



Министерство образования
Республики Беларусь

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра «Организация упаковочного производства»

Е.К. Костюкевич
И.В. Остапенко

ИНФОРМАЦИОННЫЕ И СЕТЕВЫЕ ТЕХНОЛОГИИ В УПАКОВОЧНОМ ПРОИЗВОДСТВЕ

Методическое пособие



Минск
БНТУ
2011

Министерство образования Республики Беларусь
БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

Кафедра «Организация упаковочного производства»

Е.К. Костюкевич
И.В. Остапенко

ИНФОРМАЦИОННЫЕ И СЕТЕВЫЕ ТЕХНОЛОГИИ
В УПАКОВОЧНОМ ПРОИЗВОДСТВЕ

Методическое пособие
для студентов специальности
1-36 20 02 «Упаковочное производство»

Минск
БНТУ
2011

УДК 621.798:004(075.8)

ББК 30.61я7

К 72

Рецензенты:

Н.И. Березовский, В.Ф. Одиночко

Костюкевич, Е.К.

К 72 Информационные и сетевые технологии в упаковочном производстве: методическое пособие для студентов специальности 1-36 20 02 «Упаковочное производство» / Е.К. Костюкевич, И.В. Остапенко. – Минск: БНТУ, 2011. – 76 с.

ISBN 978-985-525-402-8.

В данном пособии рассмотрены современные стандарты сетевых технологий, различные информационные сервисы Internet, служба WWW, технологии поиска информации, электронная почта. Раскрыты основы языка гипертекстовой разметки HTML. Изложены вопросы создания документов, предназначенных для использования на Web-страницах. Представлены технологии создания различных Web-документов с помощью языка гипертекстовой разметки HTML, а также интерактивных технологий Dynamic HTML и CSS. Изложены различные технологии верстки Web-сайтов: фреймовая, табличная, контейнерная.

УДК 621.798:004(075.8)

ББК 30.61я7

ISBN 978-985-525-402-8

© Костюкевич Е.К.,
Остапенко И.В., 2011
© БНТУ, 2011

ВВЕДЕНИЕ

Настоящее методическое пособие предназначено для изучения и практического освоения информационных и сетевых технологий, используемых в упаковочном производстве.

Информационные технологии, использующие возможности Internet, проникают в большинство сфер человеческой деятельности. Эффективный поиск профессиональной информации, разработка корпоративных сайтов, предложение рекламной продукции посредством Internet и другие услуги прочно входят в повседневную деловую жизнь, обеспечивают неограниченные возможности рекламы продукции и услуг, решение задач дистанционного образования. Умение использовать эти возможности в области упаковочного производства обеспечивают её успешность. В настоящее время веб-пространство стало такой же платформой для создания приложений, как и персональный компьютер. Поэтому овладение технологиями создания веб-приложений является обязательным элементом информационной подготовки современного специалиста.

Язык HTML, или универсальный язык разметки гипертекста, используется для создания самых разных интерактивных документов с гиперссылками и элементами мультимедиа — Web-страниц, интерфейсов, презентаций, электронных книг и учебных пособий. Файлы с HTML-кодом — это обычные текстовые файлы, доступные для чтения, как программе, так и человеку. Благодаря этому HTML-страницы можно редактировать и просматривать на любом компьютере и в любой операционной системе.

Для создания HTML-страниц можно пользоваться любым текстовым редактором, но существуют и специализированные программы. Эти HTML-редакторы делятся на два типа: визуальные и не визуальные. Визуальные HTML-редакторы обладают интуитивно понятным интерфейсом и не требуют много времени на освоение, но генерируют не достаточно оптимальный HTML-код, который потом необходимо редактировать. К редакторам этого типа относится, например Dreamweaver.

Не визуальные редакторы требуют знания HTML, но лишены недостатков визуальных редакторов. Кроме того, это, как правило, небольшие, компактные программы, бесплатные или условно-бесплатные.

HTML не является языком программирования. Поэтому для того чтобы добавить интерактивности HTML-странице необходимо использовать средства, расширяющие возможности HTML: каскадными таблицами стилей CSS и языком программирования JavaScript.

Лабораторная работа № 1

Браузер MS Internet Explorer

Цель работы: изучить пользовательский интерфейс и параметры настройки браузера.

Назначение основных кнопок панели инструментов

- **Назад** (Back). Возвращает к предыдущему просмотренному ранее документу.
- **Вперед** (Forward). Переносит к следующему документу из списка.
- **Остановить** (Stop). Остановить текущую загрузку документа.
- **Обновить** (Refresh). Производит перезагрузку текущего документа.
- **Домой** (Home). Осуществляет переход к документу, установленному в качестве начальной (домашней) страницы.
- **Поиск** (Search). Поиск документов на поисковых серверах WWW.
- **Избранное** (Favorites). Список ссылок на серверы, составленные пользователем.
- **Журнал**. Показывает все работы пользователя за определенный период времени.

Задание

1. Запустите программу *MS Internet Explorer*.
2. Произведите следующие параметры настройки браузера:
 - подключение панели инструментов и строку состояния;
 - включение отображения подписей на панели инструментов.

3. Настройте папку **Журнал** для хранения ссылок на посещение страницы в течение 12 дней.

Рекомендации:

- 1) в меню **Сервис** выберите команду **Свойства обозревателя...;**
 - 2) на вкладке **Общие** в разделе **Журнал** в поле **Сколько дней хранить** ссылки введите нужную информацию.
4. Включите загрузку мультимедийного содержимого.

Рекомендации:

- 1) в меню **Сервис** выберите команду **Свойства обозревателя...;**
 - 2) на вкладке **Дополнительно** в блоке **Мультимедиа** установите флажки **Воспроизводить анимацию, Воспроизводить видео, Воспроизводить звуки, Отображать рисунки.**
5. Проверьте настройку шрифта кириллица.

Рекомендации:

- 1) в меню **Сервис** выберите команду **Свойства обозревателя...;**
 - 2) на вкладке **Общие** щелкните по кнопке **Шрифты...;**
 - 3) в поле **Набор** знаков выберите кириллица.
6. Установите в качестве домашней (стартовой) страницы www.bntu.by.

Рекомендации:

- 1) откройте Web-страницу www.bntu.by;
 - 2) в меню **Сервис** выберите команду **Свойства обозревателя...;**
 - 3) на вкладке **Общие** в разделе **Домашняя страница** в поле **Адрес** введите нужный адрес.
7. Откройте домашнюю страницу (кнопка **Домой**).
 8. Откройте страницу одного из указанных **Web-сайтов**.
www.bntu.by
www.open.by
www.rambler.ru
www.yandex.ru
 9. Остановите процесс загрузки.
 10. Обновите (загрузите заново) текущую станицу.
 11. Перейдите по любой гипертекстовой ссылке.

12. Вернитесь к предыдущему документу.
13. Откройте документ по гипертекстовой ссылке в новом окне.

Рекомендации:

- 1) щелкните правой кнопкой мыши по любой гиперссылке;
- 2) выберите команду **Открыть в новом окне**.
14. Установите в качестве домашней (стартовой) страницы **пустую страницу**.
15. Отключите загрузку мультимедийного содержимого.
16. На сайте Белорусского национального технического университета с помощью гипертекстовых ссылок просмотрите информацию о факультетах этого вуза.
17. На портале *www.pravo.by* с помощью поля Поиск в **каталоге ресурсов** по запросу **Организация упаковочного производства** найти сайт кафедры.
18. Сохраните адрес текущей страницы в виде закладки в папке **Закладки**.

Рекомендации:

- 1) в меню **Избранное** выберите команду **Добавить в избранное...**;
- 2) в окне **Добавить в избранное...** щелкнуть по кнопке **Создать папку...**;
- 3) ввести имя создаваемой папки.
19. Найдите на текущей Web-странице информацию о деятельности и научной работе кафедры.
20. Используя буфер обмена перенесите информацию о деятельности кафедры в текстовый редактор MS Word.
21. Сохраните информацию в виде файла с именем **Деятельность кафедры**.
22. Сохраните текущую Web-страницу целиком в виде файла **Кафедра** в формате HTML в папке *Page*.
23. Просмотрите сохраненную Web-страницу.
24. Просмотрите список адресов посещенных страниц (кнопка Журнал).
25. Перейдите к любой Web-странице, посещенной сегодня.
26. Создайте для текущей странице ярлык на рабочем столе.

27. Закройте список адресов посещенных страниц (кнопка Журнал).
28. Удалите папку с закладками **Закладки**.
29. Удалите созданный для Web-страницы ярлык на рабочем столе.

Лабораторная работа № 2

Поиск информации в Internet

Цель работы: изучить технологию поиска и сохранения профессиональной информации в Internet.

Задание

1. Найдите ссылки на сайты ведущих вузов РБ в каталоге поисковой системе *www.all.by*.

Рекомендации:

- 1) на начальной странице поисковой системы в списке категории сайтов перейдите к ссылке **Образование**;
 - 2) выберите нужную ссылку.
2. Создайте закладку на открытую страницу в личной папке **Закладки** под именем **Вузы**.
 3. Создайте параллельно три сеанса работы с браузером. Откройте в разных окнах поисковые системы:
www.open.by
www.nigma.ru
www.yandex.ru
 4. Найдите в разных поисковых системах Web-страницы, содержащие слова **упаковочные материалы** в любой части документа.
 5. Сравните результаты поиска по количеству и качеству найденных ссылок.
 6. В окне поисковой системы *www.open.by* перейдите на первую ссылку.
 7. Сохраните информацию, содержащуюся на странице, в файле текстового редактора MS Word с именем **Специальность**.

8. В окне поисковой системы *www.yandex.ru* перейдите первую ссылку.
9. Сохраните текущую Web-страницу в своей рабочей папке.
10. Просмотрите сохраненную Web-страницу.
11. Составьте запросы в *www.nigma.ru* найдите документы, ссылающиеся на сервер Московского государственного университета *www.msu.ru*.

Рекомендации:

- 1) на начальной странице поисковой системы в поле ввода запроса введите *link=www.msu.ru* (ключевое слово *link* задает поиск документов, в которых есть ссылки на определенный сервер);
- 2) проведите поиск по составленному запросу.
12. С помощью поисковой системы *www.rambler.ru*, найдите Web-страницы, где встречаются слова **Упаковка** (и) или **Тара**.

Рекомендации:

- 1) на начальной странице поисковой системы в поле ввода запроса введите **Упаковка OR Тара** (оператор *OR* задает поиск документов, содержащих любое из нескольких ключевых слов);
- 2) проведите поиск по составленному запросу.
13. Произведите поиск фотографий объектов поиска, при просмотре найденных ссылок на Web-страницы.
14. Сохраните одну из найденных фотографий, в виде графического файла *Foto* в вашей рабочей папке.
15. Найдите Web-страницы со словами **Функции упаковки** в заголовке, в точной форме, с датой после 1 января 2008 года с использованием режима расширенного (детального) поиска в поисковой системе *www.yandex.ru*.

Рекомендации:

- 1) на начальной странице поисковой системы перейдите по ссылке **Расширенный поиск**;
- 2) введите в соответствующие поля на странице необходимую информацию;
- 3) щелкните по кнопке **Найти**.
16. Сохраните найденные ссылки в виде Web-страницы в вашей рабочей папке с именем *Profession*.

17. Сохраните подборку информации по вышеназванной тематике в виде файла с расширением .doc и именем Спецификация папке *Profession*.
18. Составить запросы в соответствии с вариантом:
 - 1 Утилизация упаковки
 - 2 Упаковочные материалы
 - 3 Упаковка и реклама
 - 4 Дизайн упаковки
 - 5 Рециклинг упаковки
 - 6 Упаковка и этикетка
 - 7 Биоразлагающаяся упаковка
 - 8 Функции упаковки
 - 9 Свойства упаковочных материалов
0. 1 Тара и упаковка
1. 1 Многослойная упаковка
1. 1 Конструирование упаковки
2. 1 Профессия упаковщик
3. 1 Полиграфия и упаковка
4. 1 Упаковочное производство в РБ
5. 1 Переработка упаковки
- 6.

19. Сохраните подборку информации по вышеназванной тематике в виде файла с вашей фамилией в папке *Profession*.
20. Удалите папку с закладками *Закладки*.

Лабораторная работа № 3

Разработка простой WEB-страницы на языке HTML с использованием форматирования шрифтов и текста

Цель работы: разработать WEB-страницу, используя различные элементы форматирования содержимого и текста .

HTML – это язык форматирования текста. HTML-документ – это просто текстовый файл с расширением *.html.

Фрагмент текста, заключенный между символами < и >, называется тегом (от английского слова TAG, в переводе – метка).

