*, *КодАвто*, *Количество*. Сделать поле *НомерДоговора* ключевым. Таблица «Договора» имеет следующий вид (см. рис. 1.4).

#### Создание связи между всеми таблицами. Разработка схемы данных.

Создаем связи между таблицами. Для этого:


- щелкаем по кнопке  на панели инструментов или выполняем команду Работа с базами данных, Схема данных. На экране появится окно «Схема данных»;

	Код агента	ФИО	Телефон	Стаж работы
+	1	Семенова Светлана Валерьевна	5555213	1
+	2	Маринина Анна Николаевна	3185695	2
+	3	Сорока Анастасия Федоровна	6757645	3
+	4	Стрижц Владимир Владимирович	5053482	3
+	5	Семенова Татьяна Валерьевна	2763589	1
+	6	Чуевская Анжелика Сергеевна	3562145	1
+	7	Цвирко Андрей Эдуардович	5236955	2
+	8	Рубашка Юлия Владимировна	5756428	3
*	(№)			

Рис. 1.3. Таблица «Агенты»

	Номер догс	Дата	Код агента	Код авто	Количество
	2	30.10.2021	3	3	8
	3	05.11.2021	1	3	1
	4	08.11.2021	4	1	3
	5	12.11.2021	5	7	1
	6	15.11.2021	6	6	9
	7	03.01.2021	8	5	3
	8	10.01.2021	7	4	6
*	(№)				

Рис. 1.4. Таблица «Договора»

- щелкаем по кнопке  на панели инструментов или выполняем команду Связи, Добавить таблицу;
- в появившемся окне будет выделено название одной таблицы. Щелкаем по кнопке <Добавить>;
- переводим выделение на имя следующей таблицы и щелкаем по кнопке <Добавить>. Аналогично добавляем оставшиеся таблицы;
- закрываем окно, щелкнув по кнопке <Заккрыть>.

Создаем связь между таблицами *АвтоСклад* и *Договора*. Для этого подводим курсор мыши к полю «КодАвто» в таблице *АвтоСклад*, щелкаем левой кнопкой мыши и, не отпуская ее, перетащите курсор на поле «КодАвто» в таблице *Договора*, а затем отпускаем кнопку мыши. На экране откроется окно «Связи»:

- устанавливаем флажок ("галочку") в свойстве *Обеспечение целостности данных*, щелкнув по нему;
- устанавливаем флажок в свойстве *Каскадное обновление связанных полей* и *Каскадное удаление связанных записей*;

**Примечание.** Задание каскадного обновления связанных полей и каскадного удаления связанных записей позволяет нам отредактировать записи только в таблице *АвтоСклад*, а в таблице *Договора* эти действия будут со связанными записями выполняться автоматически.

- щелкаем по кнопке <Создать>. Связь будет создана;
- аналогично создаем связь между полем «Номер договора» в таблице *Договора* и полем «Номер договора» в таблице *Агенты*.

Схема данных будет иметь следующий вид (см. рис. 1.5).



Рис. 1.5. Схема данных

### КОНТРОЛЬНЫЕ ВОПРОСЫ:

1. Понятие реляционной базы данных.
2. Чем поле отличается от записи?
3. Какие основные функции СУБД и что это такое?
4. Для чего между таблицами создаются связи?
5. В чем заключается связь Один-ко-многим?

6. Чем отличается тип связи Многие-ко-многим от типа связи Один-к-одному?
7. Для чего данным задаются различные типы данных?

### **ЗАДАНИЕ**

Создать указанную базу данных в Ms Access согласно варианту задания. Сохранить ее в папке Users/ № группы/ \*.mdb. Заполнить таблицы (не менее 5-ти записей). Обеспечить связь между таблицами (целостность данных).

### **ВАРИАНТЫ ЗАДАНИЙ**

**Вариант №1.** Создать следующую базу данных «Поставка товара». Объекты предметной области следующие:

*Товары:*

- ✓ код товара
- ✓ тип товара
- ✓ цена товара
- ✓ срок хранения
- ✓ страна-производитель

В качестве ключевого поля будет использоваться запись «Код товара».

*Поставщики:*

- ✓ код поставщика
- ✓ название фирмы
- ✓ адрес (город)
- ✓ лицензия
- ✓ код товара
- ✓ сроки поставки (в днях)

Ключевым полем в данной таблице будет являться поле «Код поставщика».

*Магазины:*

- ✓ код магазина
- ✓ телефон
- ✓ дата заказа
- ✓ код товара
- ✓ код поставщика

Основной характеристикой, которая будет задаваться в качестве ключевого поля, является «Код магазина».

**Вариант №2.** Создать базу данных «Аэропорт». Объекты предметной области следующие:

*Самолеты:*

- ✓ код т/с
- ✓ тип самолета
- ✓ номер рейса
- ✓ маршрут

- ✓ дата вылета
- ✓ время вылета

Ключевое поле – код т/с.

*Пассажиры:*

- ✓ фамилия
- ✓ имя
- ✓ отчество
- ✓ паспортные данные

Ключевое поле – паспортные данные.

*Билеты:*

- ✓ номер билета
- ✓ код т/с
- ✓ номер места
- ✓ дальность полета
- ✓ стоимость
- ✓ паспортные данные

Ключевое поле – номер билета.

**Вариант №3.** Создать базу данных ”Железнодорожный вокзал”.

Объекты предметной области следующие:

*Пассажиры:*

- ✓ код пассажира
- ✓ фамилия
- ✓ имя
- ✓ отчество
- ✓ № паспорта

Ключевое поле - код пассажира.

*Поезда:*

- ✓ № поезда
- ✓ тип поезда
- ✓ пункт отправления
- ✓ пункт прибытия
- ✓ количество вагонов

Ключевое поле - № поезда.

*Билеты:*

- ✓ № билета
- ✓ код пассажира
- ✓ № поезда
- ✓ время отправления
- ✓ время прибытия
- ✓ номер вагона
- ✓ пункт назначения
- ✓ стоимость проезда

Ключевое поле - № билета.



**Вариант №4.** Создать базу данных **”Библиотека”**. Объекты предметной области следующие:

*Читатели:*

- ✓ код читателя
- ✓ фамилия
- ✓ имя
- ✓ отчество
- ✓ домашний адрес
- ✓ телефон читателя
- ✓ место работы или учебы.

Ключевое поле – код читателя.

*Книги:*

- ✓ код книги
- ✓ автор (фамилия)
- ✓ название
- ✓ издательство
- ✓ год издания
- ✓ количество страниц
- ✓ стоимость

Ключевое поле – код книги.

*Заказы:*

- ✓ код заказа
- ✓ дата заказа
- ✓ код читателя
- ✓ код книги
- ✓ дата возврата

Ключевое поле – код заказа.

**Вариант №5.** Создать базу данных **”Овощной магазин”**. Объекты предметной области следующие:

*Сведения о клиентах:*

- ✓ код клиента
- ✓ название фирмы
- ✓ адрес (город указать обязательно)
- ✓ телефон
- ✓ контактное лицо

Ключевое поле – код клиента.

*Сведения о товарах:*

- ✓ код товара
- ✓ наименование
- ✓ единица измерения
- ✓ стоимость единицы продукции
- ✓ количество на складе

Ключевое поле – код товара.

*Сведения о поступивших заказах:*

- ✓ № заказа
- ✓ дата заказа
- ✓ код клиента
- ✓ код товара
- ✓ количество
- ✓ способ расчета
- ✓ дата исполнения

Ключевое поле – № заказа.

**Вариант №6.** Создать базу данных **”Командировки”**. Объекты предметной области следующие:

*Сведения о сотрудниках:*

- ✓ личный номер сотрудника
- ✓ фамилия
- ✓ имя
- ✓ отчество
- ✓ должность
- ✓ год рождения
- ✓ оклад

Ключевое поле – личный номер сотрудника.

*Сведения о филиалах:*

- ✓ № филиала
- ✓ наименование
- ✓ e-mail
- ✓ месторасположение (город)

Ключевое поле – № филиала.

*Сведения о командировках:*

- ✓ № п/п
- ✓ личный номер сотрудника
- ✓ № филиала
- ✓ дата убытия
- ✓ дата прибытия
- ✓ аванс

Ключевое поле – № п/п.

**Вариант №7.** Создать базу данных **”Деканат”**. Объекты предметной области следующие:

*Студенты:*

- ✓ № зачетной книжки
- ✓ фамилия студента
- ✓ имя студента
- ✓ отчество студента
- ✓ № группы

Ключевое поле – № зачетной книжки.

*Преподаватели:*

- ✓ код преподавателя
- ✓ фамилия преподавателя
- ✓ имя преподавателя
- ✓ отчество преподавателя
- ✓ заработная плата

Ключевое поле – код преподавателя.

*Дисциплины:*

- ✓ код дисциплины
- ✓ наименование дисциплины
- ✓ количество часов

Ключевое поле – код дисциплины.

*Оценки:*

- ✓ № п/п
- ✓ № зачетной книжки
- ✓ код дисциплины
- ✓ код преподавателя
- ✓ оценка

Ключевое поле – № п/п.

**Вариант №8.** Создать базу данных **”Оператор сотовой связи”**. Объекты предметной области следующие:

*Абоненты:*

- ✓ телефонный номер абонента
- ✓ фамилия
- ✓ имя
- ✓ отчество
- ✓ адрес (город проживания указать обязательно)
- ✓ код тарифа

Ключевое поле – телефонный номер абонента.

*Тарифы:*

- ✓ код тарифа
- ✓ наименование тарифа
- ✓ абонентская плата
- ✓ количество любимых номеров
- ✓ стоимость минуты разговора

Ключевое поле – код тарифа.

*Расчеты за месяц:*

- ✓ № п/п
- ✓ телефонный номер абонента
- ✓ код тарифа
- ✓ продолжительность разговоров

Ключевое поле – № п/п.

**Вариант №9.** Создать базу данных **”Химчистка”**. Объекты предметной области следующие:

*Клиенты:*

- ✓ код клиента
- ✓ фамилия
- ✓ адрес (город проживания указать обязательно)
- ✓ телефон

Ключевое поле – код клиента.

*Услуги:*

- ✓ код услуги
- ✓ наименование услуги
- ✓ стоимость услуги

Ключевое поле – код услуги.

*Заказы:*

- ✓ № заказа
- ✓ код клиента
- ✓ код услуги
- ✓ дата обращения
- ✓ дата выполнения заказа

Ключевое поле – № заказа.

**Вариант №10.** Создать базу данных **”Торговая политика”**. Объекты предметной области следующие:

*Информация о странах:*

- ✓ код страны
- ✓ страна
- ✓ столица
- ✓ официальный язык
- ✓ денежные единицы

Ключевое поле – код страны.

*Информация о товарах:*

- ✓ код товара
- ✓ наименование
- ✓ единица измерения
- ✓ стоимость единицы продукции в Вг

Ключевое поле – код товара.

*Экспорт Республики Беларусь:*

- ✓ № операции
- ✓ код товара
- ✓ код страны (получателя товара)
- ✓ количество экспортируемого товара

Ключевое поле – № операции.

**Вариант №11.** Создать базу данных **”Автосалон”**. Объекты предметной области следующие:

*Автомобили:*

- ✓ код авто
- ✓ марка
- ✓ модель
- ✓ цвет
- ✓ пробег
- ✓ год выпуска
- ✓ объем двигателя
- ✓ стоимость.

Ключевое поле – код авто

*Покупатели:*

- ✓ код покупателя;
- ✓ фамилия;
- ✓ имя;
- ✓ отчество;
- ✓ паспортные данные

Ключевое поле – код покупателя.

*Сведения о продажах:*

- ✓ № чека
- ✓ код покупателя
- ✓ код авто
- ✓ дата
- ✓ скидка

Ключевое поле – № чека

**Вариант №12.** Создать базу данных "Компьютерный клуб". Объекты предметной области следующие:

*Клиенты:*

- ✓ код клиента
- ✓ фамилия
- ✓ адрес
- ✓ телефон

Ключевое поле – код клиента.

*Услуги:*

- ✓ код услуги
- ✓ наименование услуги
- ✓ стоимость услуги за 1 час работы

Ключевое поле – код услуги.

*Работа компьютерного клуба:*

- ✓ № п/п
- ✓ дата
- ✓ код услуги
- ✓ код клиента
- ✓ время работы

## Лабораторная работа № 2.

### Анализ данных таблицы с помощью запросов

**Цель работы:** Изучить структуру объектов базы данных - Запросов; научиться пользоваться простейшими логическими операндами; получить практику выборки заданной информации.

#### Теоретическая часть

Для быстрого доступа к данным есть гибкое и удобное средство – запросы. Для одной и той же таблицы можно создать множество различных запросов, каждый из которых сможет извлекать из таблицы лишь малую часть информации, но именно ту часть, которая в данный момент необходима. В результате работы запроса из общей исходной базы формируется результирующая таблица, содержащая часть общей информации, соответствующая запросу.

Таблица 2.1- Примеры выражений

	Выражение	Результаты отбора
Пишется в строке «Условие отбора:»	> 250	Отбор записей, значение поля которых больше 250
	>= "Петров"	Отбор записей, содержащих фамилии, начиная от Петрова и до конца алфавита
	Between #02.02.2021# And #03.03.2021#	Отбор записей, содержащих в поле даты дату в диапазоне от 2 февраля 2021 г. до 3 марта 2021 г.
	"Гомель" Or "Минск"	Отбор записей, содержащих в поле значение «Гомель» или «Минск»
	Like "С*"	Отбор записей, имена которых в поле начинаются с буквы С
	Like "*к"	Отбор записей, названия которых в поле заканчиваются на букву к
	Like "[Б-Р]*"	Отбор записей, имена которых находятся в диапазоне от Б до Р
Пишется в строке «Поле»	ПолноеИмя: [Имя] + " " + [Фамилия]	Отображение в поле «Полное имя» значения полей «Имя» и «Фамилия», разделенных пробелом
	ОбъемЗаказа: [Количество] * [Цена]	Отображение в поле «ОбъемЗаказа» произведения значений полей «Количество» и «Цена»
	ВремяДоставки: [ДатаРазмещения] - [ДатаИсполнения]	Отображение в поле «ВремяДоставки» разницы между значениями полей «ДатаРазмещения» и «ДатаИсполнения»

Важным свойством запросов является то, что при создании результирующей таблицы можно не только выбирать информацию из базы, но и обрабатывать ее. При работе запроса данные могут упорядочиваться (сортироваться), фильтроваться, объединяться, разделяться, изменяться, и при этом никаких изменений в базовых таблицах может не происходить.

И еще одним свойством запросов является их способность выполнять итоговые вычисления. Запрос может не только выдать результирующую таблицу, но и найти, например, среднее (наибольшее, наименьшее, суммарное и т.п.) значение по какому-то полю.

Существует немало различных видов запросов, но самые простые из них и, к тому же, используемые наиболее часто – это *запросы на выборку*. Цель запроса на выборку состоит в создании результирующей таблицы, в которой отображается только нужные по условию запроса данные из базовых таблиц. Примеры условий отбора, применяемые при извлечении записей, представлены в табл. 2.1.

Существуют и другие виды запросов, некоторые из них выполняются на базе предварительно созданного запроса на выборку. Например, запрос с параметрами - это запрос, при выполнении отображающий в собственном диалоговом окне приглашение ввести данные (условие для возвращения записей) или значение, которое требуется вставить в поле.

## Практическая часть

### Ход работы

*Необходимо создать запросы, используя базу данных, созданную в лабораторной работе №1.*

**Создание запроса на выборку «Полный список договоров».** Для того чтобы создать запрос необходимо выполнить следующие шаги:

1. Открыть файл базы данных **Автосалон**.
2. Нажать кнопку **Создание**, на панели инструментов нажать кнопку **Конструктор запросов**.
3. В диалоговом окне **Добавление таблицы** указать на таблицы *Договора* и *АвтоСклад* и нажать кнопку **Добавить**, а затем кнопку **Заккрыть**.