Тег – это символьная конструкция из < (открывающей) и >(закрывающей) угловых скобок, между которыми находится конкретный символ или строка символов, предписывающие браузеру отображение последующего содержания документа в соответствии с их назначением.

Язык HTML использует различные теги, вводимые в текстовые документы, которые указывают, каким образом информация должна считываться WEB-браузером. Большинство тегов HTML парные, открывающий и закрывающий. Они охватывают помечаемый ими текст. Закрывающий тег пары всегда начинается с прямой косой черты. Теги этого текста определяют, каким образом текстовая информация и графика должны быть представлены на экране. Кроме тегов <HTML>, <HEAD>, <TITLE>и <BODY>, которые в соответствии со стандартами HTML и WWW должны присутствовать в каждой WEB-странице, при вводе текста HTML-файла для разбиения его на смысловые группы и стилизового оформления текста используются несколько тегов разметки.

Теги, задающие структуру документа

<!DOCTYPE >	Декларирование типа документа	Используется для указания, с каким стандартом HTML
-------------------	--------------------------------------	--

<HTML>	Тип структуры - HTML	совместим документ начало структуры HTML
<HEAD>	Начало описания документа	Раздел описания документа может включать метки <TITLE>, <META>, <BASE> и <LINK>
<TITLE> </TITLE>	Имя документа	То, что будет считаться заголовком (названием) документа
<META	Мета-информация	Служит для указания: а) технической информации о документе б) информации о содержании документа
HTTP-EQUIV="имя" CONTENT="значение"	Информация для HTTP-сервера: присвоить какое-либо значение какому-либо заголовку	Для использования кириллицы на HTML-страницах укажите в заголовке: <META HTTP-EQUIV="Content-Type" CONTENT="txt/html; charset=Windows-1251">
NAME="имя" CONTENT="значение">	Задание мета-переменной и присвоение ей значения	Задание мета-информации о документе. Поисковые службы судят по ней о содержании документа.

<LINK >	Указания о гиперсвязях данного документа	Указываемые атрибуты - такие же, как у метки <A> (якорь, anchor). Служит для указания не связей, а отношений между документами.
</HEAD>	Конец описания документа	
<BODY	Начало документа	указываются установки для показа документа
BACKGROUND=" URL"	Фоновая картинка	в кавычках указывается URL картинки (.gif или .jpg)
BGCOLOR="#\$\$\$\$\$\$"	Цвет фона	в кавычках указывается номер цвета или его название
TEXT="#\$\$\$\$\$\$"	Цвет текста	
LINK="#\$\$\$\$\$\$"	Цвет ссылки	
VLINK="#\$\$\$\$\$\$"	Цвет уже просмотренной ссылки	
ALINK="#\$\$\$\$\$\$" >	Цвет активной ссылки	
	Содержимое документа	
</BODY>	Конец документа	
</HTML>	Конец структуры HTML	

Задание

Кроме использования тегов <HTML>, <HEAD>, <TITLE> и <BODY>, которые в соответствии со стандартами HTML и WWW должны присутствовать в каждой WEB-странице, при организации текста внутри WEB-документа с помощью списков, а также использовании предварительного форматирования и графики применяются соответствующие элементы языка HTML с определенным набором тегов и требуемых атрибутов.

1. Используя редактор текста Блокнот (Notepad) разработайте страницу, используя все приведенные в таблице теги языка HTML. Сохраните страницу в рабочем каталоге. Отформатируйте текст Вашего варианта задания, согласно указаниям в скобках:

2. Посмотрите на свою страницу с помощью установленного на компьютере браузера и добейтесь максимального подобия в представлении страницы браузером.

Для выполнения данной работы вам необходимо использовать следующие теги элементов языка HTML:

Пример элемента	Пояснение
<HTML>...</HTML>; <HEAD>...</HEAD>; <BODY>...</BODY>	Обязательные элементы для любой страницы HTML
<TITLE>...</TITLE>	Название страницы
<H1>...</H1>; <H2>...</H2>; <H3>...</H3>; <H4>...</H4>; <H5>...</H5>; <H6>...</H6> атрибут align="left right center"	Стили заголовков внутри страницы
<P>...</P> атрибут align="left right center justify"	Элемент нового абзаца.
 	Тег перевода строки. Не разделяет два фрагмента пустой строкой.
<HR> атрибуты: size, noshade, width, align, col-	Тег горизонтальной линии.

or	
...; <I>...</I>; <TT>...</TT>; <S>...</S>; <U>...</U>; <BIG>...</BIG> ;<SMALL>...</SMALL>; _{...}; ^{...}	Оформление текста
... атрибуты: size, face, color	Задание размера, типа шрифта и цвета текста.
<BODY text="Цвет">...</BODY>	Способ управления цветом текста.
&-последовательности	Отображение специальных символов.
...; ...; <SAMP>...</SAMP>; <VAR>...</VAR>; <CODE>...</CODE>	Логические стили модификации текста.

Дизайн упаковки

(заголовок H1, выравнивание по центру)

Современные технологии дизайна упаковки

(заголовок H2, выравнивание по правому краю)

Элемент абзаца один из самых полезных. Он допускает использование только открывающего тега, так как открывающий тег следующего абзаца <P> обозначает не только начало следующего абзаца, но и конец предыдущего.

(выравнивание абзаца по левому краю, полужирный шрифт)

*Текст этого абзаца выравнивается по центру
с помощью атрибута align*

(переход на новую строку с помощью элемента
 , выравнивание абзаца по центру, курсив)



(горизонтальная линия с помощью тега <HR> , без тени, толщина 10, цвет красный)

Голограмма на саморазрушающей основе состоит из прозрачного слоя (пленка), 1^{-го} слоя клея, алюминиевого слоя, 2^{-го} слоя клея.

(использование элемента SUP для верхних индексов, выравнивание абзаца по центру)

HTML - это язык разметки гипертекста

(использование элемента FONT для задания типа, размера и цвета шрифта, тип шрифта "Arial", размер - 5, цвет - зеленый, выравнивание абзаца по левому краю)

Программисты представляют свои исходные коды моноширинным текстом. Для них создан элемент логической разметки <CODE>...</CODE>

(использование элемента логической разметки CODE для форматирования текста и &-последовательностей - для угловых скобок, выравнивание абзаца по центру)

Упаковочные материалы

(Заголовок H1, выравнивание по левому краю)

Барьерные полимеры

(Заголовок H2, выравнивание по центру)

Элемент абзаца один из самых полезных. Он допускает использование только открывающего тега, так как открывающий тег следующего абзаца `<P>` обозначает не только начало следующего абзаца, но и конец предыдущего.

(выравнивание абзаца по правому краю, полужирный шрифт)

Текст этого абзаца выравнивается по центру с помощью атрибута `align`

(переход на новую строку с помощью элемента `
`, выравнивание абзаца по центру, курсив)



(горизонтальная линия с помощью тега `<HR>`, толщина 5, длина - 50% от ширины экрана, выравнивание по правому краю)

Активные барьеры хуже удерживают CO₂, но обладают способностью связывать O₂

(использование элемента SUB для нижних индексов, выравнивание абзаца по центру)

Понятие HTML включает в себя различные способы оформления гипертекстовых документов

(использование элемента FONT для задания типа, размера и цвета шрифта, тип шрифта "Verdana", размер - 6, цвет - красный, выравнивание абзаца по левому краю)

Элемент логической разметки `<VAR>...</VAR>` обозначает или имя переменной, или данные, вводимые пользователем.

Фрагмент текста внутри тегов этого элемента обычно отображается курсивом.

(использование элемента логической разметки VAR для форматирования текста и &-последовательностей - для угловых скобок, выравнивание абзаца по центру)

Упаковочные материалы

(Заголовок H1, выравнивание по правому краю)

Характеристики упаковочных материалов

(Заголовок H2, выравнивание по центру)

Элемент абзаца один из самых полезных. Он допускает использование только открывающего тега, так как открывающий тег следующего абзаца <P> обозначает не только начало следующего абзаца, но и конец предыдущего.

(выравнивание абзаца по центру, полужирный шрифт)

Текст этого абзаца выравнивается по левому краю с помощью атрибута align

(переход на новую строку с помощью элемента
 , выравнивание абзаца по левому краю, курсив)



(горизонтальная линия с помощью тега <HR>, толщина 15, длина - 300 пикселей, выравнивание по правому краю)

Плотность полистирола (ρ_{ps}) – 1,05 г/см³

(использование элемента SUB для нижних индексов, SUP для верхних индексов, выравнивание абзаца по центру)

HTML имеет прямое отношение к такому направлению изобразительного искусства, как Web-дизайн

(использование элемента FONT для задания типа, размера и цвета шрифта, тип шрифта "Comic Sans Ms", размер - 16, цвет - синий, выравнивание абзаца по левому краю)

Элемент логической разметки ... - используется для выделения очень важных фрагментов текста. Фрагмент текста внутри тегов этого элемента обычно отображается полужирным шрифтом.

(использование элемента логической разметки STRONG для форматирования текста и &-последовательностей - для угловых скобок, выравнивание абзаца по центру)

Классификация бумаги и картона

(Заголовок H1, выравнивание по центру)

Влагопрочная бумага

(Заголовок H2, выравнивание по левому краю)

Элемент абзаца один из самых полезных. Он допускает использование только открывающего тега, так как открывающий тег следующего абзаца <P> обозначает не только начало следующего абзаца, но и конец предыдущего.

(выравнивание абзаца по левому краю, полужирный шрифт)

Текст этого абзаца выравнивается по левому краю - с помощью атрибута align

(переход на новую строку с помощью элемента
, выравнивание абзаца по левому краю, курсив)



(горизонтальная линия с помощью тега <HR>, без тени, толщина 10, длина - 100%, цвет - Orange)

Химическая формула воды - H₂O

(использование элемента SUB для нижних индексов, выравнивание абзаца по центру)

Человек, изучивший html обретает возможность делать сложные вещи простыми способами и, главное, быстро.

(использование элемента FONT для задания типа, размера и цвета шрифта, тип шрифта "Impact", размер - 4, цвет - Indigo, выравнивание абзаца по левому краю)

Элемент логической разметки <SAMP>...
</SAMP> - используется для выделения фрагментов текста моноширинным шрифтом, в том числе для демонстрации сообщений, выводимых программами.

(использование элемента логической разметки SAMP для форматирования текста и &-последовательностей - для угловых скобок, выравнивание абзаца по центру)

Классификация полимерных материалов

(Заголовок H2, выравнивание по правому краю)

Полипропилен

(Заголовок H3, выравнивание по левому краю)

Элемент абзаца один из самых полезных. Он допускает использование только открывающего тега, так как открывающий тег следующего абзаца <P> обозначает не только начало следующего абзаца, но и конец предыдущего.

(выравнивание абзаца по центру, полужирный шрифт)

Текст этого абзаца выравнивается по правому краю с помощью атрибута align

(переход на новую строку с помощью элемента
 , выравнивание абзаца по правому краю, полужирный курсив)



(горизонтальная линия с помощью тега <HR>, без тени, толщина 30, длина - 30, цвет - Navy)

Плотность полистирола – 1,05 г/см³

(использование элемента SUP для верхних индексов, выравнивание абзаца по центру)

***Для создания Web-страницы
требуется текстовый редактор
Блокнот.***

(использование элемента FONT для задания типа, размера и цвета шрифта, тип шрифта Monotype Corsiva размер - 10, цвет - Maroon, выравнивание абзаца по правому краю)

Элемент логической разметки <DFN>...</DFN> - используется для выделения какого-либо термина при первом употреблении в тексте документа. Отображается курсивом.

(использование элемента логической разметки DFN для форматирования текста и &-последовательностей - для угловых скобок, выравнивание абзаца по центру)

Лабораторная работа № 4

Разработка WEB-страницы на языке HTML с использованием списков

Цель работы: разработать WEB-страницу с использованием вложенных списков разных типов.

Существует несколько форматов списков, позволяющих выделять фрагменты информации в тексте. Пунктами списка могут быть как элементы уровня текста, так и элементы уровня блока, за исключением заголовков (H1-H6) и элементов ADDRESS. Пунктами списка могут быть также другие списки.

UL (нумерованные списки)

Для создания нумерованного списка используется элемент UL. Для этого элемента обязательны открывающий и закрывающий теги. Для маркировки заголовка или названия списка применяется элемент языка LH (от английского List Heading). Для маркировки отдельного элемента списка применяется элемент языка LI (от английского List Item). Тип маркировки (метки) элемента списка определяется атрибутом type. Допустимые значения атрибута и, соответственно, символы маркировки диск, круг, квадрат: `<UL type = "disc|circle|square">` или `<LI type = "disc|circle|square">`

OL (нумерованные списки)

Для создания нумерованного списка используется элемент OL. Тип нумерации элемента списка определяется атрибутом type. `<OL type = "1|A|a||i">` или `<LI type = "1|A|a||i">`

Атрибут start позволяет начать нумерацию списка не с единицы. Например: `<OL start=4>`

Значение атрибута value элемента LI позволяет изменить номер данного элемента списка. При этом изменяется и нумерация всех последующих элементов списка. Например: `<LI value=5>` Атрибут value может заменять атрибут start, если включен в первый элемент списка.

DL (списки определений)

В списке определений для каждого пункта предоставляется не одна, а две строки. Для написания каждого пункта списка применяются два элемента: DT – определяемый термин (Definition Term)

и DD – описание определения (Definition Data). Известное использование списка определений – словарь.

Вложенные списки

Каждый пункт списка может представлять собой самостоятельный список. Вложение списков в списки создает несколько уровней информации. Можно вкладывать друг в друга разные типы списков. Каждый внутренний список должен иметь открывающий и закрывающий теги UL или OL.

Задание

Используя редактор текста Блокнот (Notepad) разработайте страницу, содержащую:

1. Ненумерованные списки.
2. Нумерованные списки.
3. Списки определений.
4. Используя теги предварительного форматирования сделайте такое же расположение заголовков и ячеек, как в образце.
5. Сделайте фоном вашей страницы подходящее графическое изображение из каталога изображений, используя его в качестве значения атрибута background в открывающем теге элемента тела вашего документа Body.

Посмотрите на свою страницу с помощью установленного на компьютере браузера и добейтесь максимального подобия в представлении страницы.

Для выполнения данной работы вам необходимо использовать следующие теги элементов языка HTML:

Пример тега	Пояснение
<code><HTML> </HTML>; <HEAD> <TITLE> </TITLE> </HEAD>; <BODY> </BODY></code>	Обязательные теги для любой страницы HTML.
<code><LH> </LH>... атрибут type=disc circle square</code>	Оформление ненумерованных списков.
<code><LH> </LH>... атрибут type=1 A a i атрибут start =</code>	Оформление нумерованных списков.

<DL><LH> </LH> <DT><DD>...<DT><DD></DL> атрибут compact	Оформление списков определенных.
<PRE> </PRE>	Предварительное форматирование. Для сохранения расположения текста, заданного при наборе, этот текст помещают между открывающим и закрывающим тегами PRE.
	Подстановка изображения.
атрибуты: bgcolor, background открывающего тега элемента <BODY> </ BODY>	Управление цветом фона страницы и задание в качестве фона страницы графического изображения.

Вариант 1

1. Ненумерованный список

Основные функции упаковки для мясной продукции:

- сохранение первоначального качества мясных продуктов;
- участие в формировании бренда;
- представление информации о продукте и производителе;
- защита торговой марки от подделок;
- привлекательная и удобная для покупателя форма

(использование элемента UL для ненумерованных списков, маркер элемента списка – окружность, разметка заголовка списка полужирным шрифтом с помощью элементов физического форматирования)

2. Нумерованный список

Материалы, используемые для упаковывания мясных продуктов:

- B. Целлофан.
- C. Полистирол.
- D. Полиолефиновые пленки.
- E. Многослойные материалы.
- F. Комбинированные материалы.

(использование элемента OL для нумерованных списков, способ нумерации элементов списка – заглавные латинские буквы, начиная с B, разметка заголовка списка полужирным шрифтом с помощью элементов физического форматирования)

3. Список определений

Аббревиатуры для обозначения наиболее распространенных полимеров, используемых в упаковочном производстве:

PETE или PET

Полиэтилентерафталат – ПЭТФ (ПЭТ)

HDPE

Полиэтилен низкого давления, высокой плотности ПЭНД

LDPE

Полиэтилен высокого давления, низкой плотности ПЭВД

(использование элемента DL для списков определений, разметка полужирным шрифтом и курсивом с помощью элементов физического или логического форматирования)

4. Предварительное форматирование

Виды тары

<i>Вид тары</i>	<i>Конструктивное исполнение</i>	<i>По размерам</i>
Коробки	Неразборная	Малогабаритная
	Разборная	Крупногабаритная

(оформите текст по образцу, используя теги предварительного форматирования)

Вариант 2

1. Ненумерованный список

Алюминиевая фольга в упаковке:

- упаковка для скоропортящихся продуктов;
- фольга для асептической упаковки жидкостей;
- фольга для сигарет;
- фольга для упаковки кофе, детского питания, сухого молока, мучных и кондитерских изделий, чая, пряностей, супов;
- фольга для упаковки масла, маргарина, мороженого, творога и творожных сырков.

(использование элемента UL для ненумерованных списков, маркер элемента списка – диск, разметка заголовка списка полужирным шрифтом с помощью элементов физического или логического форматирования)

2. Нумерованный список

Правильно подобранная упаковка сыра позволяет:

3. улучшить показатели качества сыров, созревание и хранение;
4. увеличить рентабельность производства за счет уменьшения стоимости процессов упаковки, созревания и хранения сыра;

- 5.улучшить товарный вид и привлекательность упаковки;
6.увеличить срок хранения и удобство при транспортировании
(использование элемента OL для нумерованных списков, способ нумерации элементов списка арабские цифры, начиная с 3, разметка заголовка списка полужирным шрифтом с помощью элементов физического или логического форматирования)

3.Список определений

Аббревиатуры для обозначения наиболее распространенных полимеров, используемых в упаковочном производстве:

PVC

Поливинилхлорид **ПВХ**

PVDC

Поливиниденхлорид **ПВДХ**

PP

Полипропилен **ПП**

(использование элемента DL для списков определений, разметка полужирным шрифтом и курсивом с помощью элементов физического или логического форматирования)

4.Предварительное форматирование

Классификация тары

*Функциональное
назначение*

Потребительская

Производственная

Картонная

Бумажная

Неразборная

(оформите текст по образцу, используя теги предварительного форматирования)

Вариант 3

1.Ненумерованный список

Виды упаковки

- первичная
- *вторичная*
- **потребительская**
- транспортная

(использование элемента UL для ненумерованных списков .маркер элемента списка – квадрат, разметка заголовка и элементов списка шрифтом различного начертания с помощью элементов физического или логического форматирования)

2. Нумерованный список

Средства пакетирования:

0. поддоны
1. стяжки
2. пакетирующие стропы
3. обвязки

(использование элемента OL для нумерованных списков, способ нумерации элементов списка арабские цифры, начиная с 0, разметка заголовка списка полужирным шрифтом с помощью элементов физического или логического форматирования)

3. Список определений

Аббревиатуры для обозначения наиболее распространенных полимеров, используемых в упаковочном производстве:

PS

Полистирол *ПС*

PA

Полиамид *ПА*

PC

Поликарбонат *ПК*

(использование элемента DL для списков определений, разметка полужирным шрифтом и курсивом с помощью элементов физического или логического форматирования)

4. Предварительное форматирование

Потребительская тара

*Стекло*нная

*Пластик*овая

Виды тары

для жидкостей

Бутылки

Баллоны

(оформите текст по образцу, используя теги предварительного форматирования)

Вариант 4

1. Ненумерованный список

По конструктивному исполнению упаковка бывает:

- неразборная
- разборная
- складная
- герметичная

(использование элемента UL для ненумерованных списков, маркер элемента списка – диск с переопределением в 3-м элементе на квадрат, разметка заголовка списка полужирным шрифтом с помощью элементов физического или логического форматирования)

2. Нумерованный список

По степени жесткости конструкции различают:

- II. хрупкую упаковку
- III. жесткую упаковку
- IV. мягкую упаковку
- V. полужесткую упаковку

(использование элемента OL для нумерованных списков, способ нумерации элементов списка большие римские цифры, начиная с II, разметка заголовка списка полужирным шрифтом с помощью элементов физического или логического форматирования)

3. Список определений

Аббревиатуры для обозначения наиболее распространенных полимеров, используемых в упаковочном производстве:

PETE или PET

Полиэтилентерафталат – ПЭТФ (ПЭТ)

PVC

Поливинилхлорид ПВХ

PC

Поликарбонат ПК

(использование элемента DL для списков определений, разметка полужирным шрифтом и курсивом с помощью элементов физического или логического форматирования)

4. Предварительное форматирование

Упаковочные материалы

Полимеры	Картон	Бумага
Полистирол		Пергамент
Полипропилен	Гофрированный	Жиронепроницаемая

(оформите текст по образцу, используя теги предварительного форматирования)

Вариант 5

1. Ненумерованный список

Виды мягкой упаковки

- мешки-пакеты
- мешки-конверты

- мешки продуктовые
- мешки технические

(использование элемента UL для нумерованных списков, маркер элемента списка – квадрате переопределением в 3-м элементе на диск, разметка заголовка списка полужирным шрифтом с помощью элементов физического или логического форматирования)

2. Нумерованный список

По форме упаковку различают:

- c. цилиндрическую
- d. конусную
- e. круглую
- f. овальную
- g. сложной конфигурации

(использование элемента OL для нумерованных списков, способ нумерации элементов списка строчные латинские буквы, начиная с "с", разметка заголовка списка полужирным шрифтом с помощью элементов физического или логического форматирования)

3. Список определений

Аббревиатуры для обозначения наиболее распространенных полимеров, используемых в упаковочном производстве:

HDPE

Полиэтилен низкого давления, высокой плотности **ПЭНД**

PVDC

Поливинилденхлорид **ПВДХ**

PP

Полипропилен **ПП**

(использование элемента DL для списков определений, разметка полужирным шрифтом и курсивом с помощью элементов физического или логического форматирования)

4. Предварительное форматирование

Характеристики полимеров

Наименование материала

Характеристика материала

Полиэтилен

Плотность

Полипропилен

Степень усадки

Напряжение усадки

Полистирол

(оформите текст по образцу, используя теги предварительного форматирования)

Лабораторная работа № 5

Разработка WEB-страницы на языке HTML с использованием таблиц

Цель работы: разработать WEB-страницу с использованием таблиц.

Таблицы, в отличие от списков позволяют располагать данные не только по строкам, но и по столбцам. Каждая ячейка таблицы находится в определенной строке и столбце. Таблицы на Web-страницах часто используют для разметки структуры всей страницы. С помощью таблиц можно создать невидимый "каркас" страницы, помогающий расположить текст и изображения так, как вам нравится.

Кроме использования тегов <HTML>, <HEAD>, <TITLE> и <BODY>, которые в соответствии со стандартами HTML и WWW должны присутствовать в каждой WEB-странице, при организации текста внутри WEB-документа с помощью таблиц применяются соответствующие элементы языка HTML с определенным набором тегов и требуемых атрибутов.

Задание

Используя редактор текста Блокнот (Notepad) разработайте страницу, содержащую таблицы по представленному образцу.

Для выполнения данной работы вам необходимо использовать следующие теги элементов языка HTML:

Пример тега	Пояснение
<TABLE> </TABLE>; <CAPTION> </CAPTION>; <TR> </TR>; <TH> </TH>; <TD> </TD>	Оформление таблицы.
атрибуты открывающего тега элемента <TABLE> </TABLE>: border, cellspacing, cellpadding, width, align, bgcolor, border-color, background	Управление параметрами таблицы.
атрибуты : align, valign открывающего тега элементов: <TR> </TR>; <TH> </TH>; <TD> </TD>	Выравнивание текста в ячейках таблицы.
атрибут rowspan открывающего тега элементов: <TH> </TH>; <TD> </TD>	Объединение нескольких строк в одну ячейку.
атрибут colspan открывающего тега элементов: <TH> </TH>; <TD> </TD>	Объединение нескольких столбцов в одну ячейку.