Поле:	Дата	Марка авто	Количество	Сумма: [АвтоСклад]![Цена]*[Договоры]![Количество]
Имя таблицы:	Договоры	АвтоСклад	Договоры	
Сортировка:	по возрастанию			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:				
или:				

Рис. 2.1 – Окно создания запроса

4. Выполнить отбор полей.
5. Упорядочить по возрастанию данные по значению поля *Дата*.
6. В бланк запроса дописать в строке *Поле* новое поле *Сумма*:  $[АвтоСклад]![Цена]*[Договора]![Количество]$  (см. рис. 2.1).
7. Просмотреть запрос в режиме таблицы.
8. Задать имя запроса – *Полный список договоров*.
9. После обработки запросом результирующая таблица будет иметь следующий вид (таблица 2.2).

Дата	Марка авто	Количество	Сумма
03.01.2021	Фольксваген	3	30 000
10.01.2021	Рено	6	168 000
30.10.2021	Ауди	8	192 000
05.11.2021	Ауди	1	24 000
08.11.2021	Мерседес	3	180 000
12.11.2021	Лексус	1	100 000
15.11.2021	Жигули	9	450 000
*			

Таблица 2.2- Полный список договоров

**Создание запроса “Список договоров с условием”.** Для того чтобы создать запрос *Список договоров с условием* в MS Access необходимо:

1. Создать новый запрос с помощью конструктора на основе запроса *Полный список договоров*. Выбрать все поля.
2. Отобразить записи, относящиеся к текущему месяцу и у которых сумма <60000. Для этого в условии отбора в поле *Дата* ввести Like *"\*.11.\*"*, а в поле *Сумма* соответственно «<60000» (см. рис. 2.3):

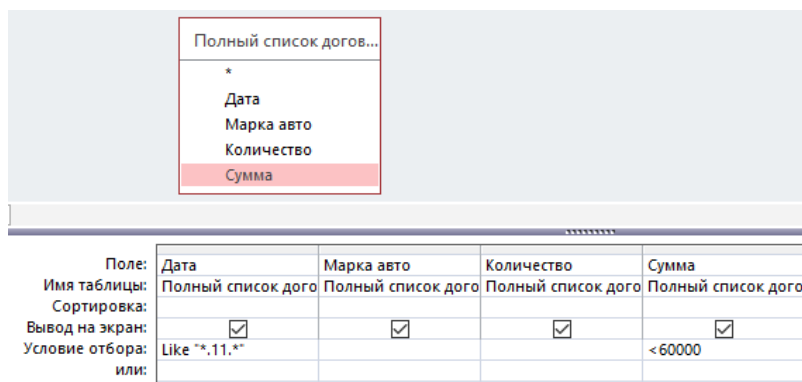


Рис. 2.3 – Окно создания запроса

3. Сохранить запрос под именем – *Список договоров с условием*
4. Просмотреть запрос в режиме таблицы. Убедиться, что выполнен отбор записей.

### **Создание запроса “Параметрический запрос”**

1. Создать новый запрос с помощью конструктора на основе запроса *Полный список договоров*. Выбрать все поля.



- В строке *Условие отбора* в поле *МаркаАвто* ввести [Введите марку автомобиля] (см. рис. 2.4):

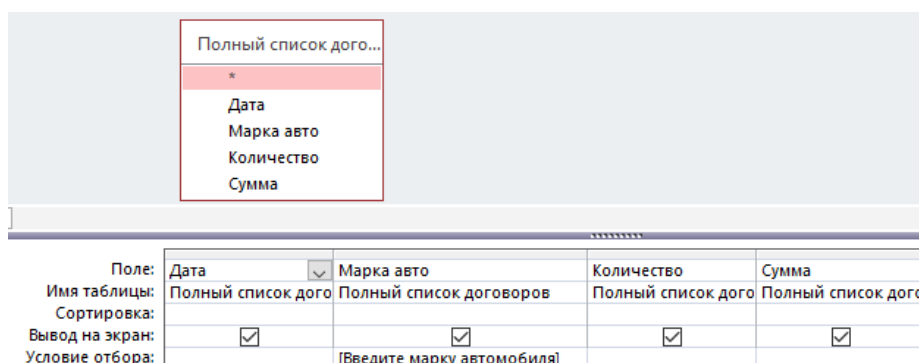


Рис. 2.4 – Окно создания запроса

- Сохранить запрос под именем – *Параметрический запрос*.

- При переходе в режим таблицы в появившемся окне ввести требующуюся марку авто.

**Создание запроса “Список договоров с условием 2”.** Для того чтобы создать запрос *Список договоров с условием 2* в MS Access необходимо:

- Создать новый запрос с помощью конструктора, на основе запроса *Полный список договоров*. Выбрать все поля.

- В условии отбора в поле *Дата* ввести «Between #01.11.2021# And #30.11.2021#» (см. рис. 2.5).

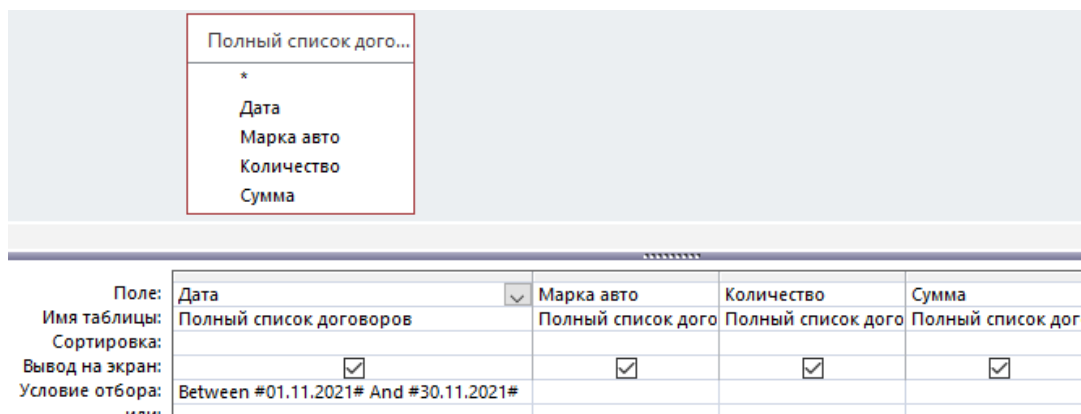


Рис. 2.5 – Окно создания запроса

- Сохранить запрос под именем – *Список договоров с условием 2*.

- Просмотреть запрос в режиме таблицы.

### КОНТРОЛЬНЫЕ ВОПРОСЫ:

- Для чего служат запросы на выборку?
- По какому количеству критериев можно отбирать информацию в одном поле запроса?
- Что является результатом работы запроса?
- Каковы основные свойства запросов?

5. Приведите примеры выражений, используемых в запросах.
6. Приведите примеры запросов с параметрами.
7. Какие способы вставки полей в запрос вы знаете?
8. Для чего служит оператор Like?

### **ЗАДАНИЕ**

Создать указанные запросы в базе данных, созданной в лабораторной работе №2 согласно варианту задания.

### **ВАРИАНТЫ ЗАДАНИЙ**

#### ***Вариант №1.***

- 1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Магазины”, “Товары”, “Поставщики”:
  - дата заказа;
  - тип товара;
  - сроки поставки (в днях);
  - страна - производитель;
  - цена товара.

Название запроса “*Полный список товаров и поставщиков*”.

- 2) Сформировать запрос на выборку, который выводит на экран список товаров, которые изготовлены в [Задать страну - изготовитель самостоятельно]. Название запроса “*Список товаров из выбранной страны*”.
- 3) Сформировать запрос на выборку, который выводит на экран список поставщиков товара, офис которых располагается в городе [Задать самостоятельно]. Название запроса “*Список поставщиков по городам*”.
- 4) Сформировать запрос на выборку, который выводит на экран список товаров, заказанных в период с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса “*Список заказанных товаров*”.
- 5) Сформировать запрос на выборку, который выводит на экран список товаров, которые были произведены в [Задать страну - изготовитель самостоятельно] по цене не более [Задать самостоятельно]. Название запроса “*Список товаров по цене, произведенных в определенной стране*”.
- 6) Сформировать запрос на выборку, который выводит на экран данные о дате прибытия товара. Для этого создать в запросе новое поле, в котором ввести *Дата доставки: [дата заказа] + [сроки поставки (в днях)]*. Название запроса “*Список товаров по дате прибытия*”.
- 7) Сформировать запрос на выборку, который выводит на экран список товаров определенного типа [Задать самостоятельно] срок хранения которых не менее [Задать самостоятельно]. Название запроса “*Срок хранения товаров определенного типа*”

- 8) Сформировать запрос на выборку, который выводит на экран список товаров поставляемых определенной фирмой. Название фирмы - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса *“Запрос с параметрами”*.

**Вариант №2.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Самолеты”, “Пассажиры”, “Билеты”:

- тип самолета;
- номер рейса;
- маршрут;
- дата вылета;
- время вылета;
- фамилия;
- номер места;
- дальность полета;
- стоимость.

Название запроса *“Данные аэропорта”*.

2) Сформировать запрос на выборку, который выводит на экран список пассажиров, фамилии которых находятся в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона [Задать самостоятельно]. Название запроса *“Алфавитный список пассажиров”*.

3) Сформировать запрос на выборку, который выводит на экран рейсы, дальность полета которых более [Задать самостоятельно]. Название запроса *“Список рейсов с большой дальностью полета”*.

4) Сформировать запрос на выборку, который выводит на экран список самолетов вылетевших из аэропорта в период времени с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса *“Список самолетов за определенный промежуток времени”*.

5) Сформировать запрос на выборку, который выводит на экран список самолетов летящих по определенному маршруту [Задать самостоятельно]. Название *“Список самолетов, летящих по заданному маршруту”*.

6) Сформировать запрос на выборку, который выводит на экран список пассажиров, чья серия паспорта начинается с буквы [Задать самостоятельно], заплативших за билет более определенной суммы [Задать самостоятельно]. Название запроса *“Список состоятельных клиентов”*.

7) Сформировать запрос на выборку, рассчитывающий стоимость билета в евро. Курс евро - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса *“Стоимость билета в евро”*.

8) Сформировать запрос на выборку, который выводит на экран список пассажиров, чья фамилия начинается с буквы [Задать самостоятельно], вылетевших по маршруту [Задать самостоятельно], в период с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса *“Список пассажиров”*.

### **Вариант №3.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Поезда”, “Билеты”, “Пассажиры”:

- тип поезда;
- пункт отправления;
- пункт прибытия;
- время отправления;
- время прибытия;
- фамилия;
- номер вагона;
- № паспорта;
- пункт назначения;
- стоимость проезда.

Название запроса “*Исходные данные железнодорожного вокзала*”.

2) Сформировать запрос на выборку, который выводит на экран список пассажиров, фамилии которых находятся в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона [Задать самостоятельно]. Название запроса “*Алфавитный список пассажиров*”.

3) Сформировать запрос на выборку, который выводит на экран список поездов направляющихся в пункт назначения - [Задать самостоятельно]. Название запроса “*Список поездов*”.

4) Сформировать запрос на выборку, который выводит на экран список поездов отправившихся в период с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса “*Список поездов за определенный промежуток времени*”.

5) Сформировать запрос на выборку, который выводит на экран список поездов, идущих из пункта отправления [Задать самостоятельно]. Название “*Список поездов из заданного пункта*”.

6) Сформировать запрос на выборку, который выводит на экран список пассажиров, чья фамилия начинается с буквы [Задать самостоятельно] и которые отправились на поезде определенного типа [Задать самостоятельно]. Название запроса “*Список пассажиров*”.

7) Сформировать запрос, позволяющий увеличить стоимость билета на определенный процент. Процент - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “*Параметрический запрос*”.

8) Сформировать запрос на выборку, который выводит на экран список пассажиров, чья фамилия начинается с буквы [Задать самостоятельно], выехавших из пункта [Задать самостоятельно] в определенное время [Задать самостоятельно]. Название запроса “*Список определенных пассажиров*”.

### **Вариант №4.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Читатели”, “Книги”, “Заказы”:

- фамилия;

- домашний адрес;
- автор;
- название;
- дата заказа;
- дата возврата.

Название запроса *“Исходные данные библиотеки”*.

2) Сформировать запрос на выборку, который выводит на экран список читателей, фамилии которых находятся в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона *[Задать самостоятельно]*. Название запроса *“Алфавитный список читателей”*.

3) Сформировать запрос с параметрами, который выводит на экран список читателей, проживающих по адресу *[адрес должен будет вводиться с клавиатуры]*. Название запроса *“Список читателей по месту жительства”*.

4) Сформировать запрос на выборку, который выводит на экран список книг, взятых читателями в период с *[Задать самостоятельно]* до *[Задать самостоятельно]*. Название запроса *“Список книг, взятых читателями”*.

5) Сформировать запрос на выборку, который выводит на экран список книг по фамилии автора *[Ввести с клавиатуры]*. Название *“Поиск книг по автору”*.

6) Сформировать запрос на выборку, который выводит на экран список читателей, чья фамилия начинается с буквы *[Задать самостоятельно]*, и которые взяли книгу автора *[Задать самостоятельно]*. Название запроса *“Список читателей”*.

7) Сформировать запрос на выборку, рассчитывающий срок пользования книгой с даты заказа до даты возврата. Для этого создать новое поле в запросе и ввести *“Срок пользования: [дата возврата] - [дата заказа]”*. Название запроса *“Срок пользования”*.

8) Сформировать запрос на выборку, который выводит на экран список книг по фамилии автора *[фамилия автора должна вводиться с клавиатуры]*, и которые были заказаны в период с *[Задать самостоятельно]* до *[Задать самостоятельно]*. Название запроса *“Список книг”*.

#### **Вариант №5.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц *“Сведения о поступивших заказах”*, *“Сведения о товарах”*, *“Сведения о клиентах”*:

- дата заказа;
- дата исполнения;
- количество;
- наименование;
- стоимость единицы продукции;
- название фирмы;
- адрес.

Название запроса *“Полный список товаров и клиентов”*.

2) Сформировать запрос на выборку, который выводит на экран список товаров, наименование которых начинается с буквы [Задать самостоятельно]. Название запроса “Список товаров”.

3) Сформировать запрос на выборку, который выводит на экран список клиентов, офис которых располагается в городе [Задать самостоятельно]. Название запроса “Список клиентов по городам”.

4) Сформировать запрос на выборку, который выводит на экран товары, заказанные в период с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса “Список заказанных товаров”.

5) Сформировать запрос на выборку, который выводит на экран товары, чья стоимость за единицу продукции выше указанной цены [Задать самостоятельно]. Название запроса “Список товаров по цене”.

6) Сформировать запрос на выборку, который выводит на экран данные о стоимости заказа. Для этого в запросе создать новое поле, в котором ввести: Стоимость: [стоимость единицы продукции]\*[количество]. Название запроса “Стоимость заказа”.

7) Сформировать запрос на выборку, который выводит на экран список товаров, которые следует заказывать у поставщика вследствие ограниченного количества на складе (меньше определенного количества [Задать самостоятельно]). Название запроса “Список товаров по количеству на складе”.

8) Сформировать запрос на выборку, который выводит на экран список товаров заказных определенным клиентом в определенный период времени. Клиент- параметр, запрашиваемый у пользователя с клавиатуры, период задать самостоятельно с... до.... Название запроса “Запрос с параметрами”.

#### **Вариант №6.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Сведения о командировках”, “Сведения о филиалах”, “Сведения о сотрудниках”:

- дата прибытия;
- дата убытия;
- аванс;
- наименование;
- месторасположение;
- фамилия;
- оклад.

Название запроса “Исходные данные по сотрудникам, находящимся в командировках”.

2) Сформировать запрос на выборку, который выводит на экран список сотрудников, фамилии которых начинаются в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона [Задать самостоятельно]. Название запроса “Алфавитный список сотрудников”.

3) Сформировать запрос на выборку, который выводит на экран список сотрудников, чей оклад более определенной суммы [*Задать самостоятельно*]. Название запроса “Список сотрудников”.