Таблица 1

Классификация тары

Функциональное назначение	Потребительская	Производственная
Картонная	Неразборная	
Бумажная		

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: border=4 width="80%" bgcolor="skyblue" bordercolor="red" cellspacing=3 и атрибутов rowspan и colspan для объединения строк и столбцов, а также атрибутов для выравнивания текста в ячейках таблицы)

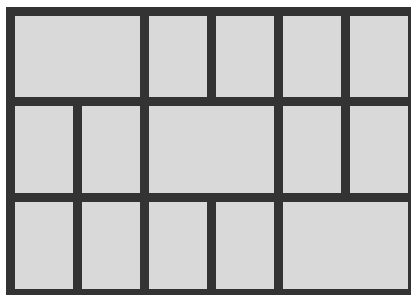
Таблица 2

ЛОГОТИП	баннер
МЕНЮ	Основная часть страницы

Таблица 3

выравнивание слева	выравнивание по центру	выравнивание справа
выравнивание слева по верхнему краю	вертикальное выравнивание по центру	вертикальное выравнивание по нижнему краю

Таблица 4



(использование элементов оформления таблицы для создания таблицы со следующими параметрами: ширина внешней рамки=0; ширина таблицы=200; высота таблицы=200; цвет фона таблицы=#999999; расстояние между границами соседних ячеек=4; расстояние между рамкой каждой ячейки таблицы и содержащимся в ней материалом=0; цвет фона ячеек, переопределенный в теге <TR> =#CCCCCC)

Вариант 2

Создание таблиц

Таблица 1

Мировое потребление (тыс. тонн) основных видов полимерных упаковочных материалов

Страна	Полимер				
	ПВХ	Полиолефины			ПС
		ПЭНД	ПЭВД	ПП	
США	4082	5428	4010	3738	2297
Япония	2028	1621	958	1793	1277
Западная Европа	4745	4982	2807	3369	1611

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: border=5 width="80%" bgcolor="silver" bordercolor="blue" cellspacing=4 и атрибутов rowspan и colspan для объединения строк и столбцов, а также атрибутов для выравнивания текста в ячейках таблицы)

Таблица 2

баннер	ЛОГОТИП
Основная часть страницы	МЕНЮ

Таблица 3

выравнивание по центру	выравнивание слева	выравнивание справа
выравнивание слева по верхнему краю	выравнивание справа по нижнему краю	вертикальное выравнивание по центру

Таблица 4

The diagram shows a table with 6 columns and 3 rows. The cells are shaded gray and separated by thick black borders. The layout is as follows:

- Row 1: 6 cells of varying widths.
- Row 2: 6 cells of varying widths.
- Row 3: 6 cells of varying widths.

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: ширина внешней рамки=0; ширина таблицы=200; высота таблицы=200; цвет фона таблицы=#999999; расстояние между границами соседних ячеек=4; расстояние между рамкой каждой ячейки таблицы и содержащимся в ней материалом=0; цвет фона ячеек, переопределенный в теге <TR> =#CCCCCC)

Создание таблиц

Таблица 1

Сравнение температурного поведения полимерных упаковочных материалов

Класс	Полимер	Температура, °С		
		плавления	разложения	стеклования
Полиолефины	ПЭВД	135	360	
	ПЭНД	110	335	-80
	ПП	170	330	-9

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: border=4 width="80%" bgcolor="lime" bordercolor="red" cellspacing=4 и атрибутов rowspan и colspan для объединения строк и столбцов, а также атрибутов для выравнивания текста в ячейках таблицы)

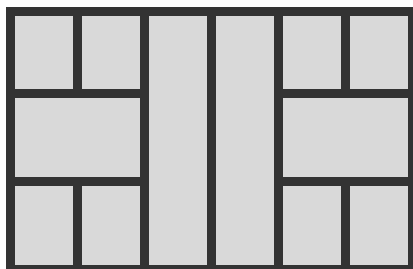
Таблица 2

Основная часть страницы	МЕНЮ
баннер	ЛОГОТИП

Таблица 3

выравнивание справа	выравнивание по центру	выравнивание слева
вертикальное выравнивание по центру	выравнивание слева по верхнему краю	выравнивание справа по нижнему краю

Таблица 4



(использование элементов оформления таблицы для создания таблицы со следующими параметрами: ширина внешней рамки=0; ширина таблицы=200; высота таблицы=200; цвет фона таблицы=#999999; расстояние между границами соседних ячеек=4; расстояние между рамкой каждой ячейки таблицы и содержащимся в ней материалом=0; цвет фона ячеек, переопределенный в теге <TR>=#CCCCCC)

Вариант 4

Создание таблиц

Таблица 1

Внешние признаки полимерных упаковочных материалов

Вид полимера	Внешние признаки			
	Механические	Состояние поверхности на ощупь	Цвет	Прозрачность
ПЭВД	Мягкая, эластичная, стойкая к раздиру	Маслянистая, гладкая	Бесцветная	Прозрачная
ПЭНД	Жестковатая, стойкая к раздиру	Слегка маслянистая, гладкая, слабо шуршащая		Полупрозрачная
ПП	Жестковатая, слегка эластичная, стойкая к раздиру	Сухая, гладкая		Прозрачная или полупрозрачная

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: border=7 width="80%" bgcolor="silver" bordercolor="red" cellspacing=3 и атрибутов rowspan и colspan для объединения строк и столбцов, а также атрибутов для выравнивания текста в ячейках таблицы)

Таблица 2

баннер	ЛОГОТИП
Основная часть страницы	МЕНЮ

Таблица 3

выравнивание слева	выравнивание справа	выравнивание по центру
выравнивание справа по нижнему краю	вертикальное выравнивание по центру	выравнивание слева по верхнему краю

Таблица 4

The diagram shows a rectangular table with a thick black border. The table is divided into a grid of cells. There are 6 columns and 3 rows. The first and last columns are wider than the others. The first and last rows are taller than the others. The central area is a 4x1 grid of cells.

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: ширина внешней рамки=0; ширина таблицы=200; высота таблицы=200; цвет фона таблицы=#999999; расстояние между границами соседних ячеек=4; расстояние между рамкой каждой ячейки таблицы и содержащимся в ней материалом=0; цвет фона ячеек, переопределенный в теге <TR> =#CCCCCC)

Создание таблиц

Таблица 1

Энергопотребление производства 1 тонны упаковок из стекла

Материал	Производственный процесс	Энергопотребление, МДж			
		Электрическая энергия	Нефтепродукты	Другие	Итого
Стекло	Сырье	0,080	2,276	0,942	3,298
	Стекло	1,224	6,195	4,977	12,396
	Итого	1,304	8,471	5,919	15,694

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: border=7 width="80%" bgcolor="aqua" bordercolor="green." cellspacing=4 и атрибутов rowspan и colspan для объединения строк и столбцов, а также атрибутов для выравнивания текста в ячейках таблицы)

Таблица 2

меню	Основная часть страницы
логотип баннер	

Таблица 3

выравнивание слева	выравнивание справа	выравнивание по центру
выравнивание слева по верхнему краю	выравнивание справа по нижнему краю	вертикальное выравнивание по центру

Таблица 4

(использование элементов оформления таблицы для создания таблицы со следующими параметрами: ширина внешней рамки=0; ширина таблицы=200; высота таблицы=200; цвет фона таблицы=#999999; расстояние между границами соседних ячеек=4; расстояние между рамкой каждой ячейки таблицы и содержащимся в ней материалом=0; цвет фона ячеек, переопределенный в теге <TR> =#CCCCCC)

Лабораторная работа № 6

Разработка WEB-страницы на языке HTML с использованием графики, гиперссылок и изображений-карт

Цель работы: разработать WEB-страницу с использованием графики, оформить с помощью гиперссылок гипертекстовые переходы, разработать интерактивную карту-изображение.

Любой Web-документ может содержать ссылки на другие документы или на определенное место текущего или другого документа. Гипертекстовая ссылка – это подчеркнутый и выделенный определенным цветом фрагмент текста, указывающий на ресурс Интернета, файл на локальном диске или на метку внутри страницы документа. При активизации ссылки осуществляется переход по указанному в гиперссылке адресу. В языке HTML гиперссылки создаются с помощью элемента <A>, включающего открывающий тег с атрибутом href, определяющим адрес ссылки. Между открывающим и закрывающим тегами помещается текст или рисунок для щелчка, активизирующий гиперссылку. Шаблон элемента гиперссылки имеет вид: текст для щелчка, отсылающий по адресу ссылки

Для перехода к определенной части страницы документа необходимо поставить в начале этой части страницы метку с помощью тегов: текст Текст не обязателен.

При использовании рисунка вместо видимой части гиперссылки ее шаблон будет иметь вид: ` `

Задание

Используя редактор текста Блокнот (Notepad) разработайте страницу, содержащую:

1. Вставленные изображения с различными типами выравнивания и обтекания текстом.

2. Создайте в текстах HTML-файлов лабораторных работ № 3, 4, 5 гиперссылки к HTML-файлу лабораторной работы № 6. Элементы этих гиперссылок разместите перед закрывающим тегом `</BODY>`.

3. Создайте в тексте текущего HTML-файла метку с именем "top". Элемент этой метки разместите сразу после открывающего тега `<BODY>`.

4. Создайте в тексте текущего HTML-файла гиперссылки В начало к началу документа, то есть к метке "top". Элемент этой гиперссылки разместите после вставленных изображений.

5. С помощью тега `` вставьте рядом с текстом созданной гиперссылки графическое изображение кнопки. Создайте графическую гиперссылку на это изображение к началу документа.

6. Создайте гиперссылки к разделам вашей страницы:

Обтекание рисунков текстом

Изображения – гиперссылки

Изображения-карты

Разместите эти гиперссылки после элемента метки с именем "top" в начале страницы.

7. Используя графический редактор Paint создайте графический файл (по образцу), содержащий изображение прямоугольников и круга, которые будут использоваться в качестве указателей для перехода по гиперссылкам к Вашим файлам лабораторных работ № 3, № 4, № 5 и на главную страницу БНТУ (<http://www.bntu.by>). Запишите необходимые для описания изображения-карты координаты созданных прямоугольника и круга. Сохраните созданный файл с именем "menu" и расширением gif, то есть как menu.gif в текущем каталоге или в папке рисунков image.

8. Используя элементы `<MAP name="karta"><AREA></MAP>` и `` с необходимыми атрибутами сделайте созданный Вами графический файл изображением-картой. Проверьте правильность переходов к файлам лабораторных работ и на страницу БНТУ при использовании в качестве гиперссылок фрагментов созданного Вами изображения-карты.

9. Сохраните страницу в рабочем каталоге.

Для выполнения данной работы вам необходимо использовать следующие теги элементов языка HTML:

Пример элемента	Пояснение
<code> текст для щелчка</code>	Элемент гиперссылки.
<code></code>	Элемент метки.
<code> текст для щелчка</code>	Элемент гиперссылки для перехода к метке. Для перехода к метке текущего файла его имя не указывается.
<code> </code>	Элемент гиперссылки при использовании изображения вместо текста гиперссылки.
<code><MAP name="Имя карты"><AREA></MAP></code>	Элемент изображения-карты.
атрибуты элемента <code><AREA></code> : <code>href, alt, shape, coords.</code>	Задание адреса ссылки, альтернативного текста, формы и координат областей изображения-карты.
атрибут элемента <code></code> : <code>usemap="#Имя карты".</code>	Придает элементу <code></code> , который используется для вставки на WEB-страницу графического файла изображения-карты, свойства гиперссылки, то есть возможность осуществления переходов к нескольким документам, используя отдельные фрагменты изображения в качестве указателей переходов.

Вариант 1

Использование графики, карт и гиперссылок

[Обычные изображения и обтекание рисунков текстом](#)

[Использование изображений в качестве гиперссылок](#)

[Использование изображений – карт в качестве гиперссылок](#)

Обычные изображения и обтекание рисунков текстом



Тег `` может иметь собственный атрибут `ALIGN=`, который управляет расположением рисунка относительно текста. `ALIGN="left"` — текст обтекает рисунок, который располагается слева.

Чтобы рисунок не слишком “прижимался” к тексту, можно оставить меж-



ду ними немного “воздуха”.

Для этого в теге `` используют атрибуты `HSPACE=` и `VSPACE=`, значения которых задают соответственно горизонтальное и вертикальное “поля” рисунка.

[в начало](#) [Home](#)

Использование изображений в качестве гиперссылок

В текст веб-страницы можно вставлять гиперссылки трех типов:

- ссылки в другие места той же страницы;
- ссылки на другие страницы этого же сайта или сервера;
- ссылки на страницы, расположенные на любом другом сервере в Интернете.

Ссылками может быть текст, изображение и часть изображения (карта)

[в начало](#) [Home](#)

Использование изображений - карт в качестве гиперссылок



[в начало](#) [Home](#)

Вариант 2

Использование графики, карт и гиперссылок

[Обычные изображения и обтекание рисунков текстом](#)

[Использование изображений в качестве гиперссылок](#)

[Использование изображений – карт в качестве гиперссылок](#)

Обычные изображения и обтекание рисунков текстом

Тег `` может иметь собственный атрибут `ALIGN=`, который управляет расположением рисунка относительно текста. `ALIGN="right"` — текст обтекает рисунок, который располагается справа.



Чтобы рисунок не слишком “прижимался” к тексту, можно оставить между ними немного “воздуха”. Для этого в теге `` используют атрибуты `HSPACE=` и `VSPACE=`, значения которых задают соответственно горизонтальное и вертикальное “поля” рисунка.