4) Сформировать запрос на выборку, который выводит на экран список сотрудников, отправившихся в командировку в период с [*Задать самостоятельно*] до [*Задать самостоятельно*]. Название запроса “Список сотрудников в командировках”.

5) Сформировать запрос на выборку, который выводит на экран список филиалов, которые находятся в городе [*Задать самостоятельно*]. Название “Список филиалов по городам”.

6) Сформировать запрос на выборку, рассчитывающий аванс, выданный сотруднику, в евро. Курс евро - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Аванс в евро”.

7) Сформировать запрос, позволяющий увеличить оклад сотрудника на определенный процент. Процент - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Параметрический запрос”.

8) Сформировать запрос на выборку, который выводит на экран список сотрудников, чья фамилия начинается с буквы [*Задать самостоятельно*], отправившихся в филиал [*Задать самостоятельно*] в определенный период времени с [*Задать самостоятельно*] до [*Задать самостоятельно*]. Название запроса “Список определенных сотрудников”.

#### **Вариант №7.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Студенты”, “Преподаватели”, “Дисциплины”, “Оценки”:

- фамилия студента;
- фамилия преподавателя;
- наименование дисциплины;
- количество часов;
- оценка.

Название запроса “Исходные данные по деканату”.

2) Сформировать запрос на выборку, который выводит на экран список студентов, фамилии которых начинаются в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона [*Задать самостоятельно*]. Название запроса “Алфавитный список студентов”.

3) Сформировать запрос на выборку, который выводит на экран список преподавателей, у которых заработная плата больше [*Задать самостоятельно*]. Название запроса “Список преподавателей”.

4) Сформировать запрос на выборку, который выводит на экран список студентов, получивших оценку в интервале от [*Задать самостоятельно*] до [*Задать самостоятельно*]. Название запроса “Оценки студентов”.

5) Сформировать запрос на выборку, который выводит на экран список дисциплин, которые были изучены определенным студентом. Фамилия

студента- параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Список дисциплин”.

б) Сформировать запрос на выборку, который выводит на экран список преподавателей, чья фамилия начинается с определенной буквы [Задать самостоятельно], ведущих дисциплину, продолжительность изучения которой не менее [Задать самостоятельно] часов. Название запроса “Список преподавателей 2”.

7) Сформировать запрос на выборку, который выводит на экран полные данные о преподавателе. Для этого создать новое поле, ввести с клавиатуры: Полное имя: [фамилия преподавателя]+ " "+[имя преподавателя]+ " "+[отчество преподавателя] Название запроса “Полные данные”

8) Сформировать запрос, позволяющий увеличить заработную плату преподавателя на определенный процент. Процент - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Параметрический запрос”.

### **Вариант №8.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Абоненты”, “Тарифы”, “Расчеты за месяц”:

- телефонный номер абонента;
- фамилия;
- адрес;
- наименование тарифа;
- абонентская плата;
- стоимость минуты разговора;
- продолжительность разговоров.

Название запроса “Исходные данные по абонентам”.

2) Сформировать запрос на выборку, который выводит на экран список абонентов, фамилии которых начинаются в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона [Задать самостоятельно]. Название запроса “Алфавитный список абонентов”.

3) Сформировать запрос на выборку, который выводит на экран список абонентов, у которых абонентская плата более определенной суммы [Задать самостоятельно]. Название запроса “Список абонентов”.

4) Сформировать запрос на выборку, который выводит на экран список абонентов, проживающих в определенном городе [Задать самостоятельно]. Название запроса “Список абонентов по городам”.

5) Сформировать запрос на выборку, который выводит на экран список абонентов, которые подключены к тарифу [Задать самостоятельно]. Название “Список абонентов по тарифам”.

б) Сформировать запрос на выборку, который выводит на экран список абонентов, длительность разговоров которых находится в диапазоне с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса “Список абонентов по длительности разговоров”.



7) Сформировать запрос на выборку, который выводит на экран стоимость разговоров абонентов. Для этого необходимо в запросе создать новое поле, в которое ввести: Стоимость разговоров: [стоимость минуты разговора] \* [продолжительность разговоров]. Название запроса “*Стоимость разговоров*”.

8) Сформировать запрос, позволяющий уменьшить стоимость минуты разговора на определенный процент. Процент - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “*Параметрический запрос*”.

#### **Вариант №9.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Клиенты”, “Услуги”, “Заказы”:

- фамилия;
- адрес;
- наименование услуги;
- стоимость услуги;
- дата обращения;
- дата выполнения заказа.

Название запроса “*Исходные данные по клиентам*”.

2) Сформировать запрос на выборку, который выводит на экран список клиентов, фамилии которых начинаются в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона [Задать самостоятельно]. Название запроса “*Алфавитный список клиентов*”.

3) Сформировать запрос на выборку, который выводит на экран список клиентов, кому были оказаны услуги по стоимости более [Задать самостоятельно]. Название запроса “*Список клиентов*”.

4) Сформировать запрос на выборку, который выводит на экран список клиентов, проживающих в городе [Задать самостоятельно]. Название запроса “*Список клиентов по городам*”.

5) Сформировать запрос на выборку, который выводит на экран список клиентов, которым была оказана определенная услуга. Наименование услуги - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “*Параметрический запрос*”.

6) Сформировать запрос на выборку, который выводит на экран список клиентов, которые обратились в химчистку в период с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса “*Список клиентов по дате обращения*”.

7) Сформировать запрос на выборку, рассчитывающий длительность оказания услуги. Для этого в запросе необходимо создать новое поле, в котором ввести: Длительность работы: [дата выполнения заказа] - [дата обращения]. Название запроса “*Длительность оказания услуг*”.

8) Сформировать запрос на выборку, который выводит на экран список услуг, стоимость которых не более определенной суммы [Задать самостоятельно] и которые были выполнены в период с [Задать

*самостоятельно]* до *[Задать самостоятельно]*. Название запроса “Список определенных услуг”.

#### **Вариант №10.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Информация о странах”, “Информация о товарах”, “Экспорт Республики Беларусь”:

- страна;
- столица;
- официальный язык;
- наименование;
- стоимость единицы продукции в Bг;
- количество экспортируемого товара.

Название запроса “Исходные данные по экспорту”.

2) Сформировать запрос на выборку, который выводит на экран список товаров, которые будут поставляться в страны, название которых начинается с буквы *[Задать самостоятельно]*. Название запроса “Список товаров”.

3) Сформировать запрос на выборку, который выводит на экран список стран, где официальный язык *[Задать самостоятельно]*. Название запроса “Список стран с определенным официальным языком”.

4) Сформировать запрос на выборку, который выводит на экран товары, у которых стоимость единицы продукции не более *[Задать самостоятельно]*. Название запроса “Список товаров 2”.

5) Сформировать запрос на выборку, который выводит на экран данные о товарах, которые экспортируются в количестве от *[Ввести с клавиатуры]* до *[Ввести с клавиатуры]*. Название запроса “Количество экспортируемого товара”.

6) Сформировать запрос, позволяющий увеличить стоимость единицы продукции в Bг на определенный процент. Процент - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Параметрический запрос”.

7) Сформировать запрос на выборку, рассчитывающий стоимость единицы продукции в евро. Курс евро - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Стоимость единицы продукции в евро”.

8) Сформировать запрос на выборку, который выводит на экран список стран, денежная единица которых - *[Задать самостоятельно]*. Название запроса “Список стран”.

#### **Вариант №11.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Автомобили”, “Покупатели”, “Сведения о продажах”:

- марка;
- модель;
- пробег;
- объем двигателя;
- стоимость;

- фамилия;
- дата;
- скидка.

Название запроса *“Полный список автомобилей и покупателей”*.

2) Сформировать запрос на выборку, который выводит на экран автомобили определенной марки. Марка автомобиля - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса *“Список автомобилей определенной марки”*.

3) Сформировать запрос на выборку, который выводит на экран список автомобилей определенной марки [Задать самостоятельно], модель которых начинается на букву [Задать самостоятельно]. Название запроса *“Список моделей”*.

4) Сформировать запрос на выборку, который выводит на экран список автомобилей, проданных в период с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса *“Список продаж”*.

5) Сформировать запрос на выборку, который выводит на экран автомобили, объем двигателя которых более [Задать самостоятельно] и пробег которых не более [Задать самостоятельно]. Название запроса *“Список автомобилей”*.

6) Сформировать запрос, позволяющий увеличить стоимость автомобиля на определенный процент. Процент - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса *“Параметрический запрос”*.

7) Сформировать запрос на выборку, рассчитывающий стоимость автомобиля в евро. Курс евро - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса *“Стоимость автомобиля в евро”*.

8) Сформировать запрос на выборку, который выводит на экран список автомобилей, год выпуска которых [Задать самостоятельно]. Название запроса *“Список автомобилей 2”*.

### **Вариант №12.**

1) Сформировать запрос на выборку. Запрос должен содержать следующие поля из таблиц “Услуги”, “Клиенты”, “Работа компьютерного клуба”:

- наименование услуги;
- стоимость услуги за 1 час;
- фамилия;
- адрес;
- дата;
- время работы.

Название запроса *“Полный список предоставляемых услуг”*.

2) Сформировать запрос на выборку, который выводит на экран список клиентов, фамилии которых начинаются в определенном буквенном диапазоне. Начальную и конечную буквы буквенного диапазона [Задать самостоятельно]. Название запроса *“Алфавитный список клиентов”*.

3) Сформировать запрос на выборку, который выводит на экран список клиентов, которым был оказан определенный вид услуг [Задать самостоятельно]. Название запроса “Список клиентов”.

4) Сформировать запрос на выборку, который выводит на экран список услуг, которые были оказаны в период с [Задать самостоятельно] до [Задать самостоятельно]. Название запроса “Список услуг”.

5) Сформировать запрос на выборку, рассчитывающий полную стоимость оказанных услуг. Для этого в запросе создать новое поле, в котором ввести: Стоимость: [стоимость услуги за 1 час работы]\*[время работы] Название запроса “Стоимость услуг”.

6) Сформировать запрос на выборку, который выводит на экран список клиентов, которые работали там больше [Задать самостоятельно]. Название запроса “Список клиентов 2”.

7) Сформировать запрос, позволяющий увеличить стоимость услуги за 1 час работы на определенный процент. Процент - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Параметрический запрос”.

8) Сформировать запрос на выборку, рассчитывающий стоимость услуги за 1 час работы в евро. Курс евро - параметр, запрашиваемый у пользователя с клавиатуры. Название запроса “Стоимость услуги за 1 час работы в евро”.

### Лабораторная работа № 3 Использование запросов

#### Цель работы:

*Приобрести практический опыт по разработке запросов на изменение, итоговых и перекрестных запросов.*

#### Теоретическая часть

В MS ACCESS можно создать следующие виды запросов:

- **Итоговые запросы**, назначение которых отдаленно напоминает итоговые функции электронных таблиц (производят математические вычисления по заданному полю и выдают результат).
- **Запросы на изменение** – позволяют автоматизировать заполнение полей таблиц. Существует четыре типа запросов на изменение: на удаление, на обновление и добавление записей, а также на создание таблицы.
- **Запрос на удаление**. Удаляет группу записей из одной или нескольких таблиц. Например, запрос на удаление позволяет удалить записи о товарах, поставки которых прекращены или на которые нет заказов. С помощью запроса на удаление можно удалять только всю запись, а не отдельные поля внутри нее.
- **Запрос на обновление**. Вносит общие изменения в группу записей одной или нескольких таблиц. Например, на 10 процентов поднимаются цены на продукты или на 5 процентов увеличивается зарплата сотрудников. Запрос

на обновление записей позволяет изменять данные в существующих таблицах.

- *Запрос на добавление.* Добавляет группу записей из одной или нескольких таблиц в конец одной или нескольких таблиц. Например, появилось несколько новых клиентов, а также база данных, содержащая сведения о них. Чтобы не вводить все данные вручную, их можно добавить в таблицу с помощью запроса на добавление.
- *Запрос на создание таблицы.* Создает новую таблицу в текущей БД или другой БД на основе всех или части данных из одной или нескольких таблиц, запросов.
- *Перекрестные запросы,* позволяющие создавать результирующие таблицы на основе результатов расчетов, полученных при анализе группы таблиц. В перекрестном запросе отображаются результаты статистических расчетов (суммы, количество записей и средние значения), выполненных по данным из одного поля таблицы. Эти результаты группируются по двум наборам данных, один из которых расположен в левом столбце таблицы, а второй — в верхней строке.
- *Специфические запросы SQL* – запросы к серверу базы данных, написанные на языке запросов SQL.

## Практическая часть

### Ход работы

*Необходимо создать запросы, используя базу данных, созданную в лабораторной работе №2.*

**Создание запроса «Запрос на создание таблицы в текущей БД».** Для того чтобы создать запрос «Запрос на создание таблицы в текущей БД» в MS Access необходимо:

1. Создать новый запрос с помощью конструктора на основе запроса *Полный список договоров*. Выбрать все поля.

2. Задать имя новой таблицы «Продажи» с помощью меню Запрос→Создание таблицы (см. рис. 3.1). Сохранить запрос под именем – *Запрос на создание таблицы в текущей БД*.

3. Для получения результата работы созданного запроса следует при запуске подтвердить изменения данных таблицы.

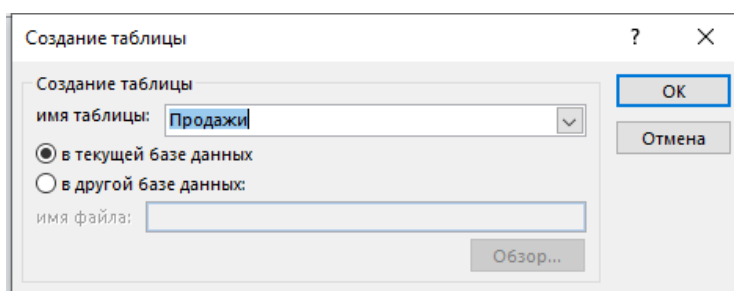


Рис. 3.1. Диалоговое окно «Создание таблицы»

4. Просмотреть результат можно, перейдя на вкладку Таблицы.

**Создание запроса «Запрос на обновление».** Для того чтобы создать запрос «*Запрос на обновление*» в MS Access необходимо:

1. Создать новый запрос с помощью конструктора на основе таблицы «Автосклад».

2. Указать тип запроса с помощью меню Запрос→Обновление.

3. Выбрать поле «Цена».

4. В строке обновление ввести: [Цена]\*1,05.

5. Сохранить запрос под именем – Запрос на обновление.

6. Для получения результата работы созданного запроса следует при запуске подтвердить изменения данных таблицы. Просмотреть результат можно, открыв таблицу «Автосклад».

**Создание запроса «Запрос на удаление».** Для того чтобы создать запрос «*Запрос на удаление*» в MS Access необходимо:

1. Создать новый запрос с помощью конструктора на основе таблицы «Агенты».

2. Указать тип запроса с помощью меню Запрос→Удаление.

3. Выбрать поле «ФИО».

4. В строке *Условие отбора* ввести: Like "Ц\*".


5. Сохранить запрос под именем – Запрос на удаление.

6. Для получения результата работы созданного запроса следует при запуске подтвердить изменения данных таблицы. Просмотреть результат можно, открыв таблицу «Агенты».

**Создание запроса «Перекрестный запрос».** Для того чтобы создать запрос «*Перекрестный запрос*» в MS Access необходимо:

1. В списке **Объекты** выбрать *Запросы* и нажать кнопку Создать. В диалоговом окне **Новый запрос** выбрать вариант *Перекрестный запрос*.

2. В появившемся диалоговом окне выбрать таблицу *Продажи*, после чего нажать кнопку Далее (см. рис. 4.2).

3. В следующем диалоговом окне с помощью кнопки  выбрать поле *МаркаАвто*, значения которого будут использованы в качестве заголовков строк, после чего нажать кнопку Далее.

4. В следующем диалоговом окне выбрать поле *Дата*, значения которого будут использованы в качестве заголовков столбцов, после чего нажать кнопку Далее.