[в начало](#) [Home](#)

Использование изображений в качестве гиперссылок

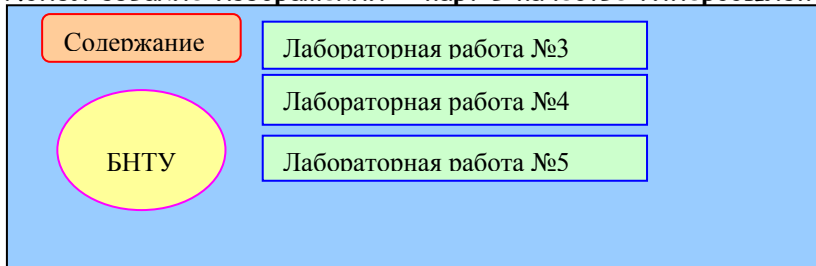
В текст веб-страницы можно вставлять гиперссылки трех типов:

- ссылки в другие места той же страницы;
- ссылки на другие страницы этого же сайта или сервера;
- ссылки на страницы, расположенные на любом другом сервере в Интернете.

Ссылками может быть текст, изображение и часть изображения (карта)

[в начало](#) [Home](#)

Использование изображений – карт в качестве гиперссылок



[в начало](#) [Home](#)

Вариант 3

Использование графики, карт и гиперссылок

[Обычные изображения и обтекание рисунков текстом](#)

[Использование изображений в качестве гиперссылок](#)

[Использование изображений – карт в качестве гиперссылок](#)

Обычные изображения и обтекание рисунков текстом

ALIGN="top" — рисунок строки, и его верхняя граница букв текста (можно также при-



полагается внутри текстовой строки, и его верхняя граница совпадает с верхней границей букв текста (можно также при-

Чтобы рисунок не слишком

“прижимался” к тексту, мож-



но оставить между ними немного “воздуха”.

Для

этого в теге используют атрибуты HSPACE= и VSPACE=, значения которых задают соответственно горизонтальное и вертикальное “поля” рисунка.

[В начало](#) [Home](#)

Использование изображений в качестве гиперссылок

В текст веб-страницы можно вставлять гиперссылки трех типов:

осылки в другие места той же страницы;

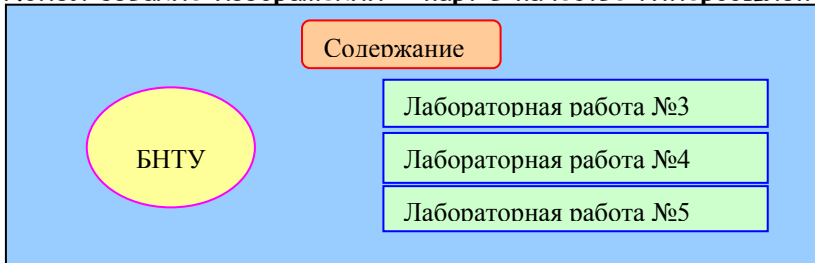
осылки на другие страницы этого же сайта или сервера;

осылки на страницы, расположенные на любом другом сервере в Интернете.

Ссылками может быть текст, изображение и часть изображения (карта)

[В начало](#) [Home](#)

Использование изображений – карт в качестве гиперссылок



[В начало](#) [Home](#)

Вариант 4

Использование графики, карт и гиперссылок

[Обычные изображения и обтекание рисунков текстом](#)

[Использование изображений в качестве гиперссылок](#)

[Использование изображений – карт в качестве гиперссылок](#)

Обычные изображения и обтекание рисунков текстом

ALIGN="middle" —



рисунок располагается

внутри текстовой строки, которая вертикально выравнивается по центру рисунка (можно также при менять значение "absmiddle")

Чтобы рисунок не слишком “прижимался” к тексту, можно оставить между ними немного “воздуха”. Для этого в теге используют атрибуты HSPACE= и VSPACE=, значения которых задают соответственно горизонтальное и вертикальное “поля” рисунка.



[в начало](#) [Home](#)

Использование изображений в качестве гиперссылок

В текст веб-страницы можно вставлять гиперссылки трех типов:

- ссылки в другие места той же страницы;
- ссылки на другие страницы этого же сайта или сервера;
- ссылки на страницы, расположенные на любом другом сервере в Интернете.

Ссылками может быть текст, изображение и часть изображения (карта)

[в начало](#) [Home](#)

Использование изображений – карт в качестве гиперссылок



[в начало](#) [Home](#)

Вариант 5

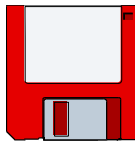
Использование графики, карт и гиперссылок

[Обычные изображения и обтекание рисунков текстом](#)

[Использование изображений в качестве гиперссылок](#)

[Использование изображений – карт в качестве гиперссылок](#)

Обычные изображения и обтекание рисунков текстом



ALIGN="bottom" – рисунок располагается внутри текстовой строки, и его нижняя граница совпадает с нижней границей букв текста (можно также применять значение "absbottom" или "baseline").

Чтобы рисунок не слишком “прижимался” к тексту, можно оставить между ними немного “воздуха”. Для этого в теге используют атрибуты HSPACE= и VSPACE=, значения которых задают соответственно горизонтальное и вертикальное “поля” рисунка.



[в начало](#) [Home](#)

Использование изображений в качестве гиперссылок

В текст веб-страницы можно вставлять гиперссылки трех типов:

- ссылки в другие места той же страницы;
- ссылки на другие страницы этого же сайта или сервера;
- ссылки на страницы, расположенные на любом другом сервере в Интернете.

Ссылками может быть текст, изображение и часть изображения (карта)

[в начало](#) [Home](#)

Использование изображений – карт в качестве гиперссылок



[в начало](#) [Home](#)

Лабораторная работа № 7

Использование форм при разработке WEB-страницы на языке HTML

Цель работы: разработать WEB-страницу с использованием вложенных списков разных типов.

Формы предназначены для размещения ответов на адресованные автором страницы посетителям вопросы. Как и любые бумажные формы они используются, в частности, для:

- анкетирования посетителей WEB-сайта или страницы
- выяснения спроса на предлагаемые товары или услуги
- выяснения цен на интересующие товары или услуги
- получения отзывов

Форма должна содержать определенное разработчиком страницы число элементов управления: поля ввода, кнопки, переключатели, флажки, меню и другие. Каждый элемент управления создается при помощи одного из соответствующих элементов языка HTML с необходимыми атрибутами, определяющими особенности работы той или иной формы.

Для выполнения данной работы вам необходимо использовать следующие элементы языка HTML:

Примеры элементов и их атрибутов	Пояснение
<FORM>...</FORM>	Элемент, определяющий область формы на WEB-странице.
Атрибуты элемента <FORM>...</FORM>:	
action	Определяет протокол доступа к серверу программы-обработчика формы, адрес и имя этой программы.
method	Атрибут, задающий метод ввода данных на сервере программы-обработчика формы.
enctype	Атрибут, указывающий способ кодирования содержимого формы для передачи программе-обработчику.
<INPUT>	Элемент позволяет создавать различные части форм, такие как <i>флажки, переключатели, поля ввода</i> .

атрибуты элемента <INPUT>:	
name	Значение атрибута определяет имя поля формы, то есть имя блока данных, введенных в это поле.
value	Ввод текста, определение надписи на кнопке.
checked	Показывает, что флажок должен быть установлен по умолчанию. В отличие от переключателей, одновременно может быть установлено любое количество флажков. Не имеет значения.
size	Задаёт длину поля ввода. Длина выражается в <u>символах</u> .
readonly	Позволяет создать элемент, недоступный для редактирования.
type	определяет вид элемента INPUT.

Значения атрибута type элемента <u>INPUT</u> :	
Значения атрибута	Пояснение
type=" text"	Создание поля ввода, в котором можно автоматически разместить произвольный текст, используя атрибут value.
type=" password"	Создание поля для ввода пароля. Введенная информация отображается звездочками.
type=" checkbox"	Создание флажка. Для нескольких флажков необходимо использовать соответствующее количество элементов <INPUT>.
type=" radio"	Определение одного переключателя из группы. Для всей группы переключателей необходимо использовать соответствующее количество элементов <INPUT.> Если один из переключателей должен быть выбран по умолчанию, то в элемент <INPUT> этого переключателя вставляется атрибут checked без значения.
type=" button"	Создание кнопки произвольного назначения.
type=" submit"	Создание кнопки, щелчок на которой подтверждает ввод информации в форму. Атрибут value используется для определения надписи на кнопке.
type=" reset"	Кнопка для отмены ввода данных в форму.

type=" image"	Создание кнопки с рисунком. Для указания графического файла используется атрибут src.
type=" file"	Средство выбора файла для присоединения к форме. Имя файла записывается в поле ввода. Кроме того, браузер автоматически создает рядом с полем ввода кнопку Обзор, которая позволяет запустить стандартный для операционной системы диалог выбора файлов.

Элементы, которые могут использоваться самостоятельно
(не в составе элемента <FORM>...</FORM>:)

Примеры элементов и их атрибутов	Пояснение
<SELECT> <OPTION> </SELECT>	Элемент SELECT используется для создания списка или меню, а элемент OPTION – для создания пунктов списка. Количество элементов OPTION определяет число пунктов списка.
Атрибуты элемента <SELECT>...</SELECT>:	
name	Определяет имя меню.
multiple	Разрешает выбрать сразу несколько пунктов списка (при нажатой клавише Ctrl). Атрибут не имеет значения.
size	Определяет количество видимых на экране пунктов списка. По умолчанию видна только первая строка, а при щелчке на кнопке со стрелкой раскрывается полный список.
Атрибуты элемента <OPTION>:	
selected	Определяет, какой из пунктов списка должен быть выбран по умолчанию. Не имеет значений. Может быть присвоен только одному пункту списка.

value	Значение атрибута в паре с именем меню (значением атрибута name) передается на сервер для обработки формы.
<TEXTAREA>...</TEXTAREA>	Элемент TEXTAREA используется для создания области фиксированного размера для ввода или просмотра текста неограниченного размера.
Атрибуты элемента <TEXTAREA>...</TEXTAREA>:	
name	Определяет имя области.
rows	Определяет количество строк.
cols	Определяет количество столбцов (в символах).

Задание

Используя редактор текста Блокнот (Notepad):

1. Создайте WEB-страницу, содержащую в составе элемента <FORM>...</FORM> элементы форм <INPUT> всех типов.
2. Создайте на WEB-странице элементы форм <SELECT> <OPTION> </SELECT> и <TEXTAREA>...</TEXTAREA> .
3. Создайте на WEB-странице форму с произвольным набором элементов.

Лабораторная работа № 8

Использование фреймов при разработке WEB-страницы на языке HTML

Цель работы: разработать WEB-страницу с использованием фреймовой верстки.

При создании страницы с фреймами разрабатывается несколько HTML-файлов, которые отличаются по своему назначению. Документ (файл) раскладки определяет структуру окна браузера, то есть количество и вид областей, на которые разделено это окно. Необхо-

димое количество областей и их вид (горизонтальные или вертикальные или комбинация) определяется разработчиком WEB-страницы. Документы (файлы) содержания предназначены для заполнения информацией каждой из областей.

Для выполнения данной работы вам необходимо использовать следующие элементы языка HTML (в дополнение к использованному ранее):

Примеры элементов и их атрибутов	Пояснение
<code><FRAMESET><FRAME> </FRAMESET></code>	Элемент для создания WEB-страницы с фреймами.
атрибуты элемента <code><FRAMESET>...</FRAMESET></code> :	
rows	Создание горизонтальных областей деления окна браузера.
cols	Создание вертикальных областей деления окна браузера.
frameborder	Задание рамки фрейма.
framespacing	Задание толщины рамки фрейма.
<code><FRAME></code>	Элемент, определяющий содержание и структуру каждой области деления окна браузера.
атрибуты элемента <code><FRAME></code> :	
name	Имя фрейма.
src	Ссылка на файл документа содержания, помещаемого в данную область.
noresize	Запрет на перемещение границ фрейма при помощи мыши.
scrolling	Управление прокруткой внутри области фрейма.
marginheight	Величина отступа содержимого фрейма от верхней и нижней границы фрейма.
marginwidth	Величина левого и правого полей содержимого фрейма.
<code> Текст меню</code>	Элемент гиперссылки с использованием атрибута target.

Задание

Используя редактор текста Блокнот (Notepad) разработайте WEB-страницу с фреймами и организацией загрузки HTML-файлов лабораторных работ № 3–7 в один из фреймов с помощью выбора пунктов меню в другом фрейме в соответствии с указаниями варианта задания.

Вариант 1

Создать WEB-страницу по предложенному образцу и организацией загрузки HTML-файлов лабораторных работ № 3–7 в правый фрейм с помощью выбора пунктов меню в левом фрейме.