5. В следующем диалоговом окне выбрать *Дата* – это интервал, с которым будет сгруппирован столбец данных типа даты и времени. После чего нажать кнопку Далее.

6. В следующем диалоговом окне выбрать поле *Количество* и функцию *Сумма*.

7. В следующем диалоговом окне задать имя запроса *Перекрестный запрос* и нажать кнопку Готово.

**Создание запроса «Итоговый запрос».** Для того чтобы создать запрос «Итоговый запрос» в MS Access необходимо:

1. Создать новый запрос с помощью конструктора на основе запроса *Полный список договоров*.
2. Нажать кнопку *Итоги* на панели инструментов.
3. Выбрать поля «МаркаАвто» и «Сумма».
4. В строке Групповая операция в поле Сумма выбрать Sum.
5. Сохранить запрос под именем – Итоговый запрос.

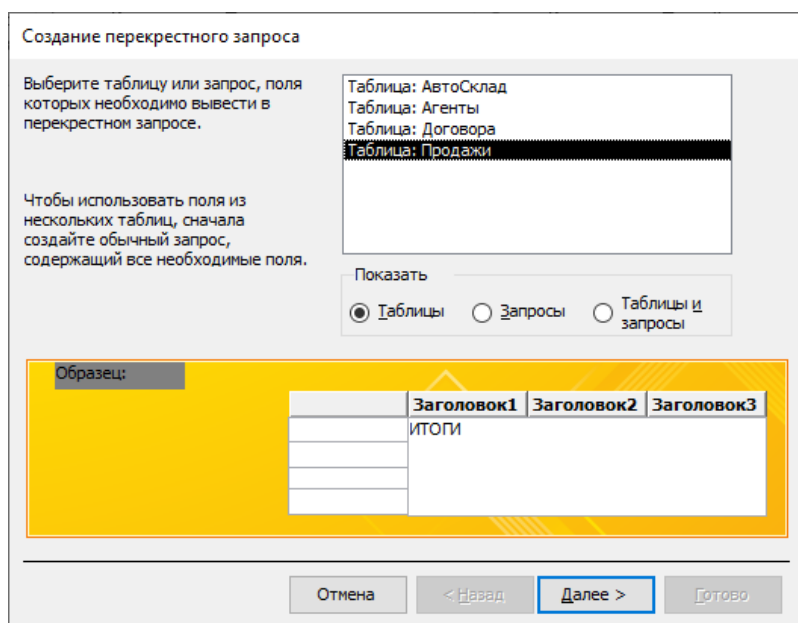


Рис. 3.2. Диалоговое окно «Создание перекрестных таблиц»

### КОНТРОЛЬНЫЕ ВОПРОСЫ:

1. Какие запросы позволяют автоматизировать заполнение полей таблиц?
2. Приведите примеры итоговых запросов.
3. Приведите примеры перекрестных запросов.
4. Для чего используются запросы на обновление?
5. Опишите действия при создании запроса на создание таблицы в другой базе данных.
6. Опишите действия при создании перекрестного запроса.

### ЗАДАНИЕ

Создать в своей папке новую базу данных Новая.mdb и указанные запросы в базе данных, созданной в лабораторной работе №3, согласно варианту задания.

### ВАРИАНТЫ ЗАДАНИЙ

#### *Вариант №1.*

1. На основе запроса «Полный список товаров и поставщиков» сформировать запрос на создание новой таблицы в текущей БД.

Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.

2. На основе запроса “Полный список товаров и поставщиков” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который уменьшает срок поставки (в днях) на одни сутки.
4. Создать запрос на удаление, который удаляет данные о товаре, который больше не заказывается.
5. Создать перекрестный запрос, источниками данных которого будут поля “Тип товара”, “Страна изготовитель”, “Цена товара” (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран суммарную стоимость заказанных товаров.

### ***Вариант №2.***

1. На основе запроса “Данные аэропорта” сформировать запрос на создание новой таблицы в текущей БД. Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.
2. На основе запроса “Данные аэропорта” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД “Новая”. Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает данные о времени вылета самолетов на 7 минут.
4. Создать запрос на удаление определенного рейса.
5. Создать перекрестный запрос, источниками данных которого будут поля “Маршрут”, “Фамилия”, «Стоимость» (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран суммарную стоимость купленных билетов.

### ***Вариант №3.***

1. На основе запроса “Исходные данные железнодорожного вокзала” сформировать запрос на создание новой таблицы в текущей БД. Название таблицы “Запрос на создание”. Название таблицы “Новая таблица”.
2. На основе запроса “ Исходные данные железнодорожного вокзала” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает время прибытия поезда на 2 минуты.
4. Создать запрос на удаление определенного пассажира.



5. Создать перекрестный запрос, источниками данных которого будут поля «Пункт отправления», «Пункт прибытия», «Пункт назначения», «Фамилия», «Стоимость проезда» (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран суммарную стоимость купленных билетов.

**Вариант №4.**

1. На основе запроса «Исходные данные библиотеки» сформировать запрос на создание новой таблицы в текущей БД. Название запроса «Запрос на создание». Название таблицы «Новая таблица».
2. На основе запроса «Исходные данные библиотеки» сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса «Экспорт данных», название таблицы «Импортированная таблица».
3. Создать запрос на обновление данных, который увеличивает стоимость всех книг на 2 %.
4. Создать запрос на удаление определенного читателя.
5. Создать перекрестный запрос, источниками данных которого будут поля «Автор (фамилия)», «Название», «Дата заказа», «Фамилия» (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран данные о самом старейшем издании в библиотеке (год издания - min).

**Вариант №5.**

1. На основе запроса «Полный список товаров и клиентов» сформировать запрос на создание новой таблицы в текущей БД. Название запроса «Запрос на создание». Название таблицы «Новая таблица».
2. На основе запроса «Полный список товаров и клиентов» сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса «Экспорт данных», название таблицы «Импортированная таблица».
3. Создать запрос на обновление данных, который увеличивает количество товара всех наименований имеющегося на складе на 30 %.
4. Создать запрос на удаление, который удаляет данные об определенном заказе.
5. Создать перекрестный запрос, источниками данных которого будут поля «Дата заказа», «Наименование», «Количество» (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран суммарное количество всех имеющихся товаров на складе.

**Вариант №6.**

1. На основе запроса «Исходные данные по сотрудникам, находящимся в командировках» сформировать запрос на создание новой таблицы в текущей БД. Название запроса «Запрос на создание». Название таблицы «Новая таблица».

2. На основе запроса “Исходные данные по сотрудникам, находящимся в командировках” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает оклад всех сотрудников на 4 %.
4. Создать запрос на удаление, который удаляет определенного сотрудника.
5. Создать перекрестный запрос, источниками данных которого будут поля “Фамилия”, “Дата убытия”, “Аванс” (на пересечении).
6. Создать итоговый запрос, который будет отображать всю сумму авансов выданных сотрудникам.

**Вариант №7.**

1. На основе запроса “Исходные данные по деканату” сформировать запрос на создание новой таблицы в текущей БД. Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.
2. На основе запроса “Исходные данные по деканату” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который уменьшает количество часов для изучения всех дисциплин на 1 %.
4. Создать запрос на удаление, который удаляет данные об определенном студенте.
5. Создать перекрестный запрос, источниками данных которого будут поля “Фамилия преподавателя”, “Наименование дисциплины”, “Количество часов” (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран наименование дисциплины, для изучения которой требуется максимальное количество часов.

**Вариант №8.**

1. На основе запроса “Исходные данные по абонентам” сформировать запрос на создание новой таблицы в текущей БД. Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.
2. На основе запроса “Исходные данные по абонентам” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает все стоимости минуты разговора на 12%.
4. Создать запрос на удаление, который удаляет определенного абонента.

5. Создать перекрестный запрос, источниками данных которого будут поля “Наименование тарифа”, “Фамилия”, “Продолжительность разговоров” (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран фамилию абонента, продолжительность разговоров которого максимальна.

**Вариант №9.**

1. На основе запроса “Исходные данные по клиентам” сформировать запрос на создание новой таблицы в текущей БД. Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.
2. На основе запроса “Исходные данные по клиентам” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает стоимость всех услуг на 6 %.
4. Создать запрос на удаление, который удаляет определенного клиента.
5. Создать перекрестный запрос, источниками данных которого будут поля “Наименование услуги”, “Фамилия”, “Стоимость услуги” (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран наименование услуги, стоимость которой максимальна.

**Вариант №10.**

1. На основе запроса “Исходные данные по экспорту” сформировать запрос на создание новой таблицы в текущей БД. Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.
2. На основе запроса “Исходные данные по экспорту” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает стоимость единицы продукции в Вг всех товаров на 7 %.
4. Создать запрос на удаление, который удаляет данные об определенном товаре.
5. Создать перекрестный запрос, источниками данных которого будут поля “Наименование”, “Страна”, “Количество экспортируемого товара” (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран суммарное количество экспортируемых товаров.

**Вариант №11.**

1. На основе запроса “Полный список автомобилей и покупателей” сформировать запрос на создание новой таблицы в текущей БД. Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.

2. На основе запроса “ Полный список товаров и покупателей ” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает скидки на 9 %.
4. Создать запрос на удаление, который удаляет определенный автомобиль.
5. Создать перекрестный запрос, источниками данных которого будут поля “Марка”, “Модель”, “Стоимость” (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран данные об автомобиле, стоимость которого минимальна.

#### **Вариант №12.**

1. На основе запроса “Полный список предоставляемых услуг” сформировать запрос на создание новой таблицы в текущей БД. Название запроса “Запрос на создание”. Название таблицы “Новая таблица”.
2. На основе запроса “Полный список предоставляемых услуг” сформировать запрос на создание новой таблицы, который экспортирует все имеющиеся данные в БД «Новая». Название запроса “Экспорт данных”, название таблицы “Импортированная таблица”.
3. Создать запрос на обновление данных, который увеличивает стоимость всех услуг за 1 час работы на 2 %.
4. Создать запрос на удаление, который удаляет определенного клиента.
5. Создать перекрестный запрос, источниками данных которого будут поля “Наименование услуги”, “Фамилия”, “Время работы” (на пересечении).
6. Создать итоговый запрос, который будет выводить на экран наименование услуги, стоимость у которой за 1 час работы максимальна.

### **Лабораторная работа № 4.**

#### **Проектирование и Использование форм**

##### **Цель работы:**

*Приобрести опыт разработки форм для ввода информации в таблицы базы данных; получить практические навыки использования различных элементов управления в формах Ms Access.*

##### **Теоретическая часть**

Форма – это средство представления информации для просмотра, изменения или печати данных в виде, удобном для восприятия пользователями. Использование форм существенно облегчает ввод и контроль данных.

Формы могут быть выведены на экран в трех видах: режим конструктора, режим формы и режим таблицы.

Существуют следующие способы создания форм:

1. Создание формы с помощью Автоформы. После выбора источника записей и макета формы (в столбец, ленточная или табличная) Автоформа создает форму, которая использует все поля источника записей (в качестве источника данных может быть таблица или запрос).

2. Создание формы с помощью мастера. Мастер задает подробные вопросы об источниках записей, полях, макете, требуемых форматах и создает форму на основании полученных ответов.

3. Создание формы без помощи мастера или в режиме конструктора, где имеется возможность разрабатывать структуру формы, модифицировать ее, изменять структуру данных и внешний вид формы, добавлять различные элементы управления (поле, надпись, переключатель, флажок, список, кнопка, рисунок, линия, прямоугольник и др.)

Макет любой формы состоит из 5 разделов в любом сочетании (заголовок формы, область данных, верхний и нижний колонтитулы, область примечаний).

Форма в целом и ее элементы имеют свойства, которые делятся на несколько групп:

- свойства данных (источник информации для элемента управления; значения по умолчанию, условия контроля ввода и др.);
- свойства макета (размер, цвет, координаты в форме, шрифт, видимость и т.п.);
- свойства событий, которые позволяют запрограммировать реакцию элемента управления на то или иное событие;
- другие свойства (имя, порядок перехода между элементами управления и др.).

## **Практическая часть**

### **Ход работы**

*Необходимо создать формы, используя базу данных, созданную в лабораторной работе №3.*

#### **Создание автоформы.**



1. Перейдите на вкладку **Формы**.
2. Щелкните по таблице **Договора**, в вкладке **Создание** нажмите кнопку **Форма**.
3. Сохраните форму под именем – *Автоформа*.

#### **Создание формы с помощью мастера.**

1. Перейдите на вкладку **Формы**.
2. Щелкните по кнопке **Создать** и выберите режим **Мастер форм**. В этом же диалоговом окне выберите в качестве источника данных таблицу **Агенты**.
3. Используя клавишу « >> », перенесите все поля данной таблицы из окна **Доступные поля** в окно **Выбранные поля**, и перейдите к следующему диалоговому окну.

4. В нем установите флажок напротив опции **в один столбец** и перейдите к следующему диалоговому окну.
5. Выберите в нем оформление и перейдите к следующему диалоговому окну.
6. В последнем окне задайте имя форме *Мастер* и нажмите на кнопку **Готово**.

### **Режим конструктора**

1. Сделайте копию формы *Мастер*. Для этого:
  - 1.1. в меню **Правка** выберите команду **Копировать**;
  - 1.2. в меню **Правка** выберите команду **Вставить**;
  - 1.3. в появившемся диалоговом окне, задайте имя **Копия**.
2. Откройте форму *Копия* в режиме конструктора.
3. Увеличьте область заголовка формы и примечания путем наведения мыши на границы соответствующих разделов и их растягивания/сжимания.
4. Измените фоновый рисунок, для этого щелкаем по кнопке *Страница свойств* на панели инструментов и на вкладке *Все* в строке *Рисунок* выбираем путь и имя графического файла.
5. В область заголовка помещаем с помощью кнопки «Аа» на панели элементов надпись *Информация об агентах*. Форматируем ее любыми способами (изменяем цвет шрифта, размер шрифта и т.д.)
6. Аналогичным способом в область примечания размещаем дату создания формы.
7. В область примечания с помощью панели элементов помещаем кнопку, которая будет закрывать форму (категория – Работа с формой, действие – Закрывать форму).
8. В область данных с помощью кнопки  на панели элементов помещаем подчиненную форму на основе имеющихся таблиц Договора (поля Дата и Количество) и Автосклад (поле МаркаАвто). Название подчиненной формы – *Количество продаж совершенных агентом*.
9. Открываем таблицу *Агенты* в конструкторе и добавляем поле Стаж работы (тип данных – Числовой). Открываем таблицу *Агенты* в режиме таблицы и заполняем это поле числами 1, 2,3 (1- стаж до 5 лет, 2- стаж от 5 до 10 лет, 3- стаж более 10 лет).
10. Открываем форму *Копия* в режиме конструктора и с помощью кнопки  на панели элементов в область данных добавляем группу переключателей. В появившемся диалоговом окне задаем подписи для каждого переключателя: до 5 лет; от 5 до 10 лет; более 10 лет. Переходим к следующему диалоговому окну.
11. В нем выбираем опцию Нет и переходим к следующему диалоговому окну.
12. В нем оставляем все по умолчанию и переходим дальше.
13. В нем выбираем опцию Сохранить значение в поле Стаж работы и переходим к следующему диалоговому окну.
14. Выберите в нем оформление и перейдите к следующему диалоговому окну.

15. В последнем окне задайте подпись для группы переключателей *Стаж работы* и нажмите на кнопку **Готово**.

### **Создание кнопочной формы**

1. В меню **Работа с базами данных** выберите команду **Диспетчер кнопочных форм**.
2. Подтвердите ее создание (*на экране появилось диалоговое окно, в котором высветилось название кнопочной формы, данное ей по умолчанию*).
3. Для изменения данной кнопочной формы щелкните по кнопке **Изменить**.
4. В следующем диалоговом окне, в верхнем поле, замените название кнопочной формы на название базы данных **«Автосалон»** (*у вас есть кнопочная форма, но нет кнопок*).
5. Щелкните по кнопке **Создать**.
6. В появившемся окне (**Изменение элемента кнопочной формы**), выберите, используя кнопки выпадающего списка: команду **Открыть форму для добавления**; форму **Копия** и задайте (в верхнем поле) имя создаваемой кнопки **Информация об агентах**. Нажмите кнопку **ОК**.
7. Щелкните еще раз по кнопке **Создать**.
8. Убедитесь, что в поле команда нет команды открытия запроса.
9. Нажмите клавишу отмены.
10. Закройте окна кнопочной формы.
11. Перейдите на вкладку **Формы**.
12. Откройте кнопочную форму **Switchboard**.
13. Убедитесь, что созданная вами кнопка работает.
14. Закройте кнопочную форму.

### **Создание макроса, открывающего запрос**

1. Перейдите на вкладку **Макросы**.
2. Выполните команду **Создать**.
3. В появившемся диалоговом окне, в поле **Макрокоманда**, используя кнопку вызова списка, выберите команду **Открыть запрос**.
4. В поле **Аргументы макрокоманды** выберите имя открываемого запроса – **«Полный список договоров»**, его режим (**Таблица**), и режим данных (**Только чтение**).
5. Закройте макрос, сохранив его под именем **Вызов**.

### **Создание кнопочной команды, запускающей макрос**

1. Выполните команду **Диспетчер кнопочных форм** из списка **Служебные программы** меню **Сервис**.
2. В появившемся диалоговом окне, нажмите кнопку **Изменить**.
3. В следующем окне выполните команду **Создать**.
4. В появившемся окне (**Изменение элемента кнопочной формы**), выберите, используя кнопки выпадающего списка: команду **Выполнить макрос**; макрос **Вызов** и задайте (в верхнем поле) имя создаваемой кнопки **Список договоров**. Нажмите кнопку **ОК**.
5. Закройте окна кнопочной формы.

6. Убедитесь, что созданная вами кнопка работает.

### **Изменение параметров запуска**

Теперь наша задача сделать недоступным, при открытии файла, окно базы данных, обеспечив, при этом, автоматический запуск кнопочной формы. Для этого необходимо проделать следующие действия:

1. Выполнить команду **Параметры** из меню **Файл**.
2. В появившемся диалоговом окне выбрать **Текущая база данных**, в поле **Форма просмотра** выставить кнопочную форму. Нажмите кнопку ОК.
3. Закройте базу данных Автосалон и откройте заново.
4. Убедившись что, автоматически запускается ваша кнопочная форма, верните параметры запуска в исходное положение.

### **КОНТРОЛЬНЫЕ ВОПРОСЫ:**

1. Для чего служит структура **Формы**?
2. Как в форме осуществляется переход между записями?
3. С какой целью создаются экранные формы?
4. Какие разновидности экранных форм допускаются в ACCESS?
5. Что может быть источником данных для экранной формы?
6. Из каких разделов и элементов может состоять экранная форма?
7. Какие элементы форм называют связанными и несвязанными?
8. Какие элементы можно вставлять в форму в режиме конструктора?
9. Как задать свойства формы и какие из них обязательны?
10. В каких случаях применяют составную форму?
11. Каковы условия правильной взаимосвязи главной и подчиненной форм?

### **ЗАДАНИЕ**

Создать указанные формы в базе данных, созданной в лабораторной работе №4, согласно варианту задания. Созданные формы должны быть отформатированы.

### **ВАРИАНТЫ ЗАДАНИЙ**

#### ***Вариант №1.***

- 1) Создать автоформу в столбец на основе таблицы “Товары”. Название формы - «Автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Магазины”, “Товары”, “Поставщики”, используя поля “Дата заказа”, “Срок поставки (в днях)”, “Тип товара”, “Цена товара”. Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую форму «Автоформа» для добавления и форму «С помощью мастера» для изменения.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой с помощью



созданных кнопок будут осуществляться переходы по записям. Название формы – «Конструктор».

### **Вариант №2.**

- 1) Создать табличную автоформу на основе таблицы “Билеты”. Название формы - «Автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Самолеты”, “Пассажиры”, “Билеты”, используя поля “Маршрут”, “Дата вылета”, “Фамилия”, “Дальность полета”. Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую запрос «Данные аэропорта» только для чтения и таблицу «Билеты» для изменения.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой с помощью созданных кнопок будут запускаться приложения Ms Excel, Ms Word и блокнот. Название формы – «Конструктор».

### **Вариант №3.**

- 1) Создать ленточную форму на основе таблицы “Поезда”. Название формы – «Автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Поезда”, “Билеты”, “Пассажиры”, используя поля “Пункт отправления”, “Время отправления”, “Фамилия”, “№ паспорта”. Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую таблицу “Пассажиры” только для чтения и форму «С помощью мастера» для добавления.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой разместить группу переключателей (подписи для каждого переключателя: плацкартный, купе, общий). Для создания группы переключателей в таблицу «Билеты» добавить поле «Тип вагона». Название формы – «Конструктор».

### **Вариант №4.**

- 1) Создать автоформу в столбец на основе таблицы “Читатели”. Название формы – «Автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Читатели”, “Книги”, “Заказы”, используя поля “Дата заказа”, “Дата возврата”, “Автор (фамилия)”, «Название», «Фамилия». Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую форму «С помощью мастера» для добавления и запрос «Исходные данные библиотеки» только для чтения.
- 4) Создать форму с помощью конструктора, используя поля из таблицы «Читатели», в которой поместить подчиненную форму, созданную на

основе таблиц «Заказы» и «Книги». Взаимосвязь с подчиненной формой будет по полю «Код читателя».

**Вариант №5.**

- 1) Создать табличную автоформу на основе таблицы “Сведения о поступивших заказах”. Название формы – «Автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Сведения о поступивших заказах”, “Сведения о товарах”, “Сведения о клиентах”, используя поля «Название фирмы», “Дата заказа”, “Дата исполнения”, «Наименование», “Количество”, “Стоимость единицы продукции”. Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую таблицу «Сведения о товарах» для добавления данных и позволяющую выйти из приложения.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой по максимуму применить средства форматирования.

**Вариант №6.**

- 1) Создать ленточную форму на основе таблицы “Сведения о командировках”. Название формы – «Автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Сведения о командировках”, “Сведения о филиалах”, “Сведения о сотрудниках”, используя поля “Дата прибытия”, “Дата убытия”, “Месторасположение (город)”, “Фамилия”. Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую форму «С помощью мастера» для добавления и таблицу “Сведения о сотрудниках” только для чтения.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой расположить кнопки, позволяющие удалить запись и запустить блокнот. Форму отформатировать.

**Вариант №7.**

- 1) Создать автоформу в столбец на основе таблицы “Дисциплины”. Название формы – «Автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Студенты”, “Преподаватели”, “Дисциплины”, “Оценки”, используя поля «Фамилия студента», «№ зачетной книжки», “Фамилия преподавателя”, “Наименование дисциплины”, “Количество часов”, «Оценка». Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую только для чтения запросы «Список дисциплин» и «Параметрический запрос» в режиме просмотра.
- 4) Создать форму с помощью конструктора, которая будет представлять собой набор из двух вкладок. На первой вкладке должна выводиться общая информация о студенте, на второй - информация о его успеваемости.

### ***Вариант №8.***

- 1) Создать табличную автоформу на основе таблицы “Тарифы”. Название формы – «Табличная автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Абоненты”, “Тарифы”, “Расчеты за месяц”, используя поля “Фамилия”, “Наименование тарифа”, “Абонентская плата”, “Стоимость минуты разговора”, “Продолжительность разговоров”. Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую форму «Табличная автоформа» для добавления и позволяющую выйти из приложения.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой с помощью созданных кнопок будут осуществляться переходы по записям. Название формы – «Конструктор».

### ***Вариант №9.***

- 1) Создать ленточную форму на основе таблицы “Услуги”. Название формы – «Ленточная автоформа».
- 2) Создать форму с помощью мастера на основе таблиц “Клиенты”, “Услуги”, “Заказы”, используя поля “Фамилия”, “Наименование услуги”, “Стоимость услуги”, “Дата обращения”, “Дата выполнения заказа”. Название формы - «Мастер».
- 3) Создать кнопочную форму, открывающую для добавления форму «Ленточная автоформа» и форму «Мастер» для изменения.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой с помощью созданных кнопок будут запускаться приложения Ms Word, Ms Excel и блокнот. Название формы – «Конструктор». Форму отформатировать.

### ***Вариант №10.***

- 1) Создать автоформу в столбец на основе таблицы “Экспорт Республики Беларусь”. Название формы – «Автоформа в столбец».
- 2) Создать форму с помощью мастера на основе таблиц “Информация о странах”, “Информация о товарах”, “Экспорт Республики Беларусь”, используя поля “Страна”, “Столица”, “Наименование”, “Количество экспортируемого товара”. Название формы - «С помощью мастера».
- 3) Создать кнопочную форму, открывающую таблицу “Информация о странах” для изменения и таблицу “Информация о товарах” для добавления.
- 4) Создать форму с помощью конструктора, используя поля из таблицы «Информация о странах», в которой поместить подчиненную форму, созданную на основе таблиц «Информация о товарах» и «Экспорт Республики Беларусь». Взаимосвязь с подчиненной формой будет по полю «Код страны».

### **Вариант №11.**

- 1) Создать табличную автоформу на основе таблицы “Автомобили”. Название формы – “Автоформа”.
- 2) Создать форму с помощью мастера на основе таблиц “Автомобили”, “Покупатели”, “Сведения о продажах”, используя поля “Дата”, “Фамилия”, “Марка”, “Модель”, “Пробег”, “Стоимость”. Название формы – «Мастер».
- 3) Создать кнопочную форму, открывающую для добавления форму «Мастер» и для изменения запрос “Полный список автомобилей и покупателей” в режиме таблицы.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой разместить группу переключателей (подписи для каждого переключателя: 1,5 года, 2,5 года, 3 года). Для создания группы переключателей в таблицу «Автомобили» добавить поле «Послепродажное обслуживание». Название формы – «Конструктор».

### **Вариант №12.**

- 1) Создать ленточную автоформу на основе таблицы “Услуги”. Название формы – “Ленточная автоформа”.
- 2) Создать форму с помощью мастера на основе таблиц “Услуги”, “Клиенты”, “Работа компьютерного клуба”, используя поля “Фамилия”, “Наименование услуги”, “Стоимость услуги за 1 час работы”, “Дата”, “Время работы”. Название формы – «С помощью мастера».
- 3) Создать кнопочную форму, открывающую таблицу «Клиенты» для добавления данных и позволяющую выйти из приложения.
- 4) Создать форму с помощью конструктора, используя поля из исходных таблиц или запросов (выбрать самостоятельно), в которой по максимуму применить средства форматирования.

## **Лабораторная работа № 5.**

### **Подготовка отчетов средствами MS Access**

#### **Цель работы:**

*Познакомиться с объектами БД Ms Access – отчетами; научиться создавать форматированные отчеты для распечатки на принтере; приобрести опыт по использованию методов сортировки и группировки данных в отчетах; получению требуемой детализации данных в конечном документе.*

#### **Теоретическая часть**

Отчет является эффективным способом представления данных в виде печатного документа. Отчеты по сравнению с формами предоставляют пользователю больше возможностей в изображении и группировки данных, подсчете итоговых значений.

Источником данных для отчета, как и для формы, является таблица или запрос БД. Запрос используют, когда нужно отсортировать данные или включить в отчет информацию из нескольких таблиц.

Существует три режима работы с отчетом: конструктор, предварительный просмотр (вывод на экран в том виде, как отчет будет напечатан), просмотр образца (для предварительной оценки макета). Вывод на печать возможен из любого режима.

Макет отчета состоит из следующих разделов:

- заголовок документа,
- заголовок и подвал каждой страницы (верхний и нижний колонтитулы),
- заголовки групп данных,
- содержательную часть (область данных),
- итоговую часть (примечания),
- заключительную часть всего отчета (область примечаний отчета).

В каждом разделе можно разместить элементы управления, состав которых тот же, что и при разработке форм.

## **Практическая часть**

### **Ход работы**

*Необходимо создать отчеты, используя базу данных, созданную в лабораторной работе №4.*

#### **Создание автоотчета.**

1. Щелкните по таблице **Автосклад**, на вкладке *Создание* нажмите кнопку **Отчёт**.
2. Сохраните отчет под именем – *Автосклад\_отчет*.

#### **Создание отчета с помощью мастера.**

1. Выберите режим **Мастер отчетов**. В этом же диалоговом окне в качестве источника данных таблицу **Договора**.
2. Используя клавишу « >> », перенесите все поля данной таблицы из окна **Доступные поля** в окно **Выбранные поля**, и перейдите к следующему диалоговому окну.
3. В нем добавьте уровень группировки – поле **Дата**, первоначально переместив поле **КодАгента** назад « < ». В этом же диалоговом окне нажмите на кнопку **Группировка...** и задайте интервал группировки по кварталам. Перейдите к следующему диалоговому окну.
4. В нем выберите поле **Количество** и нажмите на кнопку **Итоги...**, где отметьте галочкой **Sum**. Перейдите к следующему диалоговому окну.
5. В нем установите флажок напротив опции **блок** и перейдите к следующему диалоговому окну.
6. Выберите в нем оформление и перейдите к следующему диалоговому окну.
7. В последнем окне задайте имя отчету *Мастер\_отчет* и нажмите на кнопку **Готово**.

### **Создание диаграммы.**

1. Выберите **Мастер диаграмм**. В этом же диалоговом окне укажите в качестве источника данных таблицу **Автосклад**. Нажмите кнопку ОК.

2. Используя виртуальную клавишу « > », перенесите поля *МаркаАвто* и *Цена* из окна **Доступные поля** в окно **Поля диаграммы**, и перейдите к следующему диалоговому окну.

3. Выберите тип диаграммы – **Объемная гистограмма**.

4. В следующем окне проверьте, чтобы по оси X было поле *МаркаАвто*, а по оси Y - *Цена*. Первоначально по оси Y будет *Сумма\_цена*, измените ее на *Цену* (двойной щелчок левой кнопкой мыши по виртуальной кнопке *Сумма\_цена* и выберите в появившемся окне, что итоги для «Цена» - Отсутствует). Перейдите к следующему диалоговому окну.

5. В нем задайте имя диаграмме *Цена авто* и выберите опцию *Изменение структуры отчета или диаграммы*. Нажмите на кнопку **Готово**.

6. Увеличьте ее линейные размеры по ширине чистого листа. Выделите у диаграммы ось категорий, с помощью правой кнопки мыши выполните команду **Формат оси**. На вкладке **Шрифт** измените размер шрифта на 10, на вкладке **Выравнивание** измените ориентацию на 90 градусов. Нажмите кнопку ОК. Сохраните отчет под названием *Диаграмма*.

### **Создание почтовых наклеек.**

1. Перейдите на вкладку **Создание** и выберите **Почтовые наклейки**. В этом же диалоговом окне укажите в качестве источника данных таблицу **Агенты**. Нажмите кнопку ОК.

2. Выберите размер наклейки – 83,8 mm × 42,3 mm, и перейдите к следующему диалоговому окну.

3. Выберите шрифт - Times New Roman, размер шрифта – 9, начертание – курсив.

4. В следующем диалоговом окне выберите поля **ФИО** и **Телефон** и оформите прототип наклейки, как визитку. Перейдите к следующему диалоговому окну.

5. В нем выберите поле для сортировки – **ФИО** и перейдите к следующему диалоговому окну.

6. В последнем окне задайте имя отчету *Наклейки* и нажмите на кнопку **Готово**.

### **КОНТРОЛЬНЫЕ ВОПРОСЫ:**

1. Что такое отчеты?
2. Какие возможности предоставляет пользователю генератор отчетов СУБД ACCESS?
3. Каковы режимы работы с отчетом?
4. Что такое макет отчета и какова его структура?
5. Можно ли выводить в документ данные, явно несодержащиеся в таблицах базы данных?