- 1.Создайте файл с оглавлением Вашего документа в виде списка ссылок на HTML-файлы лабораторных работ № 3, 4, 5, 6, 7. Сохраните созданный файл с именем menu.html
- 2.Создайте страницу above.html, в которой разместите краткую информацию о себе.
- 3.Используя шаблон элемента FRAMESET создайте HTML-файл документа раскладки основной страницы с именем index.html, содержащий описание двух вертикально расположенных в окне браузера фреймов и одного горизонтального. С помощью атрибута cols задайте ширину левой вертикальной области в три раза уже правой и нижней области в 10 % от экрана. С помощью атрибутов frameborder и framespacing задайте рамку между фреймами и ее толщину.
- 4.Левому фрейму с помощью атрибута name присвойте имя "menu", с помощью атрибута src укажите имя созданного файла оглавления – menu.html, который будет загружаться в этот фрейм. Правому фрейму с помощью атрибута name присвойте имя "info", с помощью атрибута src укажите имя файла первого документа – **Имя файла лабораторной работы 3.html**, который будет загружаться в этот фрейм. Нижнему фрейму – name – "autor", src – about.html
- 5.Посмотрите на HTML-файл своей страницы – index.html с помощью установленного на компьютере броузера и добейтесь максимального подобию в представлении страницы броузером.
- 6.Проверьте правильность переходов к файлам лабораторных работ при использовании гиперссылок.

<p>Содержание</p> <p><u>Лаб. 3.</u></p> <p><u>Лаб. 4.</u></p> <p><u>Лаб. 5.</u></p> <p><u>Лаб. 6.</u></p> <p><u>Лаб. 7.</u></p>	
Об авторе:	

Вариант 2

Создать WEB-страницу по предложенному образцу и организацией загрузки HTML-файлов лабораторных работ № 3–7 в средний фрейм с помощью выбора пунктов меню в верхнем фрейме.

1. Создайте файл с оглавлением Вашего документа в виде списка ссылок на HTML-файлы лабораторных работ № 3, 4, 5, 6, 7. Сохраните созданный файл с именем menu.html
2. Создайте страницу above.html, в которой разместите краткую информацию о себе.
3. Используя шаблон элемента FRAMESET создайте HTML-файл документа раскладки основной страницы с именем index.html, содержащий описание двух вертикально расположенных в окне браузера фреймов и одного горизонтального. С помощью атрибута rows задайте высоту верхней области 30 %, ширину нижней области в 10 %, для средней – все, что останется. С помощью атрибутов frameborder и framespacing задайте рамку между фреймами и ее толщину.
4. Верхнему фрейму с помощью атрибута name присвойте имя "menu", с помощью атрибута src укажите имя созданного файла оглавления – menu.html, который будет загружаться в этот фрейм. Правому фрейму с помощью атрибута name присвойте

имя "info", с помощью атрибута src укажите имя файла первого документа – **Имя файла лабораторной работы 3.html**, который будет загружаться в этот фрейм. Нижнему фрейму – name – "autor", src – about.html

- 5.Посмотрите на HTML-файл своей страницы – index.html с помощью установленного на компьютере браузера и добейтесь максимального подобия в представлении страницы браузером.
- 6.Проверьте правильность переходов к файлам лабораторных работ при использовании гиперссылок.

Вид страницы:

Содержание <u>Лаб. 3.</u> <u>Лаб. 4.</u> <u>Лаб. 5.</u> <u>Лаб. 6.</u> <u>Лаб.7.</u>
Об авторе:

Вариант 3

Создать WEB-страницу по предложенному образцу и организацией загрузки HTML-файлов лабораторных работ № 3–7 в правый фрейм с помощью выбора пунктов меню в левом фрейме.

- 1.Создайте файл с оглавлением Вашего документа в виде списка ссылок на HTML-файлы лабораторных работ № 3, 4, 5, 6, 7. Сохраните созданный файл с именем menu.html
- 2.Создайте страницу above.html, в которой разместите краткую информацию о себе.
- 3.Используя шаблон элемента FRAMESET создайте HTML-файл документа раскладки основной страницы с именем index.html,

содержащий описание двух вертикально расположенных в окне браузера фреймов и одного горизонтального. С помощью атрибута rows задайте ширину верхней области в 10 % от нижней. С помощью атрибута cols задайте ширину левой вертикальной области в три раза уже правой. С помощью атрибутов frameborder и framespacing задайте рамку между фреймами и ее толщину.

4. Верхнему фрейму с помощью атрибута name присвойте имя "menu", с помощью атрибута src укажите имя созданного файла оглавления – menu.html, который будет загружаться в этот фрейм. Правому фрейму с помощью атрибута name присвойте имя "info", с помощью атрибута src укажите имя файла первого документа – **Имя файла лабораторной работы 3.html**, который будет загружаться в этот фрейм. Нижнему фрейму – name – "autor", src – about.html
5. Посмотрите на HTML-файл своей страницы – index.html с помощью установленного на компьютере браузера и добейтесь максимального подобия в представлении страницы браузером.
6. Проверьте правильность переходов к файлам лабораторных работ при использовании гиперссылок.

Вид страницы:

Об авторе:	
Содержание	
Лаб. 3.	
Лаб. 4.	
Лаб. 5.	
Лаб. 6.	
Лаб. 7.	

Вариант 4

Создать WEB-страницу по предложенному образцу и организацией загрузки HTML-файлов лабораторных работ № 3–7 в левый фрейм с помощью выбора пунктов меню в правом фрейме.

- 1.Создайте файл с оглавлением Вашего документа в виде списка ссылок на HTML-файлы лабораторных работ № 3, 4, 5, 6, 7. Сохраните созданный файл с именем menu.html
- 2.Создайте страницу above.html, в которой разместите краткую информацию о себе.
- 3.Используя шаблон элемента FRAMESET создайте HTML-файл документа раскладки основной страницы с именем index.html, содержащий описание двух вертикально расположенных в окне браузера фреймов и одного горизонтального. С помощью атрибута cols задайте ширину правой вертикальной области в три раза уже левой. С помощью атрибута rows задайте ширину нижней области в 10 % от верхней. С помощью атрибутов frameborder и framespacing задайте рамку между фреймами и ее толщину.
4. Верхнему фрейму с помощью атрибута name присвойте имя "menu", с помощью атрибута src укажите имя созданного файла оглавления – menu.html, который будет загружаться в этот фрейм. Правому фрейму с помощью атрибута name присвойте имя "info", с помощью атрибута src укажите имя файла первого документа – **Имя файла лабораторной работы** 3.html, который будет загружаться в этот фрейм. Нижнему фрейму – name – "autor", src – about.html
- 5.Посмотрите на HTML-файл своей страницы – index.html с помощью установленного на компьютере браузера и добейтесь максимального подобия в представлении страницы браузером.
- 6.Проверьте правильность переходов к файлам лабораторных работ при использовании гиперссылок.

Вид страницы:

	<p>Содержание</p> <p><u>Лаб. 3.</u> <u>Лаб. 4.</u> <u>Лаб. 5.</u> <u>Лаб. 6.</u> <u>Лаб. 7.</u></p>
<p>Об авторе:</p>	

Вариант 5

Создать WEB-страницу по предложенному образцу и организацией загрузки HTML-файлов лабораторных работ № 3–7 в средний фрейм с помощью выбора пунктов меню в нижнем фрейме.

- 1.Создайте файл с оглавлением Вашего документа в виде списка ссылок на HTML-файлы лабораторных работ № 3, 4, 5, 6, 7. Сохраните созданный файл с именем menu.html
- 2.Создайте страницу above.html, в которой разместите краткую информацию о себе.
- 3.Используя шаблон элемента FRAMESET создайте HTML-файл документа раскладки основной страницы с именем index.html, содержащий описание двух вертикально расположенных в окне браузера фреймов и одного горизонтального. С помощью атрибута rows задайте высоту верхней области 10 %, ширину нижней области в 30 %, для средней – все, что останется. С помощью атрибутов frameborder и framespacing задайте рамку между фреймами и ее толщину.
4. Верхнему фрейму с помощью атрибута name присвойте имя "menu", с помощью атрибута src укажите имя созданного файла оглавления – menu.html, который будет загружаться в этот

фрейм. Правому фрейму с помощью атрибута name присвойте имя "info", с помощью атрибута src укажите имя файла первого документа – **Имя файла лабораторной работы 3.html**, который будет загружаться в этот фрейм. Нижнему фрейму – name – "autor", src – about.html

- 5.Посмотрите на HTML-файл своей страницы – index.html с помощью установленного на компьютере броузера и добейтесь максимального подобия в представлении страницы броузером.
- 6.Проверьте правильность переходов к файлам лабораторных работ при использовании гиперссылок.

Вид страницы:

Об авторе:
Содержание <u>Лаб. 3.</u> <u>Лаб. 4.</u> <u>Лаб. 5.</u> <u>Лаб. 6.</u> <u>Лаб.7.</u>

Лабораторная работа № 9

Технология CSS. Разработка цвето-шрифтовой схемы и навигации сайта с помощью каскадных таблиц стилей

Цель работы: Изучить основы технологии CSS на примере разработки собственного сайта.

Технология CSS

Каскадные таблицы стилей или CSS (от английского Cascading Style Sheets) являются следствием дальнейшего развития HTML и дают нам возможность перейти на следующий уровень представления информации. Таблицы стилей позволяют разделить смысловое содержимое странички и его оформление.

Подключение таблиц стилей CSS к HTML-документу

Для осуществления этой задачи используют 3-х метода:

- In-line-описание,
- описание в секции заголовка.
- внешний файл,

Метод 1: Инлайн/In-line (атрибут style)

или описание, встроенное в тег:

```
<p style="color:red; text-align:center;">  
Этот текст переопределен стилем  
</p>
```

Метод 2: Описание в секции заголовка (тэг style)

Его действие распространяется на всю страничку. Определение стилей происходит при помощи классов, которые представляют собой списки с определением всех необходимых параметров оформления.

При использовании этого метода описание стилей необходимо разместить в секции заголовка:

```
<head> ....  
<style type="text/css">  
<!--  
.header {  
text-align :center;  
font-size : 27pt;}  
.red {color : red; }  
-->
```

```
</style>
</head>
```

Теперь эти стили можно применять в любом месте html-кода. Для этого используется следующая конструкция:

```
<p class=header>Этот текст написан стилем
header<p>
<p class=red>Этот текст написан красным
цветом<p>
```

Кроме определения новых классов мы также имеем возможность переопределять стандартные теги. Например, тег <p>:

```
<style type="text/css">
<!-- p { text-align : center; font-size
:12pt;}
--> </style>
```

Теперь весь текст, заключенный в теги <p></p>, будет выглядеть так, как определено данным стилем.

Метод 3:Вынесение описания стилей во внешний файл (ссылка на таблицу стилей).

Диапазон его воздействия простирается на все файлы, в которые включено описание. В случае, если нам потребуется изменить внешний вид сайта, то будет достаточно скорректировать лишь один этот файл.

Для начала создается стилевой файл с описанием всех нужных нам классов (mystyle.css):

```
.header { text-align : center; font-size :
27pt;}
.red { color :red; }
p { text-align : center; font-size : 12pt;}
```

А потом ссылка на него внедряется в документ при помощи тега <link>:

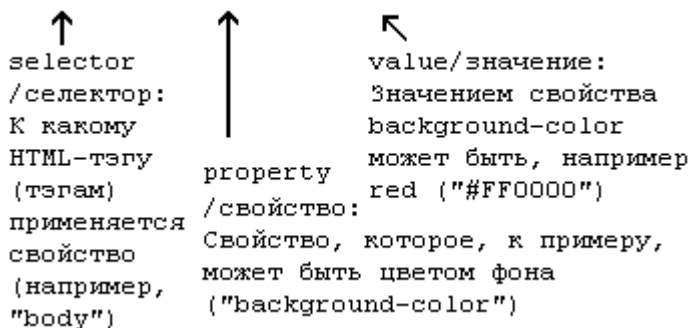
```
<head>
....
<link rel="stylesheet" type="text/css"
href="css/mystyle.css" title="MyStyleSheet">
....
</head>
```

Это самый удобный способ, и для основной таблицы стилей рекомендуется пользоваться именно им.

Синтаксис CSS

Общее описание синтаксиса CSS:

```
selector {property: value;}
```



В качестве селектора может выступать тег HTML, класс или псевдокласс.

Описание каждого класса делается при помощи конструкции, подобной этой:

```
.small { font-size: 9pt; }
```

Сначала указывается имя класса - оно может быть произвольным, но желательно все-таки давать осмысленное название. Далее, в фигурных скобках перечисляются все необходимые параметры для данного класса. Параметры отделяются друг от друга точкой с запятой.

Заметьте, что имя класса начинается с точки и таким образом определяет универсальный класс, т.е. такой, который может быть применен к любому тегу. И делается это при помощи следующей конструкции:

```
<p class=small>Накладываем стиль на этот текст</p>
```

Существуют универсальные классы и, так называемые, теговые классы:

```
p.small { font-size: 9pt; }
```

Класс, определенный таким образом, сработает только в том теге, для которого он предназначен, а для всех остальных будет проигнорирован.