6. Как составить отчет, содержащий информацию из нескольких таблиц БД?  
7. Каковы возможности группировки данных в отчете?

## **ЗАДАНИЕ**

Создать указанные отчеты в базе данных, созданной в лабораторной работе №4, согласно варианту задания.

### **ВАРИАНТЫ ЗАДАНИЙ**

#### ***Вариант №1.***

1) Создать ленточный автоотчет на основе таблицы “Товары”.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Товары* и все поля, кроме *код товара*;
- ✓ добавить уровень группировки - *Тип товара*;
- ✓ выбрать поле для сортировки *Цена товара*, порядок сортировки *По возрастанию*; вид макета *блок* и стиль *Спокойный*;

Имя отчета - *Вариант 1.*

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Поставщики*, по оси X расположить *Название фирмы*, по оси Y - *сроки поставки (в днях)*. Тип диаграммы - цилиндрическая. Название отчета - *Поставка товара*.

#### ***Вариант №2.***

1) Создать автоотчет в столбец на основе таблицы *Самолеты*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Самолеты* и все поля, кроме *код т/с*;
- ✓ добавить уровень группировки - *Тип самолета*;
- ✓ выбрать поле для сортировки *Время вылета*, порядок сортировки *По убыванию*; вид макета *Структура 1* и стиль *полужирный*.

Имя отчета - *Вариант 2.*

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Билеты*, по оси X расположить *Паспортные данные*, по оси Y - *Стоимость*. Тип диаграммы - объемная гистограмма. Название отчета - *Аэропорт*.

#### ***Вариант №3.***

1) Создать ленточный автоотчет на основе таблицы *Пассажиры*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Пассажиры* и все поля, кроме *код пассажира*;
- ✓ выбрать поле для сортировки *фамилия*, порядок сортировки *По возрастанию*; вид макета *блок* и стиль *Спокойный*

Имя отчета - *Вариант 3.*

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Билеты*, по оси X расположить *Пункт назначения*, по оси Y - Стоимость проезда. Тип диаграммы - объемная гистограмма. Название диаграммы - Железнодорожный вокзал.

**Вариант №4.**

1) Создать автоотчет в столбец на основе таблицы *Читатели*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Читатели* и все поля, кроме *домашний адрес*;
- ✓ выбрать поле для сортировки *код читателя*, порядок сортировки *По убыванию*; вид макета *Структура 1* и стиль *полужирный*.

Имя отчета - *Вариант 4*.

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Книги*, по оси X расположить *Название*, по оси Y - Стоимость. Тип диаграммы - цилиндрическая. Имя отчета - Библиотека.

**Вариант №5.**

1) Создать ленточный автоотчет на основе таблицы *Сведения о клиентах*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Сведения о клиентах* и все поля, кроме *код клиента*;
- ✓ выбрать поле для сортировки *название фирмы*, порядок сортировки *По возрастанию*; вид макета *блок* и стиль *Спокойный*.

Имя отчета - *Вариант 5*.

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Сведения о товарах*, по оси X расположить *наименование*, по оси Y стоимость единицы продукции. Тип диаграммы - цилиндрическая. Имя отчета - Овощной магазин.

**Вариант №6.**

1) Создать автоотчет в столбец на основе таблицы *Сведения о сотрудниках*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Сведения о сотрудниках* и все поля, кроме *отчество*;
- ✓ добавить уровень группировки *должность*;
- ✓ выбрать поле для сортировки *оклад*, порядок сортировки *По убыванию*, вид макета *Структура 1* и стиль *полужирный*;

Имя отчета - *Вариант 6*.

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Сведения о командировках*, по оси X расположить *Личный номер сотрудника*, по оси Y - *Аванс*. Тип диаграммы - объемная гистограмма. Имя отчета - Командировки.

**Вариант №7.**

1) Создать ленточный автоотчет на основе таблицы *Студенты*.



2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Студенты* и все поля, кроме *отчество студента*;
- ✓ добавить уровень группировки *№ группы*;
- ✓ выбрать поле для сортировки *фамилия студента*, порядок сортировки *По возрастанию*, вид макета *блок* и стиль *Спокойный*;

Имя отчета - *Вариант 7*.

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Оценки*, по оси X расположить *№ зачетной книжки*, по оси Y - оценка. Тип диаграммы - цилиндрическая. Имя отчета - Деканат.

#### **Вариант №8.**

1) Создать автоотчет в столбец на основе таблицы *Абоненты*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Абоненты* и все поля, кроме *отчество*;
- ✓ добавить уровень группировки *код тарифа*;
- ✓ выбрать поле для сортировки *телефонный номер абонента*, порядок сортировки *По возрастанию*, вид макета *Структура 1* и стиль *полужирный*.

Имя отчета - *Вариант 8*.

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Тарифы*, по оси X расположить поле наименование тарифа, по оси Y - стоимость минуты разговора. Тип диаграммы – объемная гистограмма. Имя отчета - Оператор сотовой связи.

#### **Вариант №9.**

1) Создать ленточный автоотчет на основе таблицы *Клиенты*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Клиенты* и все поля, кроме *адреса*;
- ✓ выбрать поле для сортировки *Фамилия*, порядок сортировки *По возрастанию*, вид макета *блок* и стиль *Спокойный*.

Имя отчета - *Вариант 9*.

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Услуги*, по оси X расположить поле наименование услуги, по оси Y - стоимость услуги. Тип диаграммы - цилиндрическая. Имя отчета - Химчистка.

#### **Вариант №10.**

1) Создать автоотчет в столбец на основе таблицы *Информация о странах*.

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Информация о странах* и все поля, кроме *денежные единицы*;
- ✓ добавить уровень группировки *официальный язык*;
- ✓ выбрать поле для сортировки *страна*, порядок сортировки *По возрастанию*; вид макета *Структура 1* и стиль *полужирный*.

Имя отчета - *Вариант 10.*

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Информация о товарах*, по оси X расположить поле *наименование*, по оси Y - *стоимость единицы продукции в Br.* Тип диаграммы - цилиндрическая. Имя отчета - Торговая политика.

**Вариант №11.**

1) Создать ленточный автоотчет на основе таблицы *Автомобили.*

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблицу *Автомобили* и все поля, кроме *Код авто*;
- ✓ добавить уровень группировки *марка*;
- ✓ выбрать поле для сортировки *стоимость*, порядок сортировки *По возрастанию*, вид макета *блок* и стиль *Спокойный*.

Имя отчета - *Вариант 11.*

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Автомобили*, по оси X расположить поле *модель*, по оси Y - *объем двигателя*. Тип диаграммы – *объемная гистограмма*. Имя отчета - Автосалон.

**Вариант №12.**

1) Создать автоотчет в столбец на основе таблицы *Клиенты.*

2) Создать отчет с помощью мастера. При этом:

- ✓ в диалоговом окне *Создание отчетов* выбрать таблица *Клиенты* и все поля, кроме *адрес*;
- ✓ выбрать поле для сортировки *фамилия*, порядок сортировки *По убыванию*; вид макета *Структура 1* и стиль *полужирный*;

Имя отчета - *Вариант 12.*

3) С помощью мастера диаграмм построить диаграмму на основе данных таблицы *Услуги*, по оси X расположить поле *наименование услуги*, по оси Y - *стоимость услуги за 1 час работы*. Тип диаграммы - цилиндрическая. Имя отчета - Компьютерный клуб.

## **Лабораторная работа № 6. Импорт данных из MS Excel**

### **Практическая часть**

#### **Ход работы**

*Разработать проект базы данных, выбрав вариант по первой букве фамилии.*

### **ВАРИАНТЫ ЗАДАНИЙ**

•Б, Р, Ц, Я, Л

[Банковские вклады](#)

•В, У, Ф, Ч, Т

[Библиотека](#)

•Д, Х, Ш, М, П

### Коммунальные платежи

•Е, К, Ж, Г

### Телефонные переговоры

•А, Э, Ю, Щ, И

### Торговые операции

•З, О, С, Н

### Фотосервис

#### **Часть 1**

1. Откройте свой вариант в MS EXCEL. Зарисуйте таблицы, поля, ключи и связи.
2. Запустите MS Access. Создайте новую БД под именем "№\_группы\_Фамилия".
3. Импортируйте в базу таблицу из MS EXCEL.
4. Не создавайте автоматически ключевое поле (счетчик).
5. Скопируйте таблицу столько раз, сколько будет таблиц в БД (оставить еще одну для страховки – ИСХОДНАЯ), присвойте таблицам соответствующие имена (например, АБОНЕНТЫ или ЧИТАТЕЛИ).
6. В каждой таблице – удалите лишние атрибуты.
7. Отсортируйте по предполагаемому ключу.
8. Удалите повторяющиеся записи.
9. Установите ключевые поля.
10. Создайте схему данных. В схеме должны присутствовать только связи 1-М.
11. В самой нижней (учетной) таблице – сохраняются все записи (но не все атрибуты), ключ можно не задавать.

#### **Часть 2**

1. Добавьте описания к некоторым таблицам (правый клик – Свойства). Просмотрите таблицы в режиме таблицы.
2. Просмотрите зависимости объектов для каждой из таблиц. Какую таблицу можно было бы безболезненно удалить из базы данных?
3. Для отдельных полей (на свое усмотрение) задайте новые свойства:
  - описание
  - подпись
  - формат поля (осторожно с ключевыми полями)
  - маска ввода (лучше задавать для телефонов и кодов)
  - условие на значение (например, >100 And <900)
  - сообщение об ошибке (работает, если нарушено условие на значение)
4. Откройте любую главную таблицу. Обратите внимание на + в первом столбце. При раскрытии этой структуры можно просмотреть все связанные записи подчиненной таблицы.

### **КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Цель применения индексирования в таблицах. Плюсы и минусы.
2. Средства поддержания целостности данных в MS Access.

3. Для каких целей используется маска ввода?
4. Для каких целей используется условие на значение?
5. Когда появляется сообщение об ошибке?
6. Чем отличается подпись от описания?
7. Назначение первичного ключа. Его свойства.
8. Назначение внешнего ключа. Его свойства.
9. Возможна ли таблица без первичного ключа?
10. Должно ли быть соответствие между первичным и внешним ключом?
11. Перечислите возможные действия в конструкторе таблиц:
  - ввод данных;
  - задание имен полей;
  - задание типов полей;
  - задание прочих свойств поля;
  - установка первичного ключа;
  - сортировка;
  - фильтрация
12. Перечислите возможные действия в режиме таблицы (см. п. 11).
13. Опишите работу с фильтром.
14. Назначение подстановок в таблице.
15. Как из одной таблицы отобразить связанные записи другой таблицы?
16. Перечислите способы создания таблиц в MS Access (5).
17. Виды связей в реляционной модели.
18. Способы объединения записей для связи-объединения.
19. Какая таблица считается главной? подчиненной?
20. Можно ли обеспечить уникальность значения для неключевого поля?
21. Можно ли добавить комментарий к целой таблице?
22. Как перед удалением лишней таблицы убедиться в отсутствии связей?
23. Как автоматизировать нормализацию отдельной таблицы?
24. Можно ли распечатать информацию о структуре таблицы (свойства всех полей)?
25. Можно ли обеспечить для конкретного поля ввод данных только из ограниченного набора значений?
26. При создании новой таблицы каким лучше сделать размер текстового поля – максимальный или минимальный?

## **Лабораторная работа № 7. Запросы в конструкторе**

### **Практическая часть**

#### **Ход работы**

*Присваивайте запросам имена по номеру задания (например, Запрос 15).  
В работе используется следующая нумерация вариантов работ:*

- a. *Банковские вклады.*
- b. *Библиотека.*
- c. *Коммунальные платежи.*
- d. *Телефонные переговоры.*
- e. *Торговые операции.*
- f. *Фотосервис.*

### **Часть 1.**

1. Создайте простой запрос на выборку:
  - a. Информация обо всех вкладах с размером годового процента  $>10$ , сделанных в июне или в августе.
  - b. Информация обо всех книгах, изданных в 1998 году и количеством страниц  $> 470$ .
  - c. Информация обо всех платежах за газ, сделанных до 15 июня.
  - d. Информация обо всех переговорах, продолжительностью  $> 20$  минут.
  - e. Информация обо всех продажах, если название товара начинается на буквы "М" или "С", и дата позже 19/08/04.
  - f. Информация обо всех заказах на печать фотографий.
2. Запрос с параметром:
  - a. Информация обо всех вкладах в определенной валюте.
  - b. Информация обо всех книгах, в названии которых есть ключевое слово.
  - c. Информация обо всех платежах, сделанных жильцами определенной квартиры (№ квартиры вводится параметром).
  - d. Информация обо всех звонках, сделанных в определенную страну (название страны вводится параметром).
  - e. Информация обо всех продажах по цене указанной пользователем.
  - f. Информация обо всех заказах на дату, указанную пользователем.
3. Запрос с параметром и выборкой:
  - a. Информация обо всех вкладах, которые делали вкладчики в месяц своего рождения (месяц задается параметром).
  - b. Информация обо всех книгах, которые читатели задержали дольше определенного срока (к-во дней задается параметром).
  - c. Информация обо всех платежах, сделанных позже определенного дня месяца (день вводится параметром).
  - d. Информация обо всех переговорах, тариф которых больше определенной величины.
  - e. Информация обо всех продажах, выполненных в определенный месяц.
  - f. Информация обо всех заказах в июне, сделанных клиентами, чья фамилия начинается определенным образом.
4. Вычисляемый запрос:
  - a. Определить процентное содержание от начальной суммы вклада в конце минимального срока (проценты начисляются 1 раз в год).

- b. Определить количество страниц, прочитанных в день каждым читателем по каждой книге.
  - c. Определить сумму оплаты по каждому платежу.
  - d. Определить стоимость каждого звонка.
  - e. Определить сумму каждой продажи.
  - f. Определить стоимость каждого заказа.
5. Вычисляемый запрос:
- a. Определить возраст каждого вкладчика на текущую дату.
  - b. Определить возраст каждого читателя на текущую дату.
  - c. Определить жилплощадь, приходящуюся на каждого жильца.
  - d. Определить стоимость каждого звонка, если ночные звонки бесплатны.
  - e. Определить сумму каждой продажи, если товар, расфасованный в бутылки, отпускается бесплатно.
  - f. Определить средний дневной заработок фотоателье по каждому заказу.
6. Запрос с групповыми операциями:
- a. Количество вкладов по каждому виду валюты.
  - b. Количество книг, прочитанных каждым читателем.
  - c. Определить общую сумму оплаты по каждому квартиросъемщику.
  - d. Суммарная продолжительность переговоров для каждого абонента.
  - e. Суммарное количество продаж по каждой фирме.
  - f. Количество заказов на каждого клиента.
7. Запрос с групповыми операциями:
- a. Количество людей (без повторений), которые делали вклады по каждому виду валюты (Если Иванов делал 3 вклада в USD, считаем его 1 раз).
  - b. Количество книг, прочитанных каждым читателем.
  - c. Определить общее потребление газа всеми жильцами.
  - d. Общая сумма, выплаченная абонентами за переговоры с каждой страной.
  - e. Общая сумма продаж по каждому наименованию товара.
  - f. Общая сумма заказов по каждому виду услуги.
8. Вычисляемый запрос с групповыми операциями:
- a. Средний возраст вкладчиков на момент совершения вклада.
  - b. Средний возраст читателей на момент взятия книги.
  - c. Определить среднее потребление газа на одного жильца.
  - d. Средняя продолжительность разговора для каждой страны.
  - e. Общая сумма выручки за каждый месяц.
  - f. Количество заказов, не выполненных в срок.
9. Сложный запрос:
- a. Самое популярное имя вкладчика и количество вкладчиков с таким именем.

- b. Количество книг с минимальным количеством страниц.
- c. Количество квартир с максимальным количеством жильцов.
- d. Количество разговоров с самой популярной страной.
- e. Количество продаж по самым популярным товарам.
- f. Количество заказов по самой популярной услуге.