Мы можем определять параметры не только для одного тега, но и сразу для нескольких. Для этого в определении стиля достаточно перечислить их через запятую:

```
p, td { font-size: 9pt; color:green; }
```

Такой прием называется группировкой, и в данном случае мы определили и для <p>, и для <td> одинаковый размер и цвет текста.

В случае переопределения существующих тегов, в описании стиля можно указывать не все параметры, а лишь те из них, которые мы хотим изменить. Все остальные параметры примут значения по умолчанию, которые для разных тегов различны.

Псевдоклассы

В CSS есть такое понятие как псевдокласс. В отличие от обычного класса, действие псевдокласса распространяется не на весь текст, к которому применен данный стиль, а лишь на его часть и возможно лишь в определенном состоянии. Чтобы было понятнее, давайте разберем эффект, при котором ссылки подчеркиваются лишь при наведении на них курсора. Эффект достаточно распространенный, и его можно наблюдать в том числе и на этом сайте. Вот фрагмент таблицы стилей, который позволяет достигать вышеописанного эффекта:

```
a { text-decoration: none; }  
a:hover { text-decoration: underline; }
```

Верхняя строчка - это переопределение стандартного тега <a>, которое запрещает подчеркивать ссылки, а вот нижняя - это определение стиля для псевдокласса hover, который описывает стиль ссылки в момент, когда курсор находится над ней.

Пример псевдокласса - определение буквы в начале абзаца:

```
p:first-letter { font-size: 200\%; font-weight: bold; }
```

И в другом случае действие стиля распространяется либо на определенное состояние (пользователь собирается щелкнуть по ссылке), либо на фрагмент текста (изменяется только первая буква абзаца). В этом и заключается смысл псевдоклассов.

При создании таблицы стилей можно пользоваться комментариями.:

```
/* Этот текст является комментарием */
```

Основные параметры шрифта

- font-weight: [bold|normal|number] - жирность шрифта

- font-style: [normal|italic|oblique] - наклон шрифта
- font-size: number - размер шрифта
- font-family: name - гарнитура шрифта
- color:number - цвет шрифта
- background-color: number - цвет подложки
- background: url – изображение в качестве подложки

псевдоклассы Ссылок

- A:active {} Таблица стилей для активных ссылок (при нажатии)
- A:link {} Таблица стилей для собственно ссылок
- A:visited {} Таблица стилей для посещённых ссылок
- A:hover {} Таблица стилей для ссылок при наведении указателя мыши

Основные параметры абзаца

- text-align: [left|right|center|justify] - выравнивание
- text-indent: number - отступ красной строки
- line-height: number –межстрочное расстояние
- letter-spacing: number – интервал между буквами

Единицы измерения в CSS

В свойствах, которым требуется указание размеров, можно использовать несколько способов для их задания:

- относительный размер в процентах (%)
- относительный размер при помощи словесного описания (larger, smaller, xx-small, x-small, small, medium, large, x-large, xx-large)
- абсолютный размер в типографских единицах - размер может задаваться в пунктах (pt), пиках (pc), пикселях (px), средней шириной буквы "m" (em), средней шириной буквы "x" (ex)
- абсолютный размер в стандартных единицах длины - размер может задаваться в сантиметрах (cm), миллиметрах (mm), дюймах (in) абсолютный в пикселях (px)

Задание цвета в CSS

Цвет для тех свойств, где это нужно, может быть определен одним из трех способов:

- при помощи названия цвета: yellow, red, green, grey,...
- шестнадцатеричным заданием цвета в формате #RRGGBB: #ff0000, #883490, #ffffff,...
- десятичным заданием составляющих цвета в формате rgb(red, green, blue): rgb(255,0,0), rgb(100,23,78),...

Примеры описания таблицы стилей:

```
.epigraph {
    font-size: 12pt;
    font-style: italic;
    text-align: right;
    color: rgb(127,127,0);
}

p.big {
    font-size: 16px;
    font-weight: bold;
    color: #ff0000;
}

.menu {
    font-weight: bold;
    font-size: 9pt;
    font-family: arial, helvetica, sans-serif;
}

a:hover {
    color: #b63a3a;
    text-decoration: none;
}
```

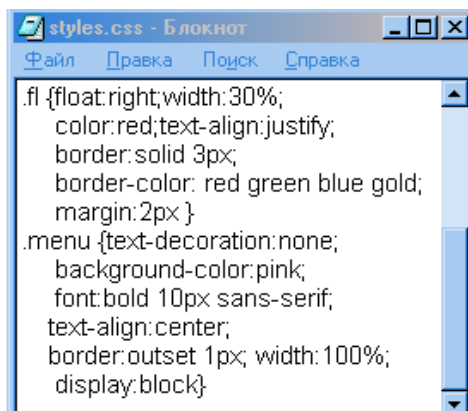
Задание

1. Создать внешнюю таблицу стилей (файл с расширением CSS).
2. Разработайте цвето-шрифтовую схему сайта, которая будет повторяться на всех его страницах.

В таблице стилей должны присутствовать следующие элементы оформления:

- цвет фона страницы
- фоновое изображение на странице
- стилевое оформление абзацев
- контекстные селекторы для оформления слов в абзацах полужирным, курсивным начертанием
- стилевое оформление гиперссылок

- стилевое оформление рамок таблиц
3. Подключить внешнюю таблицу стилей ко всем лабораторным работам (в тег <LINK>)
 4. Для страницы с содержанием переопределить стиль для оформления меню (гиперссылки) с использованием псевдоклассов. Для создания "вместилища" панели используйте div. Примерные стили для вместилища и для ссылок даны на рисунке.



```
styles.css - Блокнот
Файл  Правка  Поиск  Справка
.fl {float:right;width:30%;
    color:red;text-align:justify;
    border:solid 3px;
    border-color: red green blue gold;
    margin:2px }
.menu {text-decoration:none;
    background-color:pink;
    font:bold 10px sans-serif;
    text-align:center;
    border:outset 1px; width:100%;
    display:block}
```

5. Создайте отдельный класс для выделения той гиперссылки, которая указывает на текущую страницу. Это позволит пользователю по виду панели определить, на какой странице он находится.
6. Для страницы со списками определить стиль для маркированных списков, в котором для маркера используется изображение. Изображение найти в стандартной коллекции Windows.

Лабораторная работа № 10

Технология CSS. Проектирование оформления web-сайта с помощью каскадных таблиц стилей

Цель работы: Изучить основы технологии CSS на примере разработки собственного сайта.

Основные параметры границ (рамок)

- border-width: number – толщина рамки
- border-color: number – цвет рамки
- border-style: [dotted|dashed|solid|double|groove|ridge|inset|outset] – тип рамки

Основные параметры блоков (элементов типа "Box")

- padding-left: number – отступ от текста слева
- padding-right: number – отступ от текста справа
- padding-top: number – отступ от текста сверху
- padding-bottom: number – отступ от текста снизу
- margin-left: number – отступ от границы слева
- margin-right: number – отступ от границы справа
- margin-top: number – отступ от границы сверху
- margin-bottom: number – отступ от границы снизу

Свойство writing-mode:tb-rl позволяет размещать текст под углом 90°.

Задание

Задание 1

В лабораторной работе № 3 для всех абзацев с помощью стилей переопределите в секции заголовка (тег style) тег <p> и псевдо-классы тега <p>:

- выравнивание абзаца
- отступ красной строки-
- размер и цвет первой буквы

Задание 2

В лабораторной работе № 4 с помощью стилей встроенных в теги, оформить нумерованные списки с помощью маркеров-рисунков из стандартной коллекции Windows.

Нумерованные списки оформить с помощью стилей, встроенных в теги.

Задание 3

В работе № 6 задайте с помощью стилей обтекание рисунков текстом в соответствии с вариантом.

Задание 4

В лабораторную работу № 5 добавьте таблицы, оформленные с помощью стилей.

Таблица 1

Текст				
Текст				

Таблица 2

Текст				
Текст				

Таблица 3

График			
2008	2009	2010	Показатель
50	25	100	

Задание 5

1. Добавьте в документ следующий HTML-код:

```
<html>  
<body>  
  <span>ВНИМАНИЕ!</span>  
</body>  
</html>
```

Определите стили таким образом, чтобы страница выглядела так, как показано ниже: (Новые HTML-теги добавлять нельзя!)

ВНИМАНИЕ!

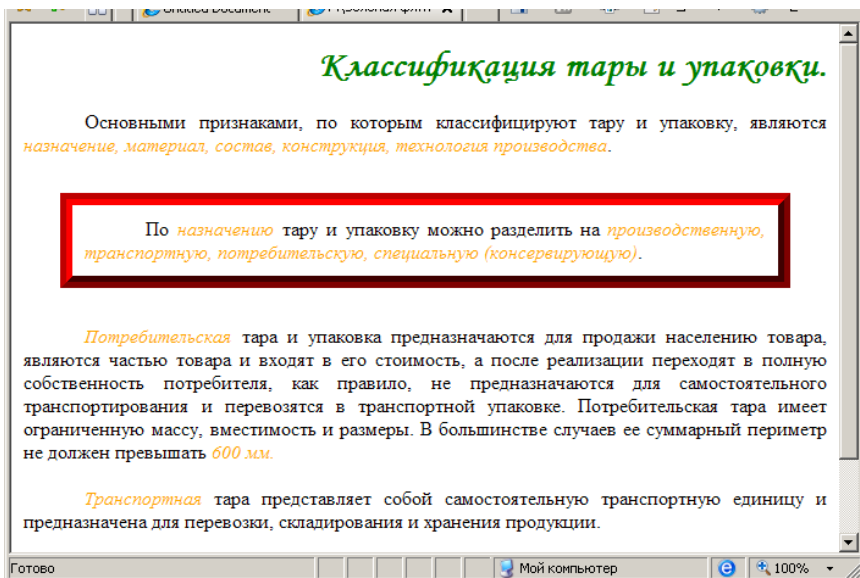
ВНИМАНИЕ!

Задание 6

2. Добавьте в документ следующий HTML-код:

```
<body>
<h1>Классификация тары и упаковки.</h1>
<p>Основными признаками, по которым классифицируют тару
и упаковку, являются <em>назначение, материал, состав, кон-
струкция, технология производства</em>.
<p >По <em>назначению</em> тару и упаковку можно разде-
лить на <em>производственную, транспортную, потребитель-
скую, специальную (консервирующую)</em>.</p>
<p><em>Потребительская</em> тара и упаковка предназнаचा-
ются для продажи населению товара, являются частью товара и
входят в его стоимость, а после реализации переходят в полную
собственность потребителя, как правило, не предназначаются
для самостоятельного транспортирования и перевозятся в транс-
портной упаковке. Потребительская тара имеет ограниченную
массу, вместимость и размеры. В большинстве случаев ее сум-
марный периметр не должен превышать <em>600 мм.</em></p>
<p><em>Транспортная</em> тара представляет собой самосто-
ятельную транспортную единицу и предназначена для перевоз-
ки, складирования и хранения продукции.</p>
<p><em>Производственная</em> тара предназначена для вы-
полнения внутрицеховых, внутризаводских и межзаводских
перевозок и накопления сырья, материалов, полуфабрикатов,
заготовок, готовых изделий и отходов.</p>
</body>
```

Определите стили таким образом, чтобы страница выглядела так, как показано ниже: (Новые HTML-теги добавлять нельзя!)



Лабораторная работа № 11

Технология CSS. Позиционирование и визуализация, использование статических и динамических фильтров

Цель работы: Знакомство со свойствами позиционирования и визуализации CSS. Использование статических и динамических фильтров.

Абсолютное позиционирование

Элемент, позиционированный абсолютно, не получает никакого пространства к документу. Это означает, что после позиционирования он не оставляет после себя пустое пространство.

Для абсолютного позиционирования элемента свойство `position` должно иметь значение `absolute`. Вы можете использовать значения `left`, `right`, `top` и `bottom` для размещения бокса.

Относительное позиционирование

Позиция элемента, размещаемого относительно, обчитывается относительно его оригинальной позиции в документе. Это означает, что вы смещаете элемент вправо, влево, вверх или вниз. Таким об-

разом, элемент всё ещё занимает в документе пространство после позиционирования.

Чтобы позиционировать элемент относительно, установите в свойстве position значение relative.

Перекрывание элементов

CSS оперирует в трёх измерениях - высота, ширина и глубина.

Для того, чтобы наложить элементы друг на друга необходимо присвоить каждому элементу номер (z-index). Элемент с большим номером перекрывает элемент с меньшим номером.

Задание

1. Создать новую страницу. На странице с помощью стилевого оформления, используя абсолютное позиционирование, создать объемный заголовок.

1.1. Создайте HTML-документ, используя в качестве содержимого следующий код:

```
<div
style="position:absolute;left:55;top:13">
  COMPUTER
</div>
<div
style="position:absolute;left:50;top:10;color:red">COMPUTER
</div>
```

1.2. Задайте значение **Z-индекса** для второго блока так, чтобы *верхний* текст оказался впереди соответствующего слоя. Просмотрите полученные результаты.

1.3. Установите значение свойства visibility текстового слоя так, чтобы текст стал невидимым.

1.4. Измените свойства слоя и текста так, чтобы текст позиционировался, начиная с позиции top:50 left:50, а серый слой позиционировался относительно текста на расстоянии 100 пикселей левее текста.

1.5. Создайте html-документ, используя позиционирование и **Z-индекс** для воспроизведения заголовка, приведенного на рисунке.



- 1.7. Измените позиционирование вертикального слоя в примере и расположите его в левой части окна браузера.
- 1.8. Измените позиционирование логотипа произвольным образом.
2. На этой же странице набрать текст объемом не менее 8 строк (по вашему усмотрению). Оформить текст буквицей. В качестве буквицы использовать изображение. Изображение создать в графическом редакторе Paint. Файл сохранить с расширением gif или jpg. Буквицу создать с помощью абсолютного позиционирования.
3. На странице создать таблицу 2x4. В ячейки таблицы вставить изображение `van_gog76_big.jpg`. Применить к изображениям фильтры:
 1. Alpha
 2. Blur
 3. Choma
 4. FlipH
 5. FlipV
 6. Gray
 7. Invert
 8. XRay
4. Создать 5 таблиц с одной ячейкой в которой находится текст: Пример фильтра.
 1. Применить к ячейкам таблицы следующие фильтры.
 2. DropShadow
 3. Shadow
 4. Glow
 5. Mask
 6. Wave

Лабораторная работа №12

Разработка динамических страниц с использованием JavaScript

Цель работы: ввести понятия сценария, теги интерактивности

Сценарии

Язык HTML служит для описания структуры текста, язык CSS – для описания того, как эта структура должна отображаться браузером. Эти два языка не предусматривают возможности обмена информацией с пользователем. Однако часто это бывает необходимо. Например, при создании учебного сайта есть потребность с веб-странице, реализующей тестирование.

Существует язык, называемый языком сценариев, который позволяет реализовывать простейшие действия над данными, которые вводит пользователь, а также над свойствами элементов веб-страницы. В нашем случае этот язык называется JScript.

Модель веб-страницы, в которой все элементы образуют иерархию объектов, где на вершине находится объект window, называется Document Object Model (DOM). У каждого объекта есть составляющие, к которым можно обращаться с помощью символа "точка" (прежде всего это *свойства*):

объект.свойство

Пример: обращение к свойству в команде присваивания
`window.status = 'Это строка состояния'`

Эта команда ("записать в строку состояния окна данный текст") могла бы работать в какой-либо определенной ситуации, например, при наведении указателя мыши на тот или иной элемент веб-страницы. Подобные ситуации называются *событиями*. *Сценарий* – это одна или несколько команд, которые запускаются при наступлении события.

Рассмотрим типичный случай, когда в качестве свойства рассматривается стиль некоторого элемента. Для обращения к объекту (элементу P) используется слово `this`. Событием будет наведение мыши: `onmouseover`

```
<p style="color:green"  
  onmouseover="this.style.color='red'">Интерактивный  
абзац</p>
```

Мы можем видеть, что цвет текста абзаца при наведении на текст указателя мыши изменяется на красный. Чтобы сработала обратная ситуация (цвет менялся снова на зеленый), нужно внести сценарий в описание события `onmouseout`:

```
<p style="color:green"
  onmouseover="this.style.color='red'"
  onmouseout="this.style.color='green'">Интерактивный
абзац</p>
```

Обработка данных пользователя

Рассмотрим два способа получения информации от пользователя:

- ввод текста;
- подтверждение щелчком.

Для приема текста от пользователя предназначен элемент `<input type="text" name="имя" />`

Для создания кнопки предназначен элемент

```
<input type="button" value="Надпись" onclick="сценарий" />
```

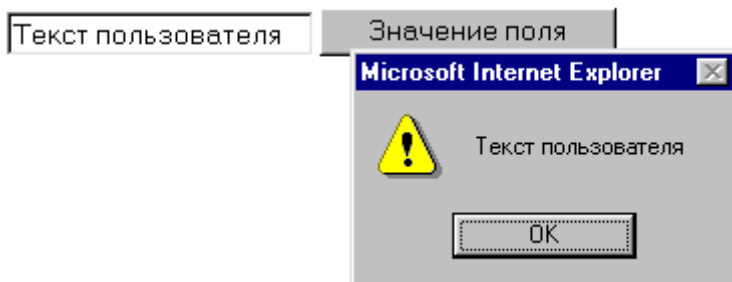
Вспользуемся командой `alert(текст)` для вывода на экран окна с сообщением:

```
<input type="text" name="поле" />
```

```
<input type="button" value="Значение поля" onclick="alert
(поле.value)" />
```

Текст п	Значение поля
---------	---------------

после щелчка по кнопке появляется текст, введенный пользователем



Применение сценариев

Точно так же, как вместо повторений одного и того же содержимого атрибута `style` пользуются однократной записью стилей во внешнем файле или в элементе `<style></style>`, вместо записи непосредственно в описании события можно сослаться на сценарий, который размещен в элементе `<script></script>`. Как и `<style></style>`, он размещается в элементе `<head></head>`.

Чаще всего сценарий появляется там в виде *функции*.

```
function имя_функции(ноль или более аргументов){  
    команды сценария  
}
```

Рассмотрим пример:

```
<script>
```

```
function приветствие (кого) {  
    alert ("Привет, " + кого + "!")  
}
```

```
</script>
```

```
...
```

```
<input type="text" name="кто" />
```

```
<input type="button" value="Здороваться!"  
onclick="приветствие (кто.value)" />
```

Мы начали обсуждение сценариев с упоминания тестов на веб-странице. Чтобы создать примитивный тест, нам понадобится еще один элемент HTML, принимающий ввод – радиопереклюатель – и конструкция языка JavaScript, выполняющая одно или другое действие в зависимости от условия – условный оператор. Если выбран правильный вариант ответа, сценарий выводит положительную реакцию, иначе – отрицательную.

```
if (условие) команда1; else команда2
```

Если *условие* истинно, то будет выполнена *команда1*, иначе будет выполнена *команда2*.

Пример:

В каком году заложен г. С.-Петербург?

```
<br/>
```

```
<input type="radio" name="вопрос1" />1812<br/>
```

```
<input type="radio" name="вопрос1" />1703<br/>
```

```
<input type="radio" name="вопрос1" />1624<br/>
```

```
<input type="button" value="Проверить!"
```



```
onclick="if (вопрос1[1].checked) alert("Верно!"); else  
alert('Нет верного ответа!)" />
```

В данном примере условие – это выбранность (*checked*) одного из радиопереклюателей. Всего имеется три переключателя, у которых одно имя (Вопрос1). Доступ к ним осуществляется с помощью *индекса* (номера в квадратных скобках, отсчитываемого с нуля – Вопрос1[0], Вопрос1[1] и Вопрос1[2]).

Если вопросов два и более, удобнее создать одну на всех функцию проверки. Она могла бы выглядеть следующим образом:

```
function проверка(верный) {  
    if (верный.checked) alert ('Верно!'); else alert ('Нет верно-  
го ответа!')
```

```
}
```

```
...
```

```
<input type="button" value="Проверить!"  
onclick="проверка(вопрос1[1])" />
```

В каком году заложен г. С.-Петербург?

1812

1703

1624

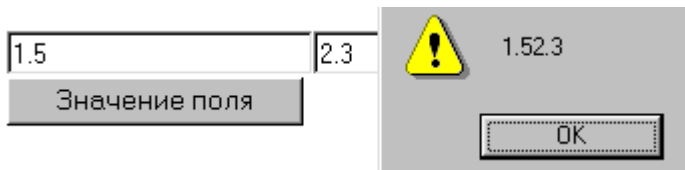
Проверить!



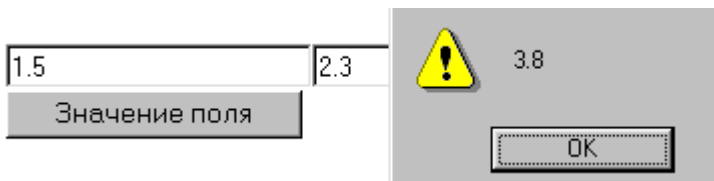
Задание

1. Создайте функцию, которую можно подключать как к событию *onmouseover*, так и к событию *onmouseout*, которая меняет цвет границы либо на цвет фона, либо на контрастный. Тогда при наведении мыши на элемент вокруг него будет вырисовываться граница, а при удалении – исчезать.

2. Пусть есть два текстовых поля: *число1* и *число2*. Команда `alert(число1.value+число2.value)` не будет давать сумму, если в полях введены два числа.



Чтобы содержимое поля трактовалось как число, нужно подвергнуть его преобразованию: `parseFloat(число1.value)`



Создайте калькулятор, выполняющий четыре арифметических действия (+, -, * и /). Реализуйте для функции, отвечающей за деление, проверку деления на ноль.

3. Создайте тест из пяти вопросов.

3*. Если в элементе Script поместить команду `var имя`, то в памяти будет создана переменная *имя*. Создайте функцию, в которой значение переменной увеличивается на 1 (`имя = имя + 1` или `имя++`) в случае правильного ответа, а затем выдается общее число правильных ответов. Соответственно, в этом тесте будет не пять кнопок после каждого вопроса, а одна после всего теста.

Добавления:

1. Если в блоке ветвления нужно использовать несколько команд, применяется форма:

```
if (условие) {  
    блок 1  
} else {  
    блок 2  
}
```

2. При необходимости использовать цикл применяют конструкцию for:

```
for (переменная=значение; условие; операция) {  
    тело цикла  
}
```

Например, подсчитаем сумму чисел 1..10:

```
s = 0;  
for (i=1; i<11; i++) {  
    s+= i;  
    alert (i);  
}
```

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Протоколы информационно-вычислительных сетей. Справочник. Под ред. И. А. Мизина и А. П. Кулешова, Радио и связь, Москва, 1990.
2. Локальные вычислительные сети. Справочник. Под ред. С.В.Назарова. М.: Финансы и статистика, 1994.
3. Бождай А.С., Финогеев А.Г. Сетевые технологии. Электронное учебное пособие, 2004.
4. Нанс Б. Компьютерные сети. - Пер с англ. - М.: Бином, 1995.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
Лабораторная работа № 1	
Браузер MS Internet Explorer	4
Лабораторная работа № 2	
Поиск информации в Internet	7
Лабораторная работа № 3	
Разработка простой WEB-страницы на языке HTML с использованием форматирования шрифтов и текста	10
Лабораторная работа № 4	
Разработка WEB-страницы на языке HTML с использованием списков	20
Лабораторная работа № 5	
Разработка WEB-страницы на языке HTML с использованием таблиц	28
Лабораторная работа № 6	
Разработка WEB-страницы на языке HTML с использованием графики, гиперссылок и изображений-карт	35
Лабораторная работа № 7	
Использование форм при разработке WEB-страницы на языке HTML	44
Лабораторная работа № 8	
Использование фреймов при разработке WEB-страницы на языке HTML	47
Лабораторная работа № 9	
Технология CSS. Разработка цвето-шрифтовой схемы и навигации сайта с помощью каскадных таблиц стилей	56
Лабораторная работа № 10	
Технология CSS. Проектирование оформления web-сайта с помощью каскадных таблиц стилей	62
Лабораторная работа № 11	
Технология CSS. Позиционирование и визуализация, использование статических и динамических фильтров	66
Лабораторная работа № 12	
Разработка динамических страниц с использованием JavaScript	69
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	74

Учебное издание

КОСТЮКЕВИЧ Елена Казимировна
ОСТАПЕНКО Инесса Витальевна

ИНФОРМАЦИОННЫЕ И СЕТЕВЫЕ ТЕХНОЛОГИИ
В УПАКОВОЧНОМ ПРОИЗВОДСТВЕ

Методическое пособие
для студентов специальности
1-36 20 02 «Упаковочное производство»

Технический редактор О.В. Песенько

Подписано в печать 27.05.2011.

Формат 60x84 1/16. Бумага офсетная.

Отпечатано на ризографе. Гарнитура. Таймс.

Усл. печ. л. 4,42. Уч.-изд. л. 3,45. Тираж 100. Заказ 493.

Издатель и полиграфическое исполнение:

Белорусский национальный технический университет.

ЛИ № 02330/0494349 от 16.03.2009.

Проспект Независимости, 65. 220013, Минск.