10. Перекрестный запрос:

- a. Количество вкладчиков по каждому наименованию вклада и каждому виду валюты.
- b. Количество книг по каждому читателю и каждому издательству.
- c. Сумма оплаты по каждой услуге каждым жильцом.
- d. Суммарная стоимость переговоров по каждому абоненту и каждой стране.
- e. Сумма выручки по каждому товару и каждой фирме.
- f. Сумма заказов по каждой услуге за каждый месяц.

11. Перекрестный запрос. Добавьте итоговый результат по каждой строке предыдущего запроса.

**Часть 2.**

12. Записи без подчиненных.

- a. Добавьте новые виды вкладов. Создайте запрос 'Записи без подчиненных' – невостребованные виды вкладов.
- b. Добавьте новые книги. Создайте запрос 'Записи без подчиненных' – невостребованные книги.
- c. Добавьте новые услуги. Создайте запрос 'Записи без подчиненных' – невостребованные услуги.
- d. Добавьте новые страны. Создайте запрос 'Записи без подчиненных' – страны, в которые никто не звонил.
- e. Добавьте новые товары. Создайте запрос 'Записи без подчиненных' – невостребованные товары.
- f. Добавьте новые услуги. Создайте запрос 'Записи без подчиненных' – невостребованные услуги.

13. Запрос на обновление:

- a. Вклад "Пенсионный" переименовать в "Почетный".
- b. Издательство "СПб.: БХВ-Петербург" заменить на "СПб.: ВHV. Санкт-Петербург".
- c. Название услуги "Газ" заменить на "Природный газ".
- d. Страна "Кот д'Ивуар" вернула себе прежнее название ("Берег слоновой кости").
- e. Компания "Павлова Ltd" теперь называется "Пушкарева Ltd".
- f. Услуга "Ретуширование снимка" теперь называется "Художественная обработка фото".

14. Запрос на обновление с вычислением:

- a. Для всех вкладов в евро увеличить годовой процент на 1.25.

- b. Изменить номера читательских билетов: если номер начинается на 1 – заменить его двузначным (125 на 25, 178 на 78 и т.д.).
- c. Тариф на отопление поднять на 15%.
- d. Тариф на звонки в Израиль поднять на 25%.
- e. Цены на рис упали в 2 раза.
- f. Цифровое фото подешевело на 10%.

## **КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Перечислите режимы отображения запросов.
2. Перечислите типы запросов.
3. Как долго хранится таблица, создаваемая запросом на выборку?
4. Как включить в запрос все поля из таблицы?
5. Логическая операция между полями в условии отбора.
6. Логическая операция между строками в условии отбора.
7. Как отобразить записи, в которых по полю «Количество» значения  $> 5$  и  $< 20$ ? (2 способа)
8. В каких строках конструктора можно использовать вычисляемое выражение?
9. Что такое параметр? Как добавить параметр в запрос?
10. Как настроить свойства параметра?
11. Групповые операции – как и для чего применяется?
12. Как в запросе получить дату в определенном формате?
13. Что произойдет, если запрос на создание таблицы попытается создать уже существующую таблицу?
14. Можно ли одним запросом удалить записи из нескольких таблиц?

## **Лабораторная работа № 8 SQL. Запросы на выборку.**

### **Практическая часть**

*Все запросы создаются в SQL. Присваивайте запросам имена по номеру задания (например, SQL 15). В работе используется следующая нумерация вариантов работ:*

- a. Банковские вклады.*
- b. Библиотека.*
- c. Коммунальные платежи.*
- d. Телефонные переговоры.*
- e. Торговые операции.*
- f. Фотосервис.*

### **ЗАДАНИЕ**

1. Простой запрос. Создайте запрос на выборку указанных полей из первоначальной таблицы в ИНФ. Исключите повторяющиеся записи в рабочей таблице.



- a. Название вклада, Валюта, Годовой процент.
  - b. Автор, Название книги, Издательство.
  - c. Фамилия, Жилплощадь, К-во жильцов.
  - d. Код страны, Страна.
  - e. Название фирмы-клиента, Фамилия руководителя.
  - f. Наименование услуги, Срок исполнения.
2. Запрос с вычисляемыми полями.
- a. Для таблицы «Вклады» определить количество дней, прошедших с даты вклада.
  - b. Для таблицы «Книги» определить возраст каждой книги на текущую дату.
  - c. Для таблицы «Жильцы» определить жилплощадь, приходящуюся на одного жильца в каждой квартире.
  - d. Для таблицы «Тарифы» определить стоимость 1 сек переговоров
  - e. Для таблицы «Товары» определить стоимость каждого товара с 20% НДС.
  - f. Для таблицы «Услуги» определить стоимость каждой услуги со скидкой 15%.
3. Запрос с операторами сравнения.
- a. Отобрать вкладчиков с датой рождения не раньше 1980 года.
  - b. Отобрать книги с датой издания не раньше 2000 года.
  - c. Отобрать услуги с кодом < 200.
  - d. Отобрать все переговоры, продолжительностью меньше получаса.
  - e. Отобрать все товары с ценой < 100.
  - f. Отобрать все заказы, где к-во > 5.
4. Запрос с операторами сравнения (используйте запрос SQL2).
- a. Отберите вклады, сделанные не менее 100 дней назад.
  - b. Отберите книги, которым не более 10 лет.
  - c. Отберите квартиры, в которых приходится более 50 кв.м на человека.
  - d. Отберите тарифы со стоимостью 1 сек более 1000.
  - e. Отберите товары с ценой с НДС > 200.
  - f. Отберите услуги, у которых стоимость со скидкой превышает 30 000.
5. Запрос с фразой BETWEEN.
- a. Отберите все вклады с годовым процентом от 9 до 11.
  - b. Отберите книги с количеством страниц от 500 до 100.
  - c. Отберите услуги с кодом от 100 до 200.
  - d. Отберите все переговоры продолжительностью от 20 до 40 минут.
  - e. Отберите все товары с ценой от 100 до 250.
  - f. Отберите все услуги с кодом от 100 до 300.
6. Запрос с фразой IN.
- a. Отберите все вклады с кодом 250 и 801.

- b. Отберите все выдачи книг с кодом 617 и 645.
  - c. Отберите все платежи за услуги с кодом 155 и 164.
  - d. Отберите все звонки в страны с кодом 8--10—225 и 8--10—53.
  - e. Отберите все продажи товаров с кодом 445160 и 620870.
  - f. Отберите все заказы на услуги с кодом 301 и 221.
7. Запрос с фразой LIKE.
- a. Отберите всех вкладчиков, чей паспорт имеет серию OT.
  - b. Отберите книги всех авторов по имени Виктор.
  - c. Отберите все платежи из квартиры 60.
  - d. Отберите все звонки в страны с названием из трех букв.
  - e. Отберите все продажи всех видов риса.
  - f. Отберите все услуги по оцифровке.
8. Запрос с сортировкой. Выполните предыдущий запрос с сортировкой по убыванию. Поле для сортировки выберите самостоятельно.
9. Запрос с фразой TOP. Из предыдущего запроса отобразите 10% записей.
10. Запрос с использованием SQL-функций (без фразы GROUP BY).
- a. Определите количество вкладчиков и их средний возраст.
  - b. Определите количество книг и суммарное количество страниц в них.
  - c. Определите количество квартиросъемщиков и количество жильцов.
  - d. Определите количество звонков и суммарную продолжительность переговоров.
  - e. Определите количество продаж и суммарное количество проданного товара.
  - f. Определите количество предоставляемых услуг и среднюю цену за услугу.
11. Запрос с фразой GROUP BY.
- a. Определите средний годовой процент по каждому виду валюты.
  - b. Определите количество книг по каждому году издания.
  - c. Определите количество платежей по каждому виду услуги.
  - d. Определите среднюю продолжительность разговора по каждому коду страны.
  - e. Определите суммарное количество проданного товара по каждому коду товара.
  - f. Определите суммарное количество заказов по каждому коду услуги (сумма по полю количество).
12. Запрос с фразой GROUP BY. К предыдущему запросу добавьте сортировку по возрастанию и еще один уровень группировки:
- a. по коду вклада.
  - b. по каждому издательству.
  - c. по каждой дате оплаты.
  - d. по каждому абоненту.
  - e. по каждой дате продажи.

- f. по каждой дате заказа.
13. Запрос с фразой HAVING. (Используйте запрос SQL12).
- a. если средний процент не меньше 10.
  - b. если количество не меньше 5.
  - c. если количество меньше 5.
  - d. если средняя продолжительность меньше 30 мин.
  - e. если суммарное количество больше 100.
  - f. если суммарное количество больше 10.

## Лабораторная работа № 9 Проектирование интерфейса.

### Практическая часть

*Ход работы.* В работе используется следующая нумерация вариантов работ:

- a) Банковские вклады.
- b) Библиотека.
- c) Коммунальные платежи.
- d) Телефонные переговоры.
- e) Торговые операции.
- f) Фотосервис.

### Часть 1.

1. Создайте форму на основе таблиц:
  - a. Вкладчики;
  - b. Читатели;
  - c. Жильцы;
  - d. Абоненты;
  - e. Клиенты;
  - f. Клиенты.
2. Создайте форму с помощью мастера на основе таблиц (оформление может быть на Ваше усмотрение):
  - a. Виды вкладов.
  - b. Книги.
  - c. Услуги.
  - d. Тарифы.
  - e. Товары.
  - f. Услуги.
3. Создайте форму с подчиненной **Form 3-1** с помощью мастера:
  - 1) включите поля всех трех таблиц;
  - 2) в качестве главной таблицы используйте ту же, что и в **Form1**;
  - 3) для подчиненной формы используйте ленточный формат;
  - 4) доработайте формы в режиме конструктора – удалите дубликаты полей, настройте размеры так, чтобы все поля одновременно отображались на экране.

5) добавьте на главную форму с помощью мастера пользовательские кнопки: переходы по записям, добавление записи, закрытие формы.

**Macro 1.** Напишите макрос, который при открытии формы Form 3 открывает все три соответствующие таблицы и располагает их по разным углам экрана. Для этого необходимо:

- 1) Зайти в меню *Файл* → *Параметры* → *Текущая база данных*. В *Параметрах* окна документа поставить галочку напротив *Перекрывание окон*. Перезапустить файл MS Access.
- 2) Создать макрос. Добавить макрокоманду Открыть Таблицу, выбрать первую таблицу (например, Абоненты), Режим Таблица, Режим данных Изменение.
- 3) Добавить новую макрокоманду Размер Положение Окна. Ввести необходимые значения.
- 4) Таким образом добавить еще две таблицы. После выполнения задания зайти в меню *Файл* → *Параметры* → *Текущая база данных*. В *Параметрах* окна документа поставить галочку напротив *Вкладки*. Перезапустить файл MS Access.

**Macro 2.** Напишите макрос, который при выполнении определенного условия для текущей записи (например, дата позже 1/01/2020) отображает на форме надпись: "Условие выполнено".

## **Часть 2.**

1. Создайте автоотчет для любой из таблиц Вашей БД. Доработайте его в режиме конструктора: полностью видны все поля и надписи, минимальный размер пустых областей, сортировка по одному из полей по возрастанию и группировка.
2. При помощи мастера создайте сложный отчет:
  - a. Главный – "Вкладчики", подчиненный "Вклады", с информацией о виде каждого вклада.
  - b. Главный – "Читатели", подчиненный "Книги на руках" с информацией о каждой выданной книге.
  - c. Главный – "Жильцы", подчиненный "Платежи" с информацией о каждой оплаченной услуге.
  - d. Главный – "Абоненты", подчиненный "Звонки" с информацией о каждом сделанном звонке.
  - e. Главный – "Клиенты", подчиненный "Продажи" с информацией о каждом проданном товаре.
  - f. Главный – "Клиенты", подчиненный "Заказы" с информацией о каждой заказанной услуге.
3. Сконструируйте отчет, на основе запроса с параметром.

## **Часть 3. Связь приложений.**

1. Организуйте загрузку одной из таблиц в документ MS Excel.
2. В Ms Word сформируйте текст сообщения для рассылки писем. Данные для полей слияния будут взяты из Вашей БД.

## КОНТРОЛЬ ЗНАНИЙ

### *Экзаменационные вопросы*

1. Основные объекты СУБД Ms Access.
2. Типы данных Ms Access.
3. Назначение и виды запросов Ms Access.
4. Назначение и виды форм, способы их создания.
5. Запросы на изменение в Ms Access.
6. Проектирование реляционных СУБД.
7. Понятие и классификация баз данных.
8. Виды связей в реляционной модели базы данных.
9. Подготовка отчетов средствами Ms Access.
10. Анализ данных таблицы с помощью запросов в Ms Access.
11. Проектирование и использование форм в Ms Access.
12. Порядок формирования схемы базы данных.
13. Макросы в Ms Access.
14. Работа с таблицами в Ms Access. Проектирование макета таблицы.  
Режимы создания таблицы.
15. Свойства полей в Ms Access.
16. Запросы с параметром в Ms Access.
17. Работа с формами в Ms Access: режимы работы с формой, способы создания формы, области формы, элементы управления.
18. Разработка отчетов в Ms Access: режимы работы с отчетом, способы создания отчета, области отчета, группировка данных в отчете.
19. Целостность данных в Ms Access.
20. Работа с запросами в Ms Access: режимы работы с запросами, вычисляемые поля в запросах, использование групповых операций.
21. Взаимодействие Ms Access с приложениями Ms Office.
22. СУБД Ms Access: назначение, основные функции и интерфейс.
23. Назначение первичного ключа. Его свойства.
24. Групповые операции в запросах Ms Access.
25. Вычислительная сеть. Ее основные преимущества.
26. Вычислительная сеть. Основные понятия.
27. Интернет – хостинг и Интернет – провайдеры.
28. Менеджер загрузок Download Master. Основные возможности.
29. Навигация в сети WWW, программы-браузеры.
30. Основные источники информации и их размещение в Internet.  
Классификация поисковых систем, стратегия поиска.
31. Основные компоненты локальной сети.
32. Планирование сайтов.
33. Понятие и структура IP-адреса.
34. Признаки классификации компьютерных сетей.

35. Протоколы компьютерных сетей.
36. Публикация Web-сайта.
37. Размещение сайта в Интернете. Классификация Интернет сайтов.  
Недостатки размещения сайтов на бесплатном хостинге.
38. Система доменных имен.
39. Способы подключения к сети Интернет.
40. Средства удержания постоянной аудитории сайта.
41. Топологии локальных вычислительных сетей.
42. Торговые системы Internet: понятие и классификация торговых систем Internet. Internet-магазины (понятие и функции Internet-магазина, способы оплаты товаров, способы доставки товаров, требования к организации Internet-магазина).
43. Трафик. Основные характеристики трафика.
44. Язык HTML: назначение, структура HTML-документа, основные тэги.

### *Задание для выполнения курсовой работы*

Курсовая работа нацелена на проектирование в теории и создание на практике базы данных определенной предметной области в системе управления базами данных.

Разработка базы данных (БД) на основе реляционных моделей данных требует, с одной стороны, знания предметной области, а с другой, владения современными информационными технологиями.

У каждого студента определенная предметная область. Возможные варианты предметных областей приведены ниже:

- «Банковские операции»;
- «Услуги автосервиса»;
- «Справочная ГИБДД»;
- «Реализация проездных документов в городском пассажирском транспорте»;
- «Импорт товаров»;
- «Экспорт товаров»;
- «Грузовые перевозки автомобильным транспортом»;
- «Пассажирские перевозки автомобильным транспортом»;
- «Пассажирские перевозки железнодорожным транспортом»;
- «Пассажирские авиаперевозки»;
- «Интернет-магазин»;
- «Библиотека»;
- «Деканат»;
- «Услуги Интернет-провайдера».

Студенту предоставляется по согласованию возможность выбора предметной области.

Индивидуальное задание включает следующие элементы: проектирование БД; создание БД и занесение в нее данных; организация запросов к базе; создание экранных форм; оформление отчетов; создание кнопочной формы, объединяющей созданные объекты в единое приложение.

Работа состоит из выполнения следующих заданий:

1. Необходимо спроектировать БД, содержащую не менее трех таблиц, используя метод “сущность-связь”.
2. Создать БД и занести в нее данные (не менее 50 записей в каждой таблице). Использовать свойства полей. Оценивается многообразие типов полей.
3. Организовать связи между таблицами. Убедиться, что: данные, внесенные в таблицы непротиворечивы; система поддержки целостности БД функционирует.
4. Организовать следующие типы запросов к БД:
  - выборки;
  - итоговые;

- с параметрами;
- перекрестные;
- запросы действия.

В запросах использовать создание новых полей, условия выборки, свойства полей, их форматирование.

5. Создать экранные формы для ввода данных, их просмотра и редактирования.
6. Оформить отчеты. Создать вычисляемые поля. Использовать операции группировки. Отчеты должны иметь заголовок, заголовок группы, расчетную строку по группе, итоговую строку по отчету. Использовать вложенные группы.
8. Создать кнопочную форму. На форму вывести рисунок-эмблему предприятия, подготовленный в графическом редакторе.
9. Использовать в работе макросы, программы на VBA.

Содержание библиографического списка определяет сам студент, исходя из цели и задач выполнения курсовой работы. Библиографический список включает библиографические записи цитируемых, упоминаемых и изученных автором работы документов. Особое внимание уделяется отражению документов последних 3–5 лет, как показателю осведомленности автора, о современном состоянии изучения рассматриваемой им темы. В библиографический список включаются библиографические записи на все документы, независимо от их носителя (печатные материалы: книги, статьи из журналов, сборников, главы из книг; электронные документы, в том числе Интернет-ресурсы; аудиовизуальные, архивные документы).

Документы, включенные в список, представляются в виде библиографических записей, которые следует составлять в соответствии с требованиями государственных стандартов:

1. ГОСТ 7.1-2003 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления».
2. ГОСТ 7.80-2000 «Библиографическая запись. Заголовок. Общие требования и правила составления».
3. ГОСТ 7.82-2001 «Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления».
4. ГОСТ 7.12-93 «Библиографическая запись. Сокращение слов на русском языке. Общие требования и правила».
5. СТБ 7.12-2001 «Библиографическая запись. Сокращение слов и словосочетаний на белорусском языке».
6. ГОСТ 7.11-2004 (ИСО 832:1994) «Библиографическая запись. Сокращение слов и словосочетаний на иностранных европейских языках».

Пример оформления Интернет-ресурса:



1. Национальный правовой Интернет-портал Республики Беларусь [Электронный ресурс]. – Режим доступа: <http://www.pravo.by>. – Дата доступа: 22.11.2021.

**Краткие требования к оформлению работы:**

1. Размер бумаги А4, ориентация бумаги – книжная.
2. Размеры полей:  
левое – 3 см; правое – 1 см; верхнее – 2,5 см; нижнее – 2 см.
3. Рекомендуемый шрифт – **Times New Roman**.  
Размер шрифта - **14 пт**. Для выделения заголовков – полужирный.
4. Установить межстрочный интервал – 18 пт по всей работе.
5. Абзацный отступ в тексте - 1,25 см.
6. Выравнивание основного текста работы – **по ширине**.
7. Выравнивание заголовков, подзаголовков – **по центру**.
8. Каждый новый раздел с новой страницы.
9. Заголовки Содержание и Список использованных источников не нумеровать, записывать симметрично тексту (по центру).
10. В тексте работы обязательны ссылки на литературу или другие источники информации, по образцу: [1].
11. Номера страниц (арабские цифры без точки) располагаются в правом верхнем углу страницы.
12. **Работа сшивается по левому краю.**

## **ВСПОМОГАТЕЛЬНЫЙ РАЗДЕЛ**

### **Выдержки из учебной программы**

Согласно учебному плану для очной формы получения высшего образования на изучение учебной дисциплины отведено всего 200 ч., из них аудиторных - 102 часа.

Распределение аудиторных часов по курсам, семестрам и видам занятий приведено ниже.

Таблица 1.

Очная форма получения высшего образования					
Курс	Семестр	Лекции, ч.	Лабораторные занятия, ч.	Практические занятия, ч.	Форма текущей аттестации
2	3	34	68		курсовая работа, экзамен

## **СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА**

### **Раздел I. БАЗЫ ДАННЫХ**

#### **Тема 1.1 Основные концепции организации данных и реляционная модель данных**

Понятие и классификация СУБД. Иерархические, сетевые, реляционные и объектно-реляционные базы данных. Основные понятия реляционной модели данных, проектирование реляционных СУБД. СУБД с централизованной архитектурой, архитектурой файл-сервер, клиент-сервер и трехуровневой архитектурой (тонкий клиент-сервер приложений -сервер базы данных). Базы данных оперативной обработки транзакций (OLTP) и системы делового анализа (OLAP). Распределенные базы данных. Обзор существующих СУБД.

#### **Тема 1.2 Проектирование реляционных баз данных**

Типовая организация современной СУБД. Ранние подходы к организации СУБД. Общие понятия реляционного подхода к организации баз данных. Базисные средства манипулирования реляционными данными Проектирование реляционных баз данных с использованием нормализации. Внутренняя организация реляционных СУБД. Структуры внешней памяти, методы организации индексов. Управление транзакциями, сериализация транзакций. Журнализация изменений БД. Элементы языка SQL. Функции и основные возможности языка SQL. Выборка данных с использованием предложения SELECT. Вложенные подзапросы.

Манипулирование данными. Архитектура клиент-сервер (InterBase, MySQL, Oracle). Основные особенности архитектуры клиент-сервер. Описание данных на основе SQL. Триггеры и хранимые процедуры. Работа с BLOB и

функции, определенные пользователем. Транзакции. Механизм транзакций. Разработка приложений для работы с БД. Описание интерфейса среды (Delphi, C++ Builder, FoxPro) и ее компонентов для работы с клиент-серверной БД. Компоненты доступа к данным и визуальные компоненты. Технология InterBase Express (IBX). Технология dbExpress. Технология доступа к данным ADO.

### **Тема 1.3 Разработка приложений для работы с базой данных**

Система безопасности СУБД: установка пароля на базу данных, защита структуры базы данных в MDE-файлах, шифрование и дешифрование базы данных.

Защита на уровне пользователей: модель разграничения доступа, файл рабочей группы.

Макросы, использование программ VBA в модулях и формах MS Access. Объектные модели DAO и ADO.

### **Тема 1.4. Введение в язык SQL**

Язык SQL - назначение, особенности, преимущества, существующие стандарты.

Запросы в SQL: простые и многотабличные запросы, внутренние и внешние объединения, итоговые запросы, запросы с группировкой, подчиненные запросы. Представления в SQL. Псевдонимы и индексы таблиц.

Целостность данных: правила каскадного удаления и обновления данных, ограничения на значение столбцов, домены и утверждения в SQL.

Изменение данных при помощи SQL. Понятие транзакции, обработка транзакций. Работа в многопользовательском режиме и блокировка.

## **Раздел II. СЕТЕВЫЕ ТЕХНОЛОГИИ**

### **Тема 2.1. Введение в компьютерные сети. Аппаратное и программное обеспечение компьютерных сетей. Протоколы компьютерных сетей**

Понятие и состав компьютерной сети. Признаки классификации компьютерных сетей. Локальные и глобальные компьютерные сети, понятие "интранет". Одноранговые сети и сети с выделенным сервером, технология клиент-сервер.

Основные типы кабельных систем (коаксиальный кабель, витая пара, оптическое волокно), топология компьютерной сети. Основные типы сетевых адаптеров (сетевых карт). Сетевое оборудование: концентраторы, мосты, коммутаторы, маршрутизаторы, понятие шлюза, коммутаторы третьего уровня. Ограничения, существующие при построении локальной сети.

Понятие протокола, модель OSI ISO, как решение проблемы совместимости компьютерных сетей. Основные протоколы канального уровня,

основные протоколы сетевого и транспортного уровня, стек протоколов TCP/IP. Понятие и структура IP-адреса, классы IP-адресов, маска подсети, бесклассовая междоменная маршрутизация CIDR, широковещательные адреса, зарезервированные IP-адреса. Назначение протокола ICMP. Назначение протокола ARP. Протоколы транспортного уровня TCP и UDP, понятие "порт" и "сокет". Протоколы прикладного уровня. Документы RFC.

Обзор существующих сетевых операционных систем, основы сетевого администрирования в выбранной для изучения ОС.

Развертывание и особенности корпоративных сетей. Преимущества и проблемы компьютерных сетей.

## **Тема 2.2. Глобальные компьютерные сети**

Глобальные сети X.25, FrameRelay, ATM и Internet. История возникновения сети Internet и перспективы ее развития. Структура сети Internet, особенности адресации в Internet, понятие автономной системы, служба whois, служба DNS.

Подключение к Internet: выделенные и коммутируемые, цифровые и аналоговые линии, модемы, линии PDH (E1-E4) и SDH, сети ISDN, технология ADSL. Выбор провайдера (ISP).

Ресурсы Internet: основные сервисы Internet, концепция URL. Основные источники информации и их размещение в Internet. Классификация поисковых систем, стратегия поиска. Навигация в сети WWW.

Большие данные (Big Data): современные подходы к обработке и хранению. Извлечение и визуализация данных. Процесс построения моделей. Формы представления данных, типы и виды данных. Представления наборов данных. Технологии KDD и Data Mining. Методика извлечения знаний. Data Mining. Причины распространения KDD и Data Mining. Актуальность технологий Data Mining как средств обработки больших объемов информации. Программное обеспечение в области анализа данных. Аналитические платформы: классификация и особенности применения.

## **Тема 2.3. Основы бизнеса в Internet**

Понятие электронного бизнеса и его основные направления. Исследование аудитории Internet, организация обратной связи. Web-представительство - основа бизнеса в Internet.

Электронная коммерция: понятие и особенности электронной коммерции, категории электронной коммерции, проблемы и прогнозы развития электронной коммерции.

Торговые системы Internet: понятие и классификация торговых систем Internet, Internet-магазины (понятие и функции Internet-магазина, способы

оплаты товаров, способы доставки товаров, требования к организации Internet-магазина).

Internet - аукционы: понятие и функции Internet-аукциона, схемы торгов на Internet-аукционе.

Торговые Internet-площадки. Риски и выгоды участников.

Финансовые системы Internet: Internet-банкинг, Internet-трейдинг, Internet-страхование.

Платежные системы Internet: понятие и классификация платежных систем, обзор кредитных систем, обзор дебетовых систем, российские платежные системы.

#### **Тема 2.4. Вопросы безопасности информации в Internet**

Понятие безопасности информации. Законодательная база в области информационной безопасности (международное и белорусское законодательство).

Основные виды компьютерных атак (атаки типа "отказ в обслуживании" DOS, атаки на протоколы, атаки на операционные системы и сервисы, атаки с использованием уязвимостей ПО конечного пользователя, атаки с использованием "социальной инженерии", понятие "сканирование портов") и инструменты применяемые хакерами.

Методы обеспечения безопасности в сети Internet: межсетевые экраны (firewalls), трансляция сетевых адресов (NAT), прокси-серверы, системы обнаружения атак (IDS), системы анализа защищенности сети, криптографическая защита информации (симметричные и ассиметричные алгоритмы шифрования), стеганография, электронная цифровая подпись, понятие электронного сертификата, виртуальные частные сети (VPN).

#### **Тема 2.5. Общие сведения о проектировании web-узлов. Язык HTML**

Этапы разработки web-узла. Разработка Web-документов средствами программ пакета MS Office.

Язык HTML: назначение, история создания, структура HTML-документа, основные тэги, гиперссылки, закладки, тэги оформления текста, графика, создание таблиц, списки, создание и использование форм, фреймы, разделенные рисунки (карты-изображения), стили CSS в DHTML.

Разработка Web-документов с помощью сред визуального проектирования.

#### **Тема 2.6 Использование программ, выполняющихся в контексте Web-браузера, при разработке Web-документов. Динамическая генерация содержимого Web-документа**

Язык JavaScript: назначение, существующие версии языка, включение программ в Web-страницу, основные конструкции языка, использование

объектной модели браузера в JavaScript, создание сценариев с учетом типа Web-браузера.

Язык Java: независимость от платформы (виртуальная Java машина), апплеты Java, встраивание апплетов в Web-страницу, ограничения возможностей апплетов, система безопасности Java.

Объекты ActiveX: назначение, методы включения в Web-страницу, возможности, проблемы с безопасностью. Редактор Macromedia Flash: обзор возможностей по созданию анимации и интерактивному взаимодействию с пользователем (язык ActionScript).

CGI-программирование, директивы включения на стороне сервера (SSI), активные серверные сценарии (ASP-файлы), технология PHP.

**УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ**  
**очная форма получения высшего образования**

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1.	Базы данных							
1.1	Основные концепции организации данных и реляционная модель данных	2			4			
1.2	Проектирование реляционных баз данных	6			20			опрос
1.3	Разработка приложений для работы с базой данных	2			6			
1.4	Введение в язык SQL	2			4			опрос
2.	Сетевые технологии							
2.1	Введение в компьютерные сети. Аппаратное и программное обеспечение компьютерных сетей. Протоколы компьютерных сетей	4			4			
2.2	Глобальные компьютерные сети	4			6			опрос
2.3	Основы бизнеса в Internet	6			4			
2.4	Вопросы безопасности информации в Internet	2			4			опрос
2.5	Общие сведения о проектировании web-узлов. Язык HTML	4			10			опрос
2.6	Использование программ, выполняющихся в контексте Web-браузера, при разработке Web-документов. Динамическая генерация содержимого Web-документа	2			6			опрос, КР
	Итого за семестр	34			68			экзамен
	Всего аудиторных часов				102			

### *Рекомендуемая литература*

1. Автоматизация проектирования БД. Разработка базы данных: практикум / Е.Г. Гриневич, И.Г. Орешко, Ю.Н. Силкович. - Минск: Печатный Дом "Вишневка", 2018. - 143 с.
2. Базы данных: [в 2 кн.] / В. П. Агальцов. - 2-е изд., переработанное. - Москва: Форум, Инфра-М, 2017.
3. Базы данных / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. - 3-е изд., переработанное и дополненное. - Москва: Юрайт, 2018. - 418, [2] с.
4. Компьютерные информационные технологии / Министерство образования Республики Беларусь, Учреждение образования "Полоцкий государственный университет". Ч. 3, кн. 2: Технологии баз данных и знаний: в 2 кн. / С. Е. Рясова. — 2017. — 140 с.
5. Компьютерные сети / Л. П. Ганчарик. - Минск: Академия управления при Президенте Республики Беларусь, 2015. - 119 с.
6. Компьютерные сети. Принципы, технологии, протоколы / В. Олифер, Н. Олифер. - 4-е изд. - Санкт-Петербург [и др.]: Питер, Питер Пресс, 2015. - 943 с.
7. Компьютерные сети / Э. Таненбаум, Д. Уэзеролл. - 5-е изд. - Санкт-Петербург [и др.]: Питер, Питер Пресс, 2017. - 955 с.
8. Коннолли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика: пер. с англ. / Т. Коннолли, К. Бегг. - М.: Вильямс, 2017. - 1439 с.
9. Лукин, В.Н. Введение в проектирование баз данных / В.Н. Лукин. - М.: Вузовская книга, 2015. - 144 с.
10. Основы информационных технологий: курс лекций / С.Н. Батан, Л.В. Батан, О. В. Малашук. - Могилев: МГУ, 2016. - 118, [1] с.
11. Сети и системы передачи данных "CCNA Routing&Switching" / Министерство связи и информатизации Республики Беларусь, Учреждение образования "Белорусская государственная академия связи". Ч. 1: Основы сетевых технологий. — 2019. — 58 с.
12. Стружкин, Н.П. Базы данных: проектирование. Практикум / Н.П. Стружкин, В.В. Годин. — Москва: Издательство Юрайт, 2019. — 291 с.
13. Технологии создания компьютерных баз данных: пособие / Н.Ф. Богданова. - Минск: ИВЦ Минфина, 2019. - 88 с.