

Литература

1. Glossary of international Customs terms // Word Customs Organization [Electronic resource]. – 2022. – Mode of access: <http://www.wcoomd.org/en/topics/facilitation/instrument-and-tools/tools/glossary-of-international-customs-terms.aspx>. – Date of access: 11.02.2023.
2. Customs destruction procedure [Electronic resource]. Mode of access: <https://vinculum.ru/en/oformlenie/tamozhennye-protsedury/unichtozhenie>. Date of access: 12.02.2023.
3. Веремейчик, О.В. Английский язык для таможенников = English for Customs Officers : учебник / О.В. Веремейчик. – Минск : Вышэйшая школа, 2018. – 327 с.
4. Таможенная процедура уничтожения [Электронный ресурс]. – Режим доступа: <https://sbcargo.ru/poleznaya-informatsiya/tamozhennaya-protsedura-unichtozheniya/>. – Дата доступа: 07.10.2022.

PROGRAMACIÓN ORIENTADA A OBJETOS

Константинов Н.С.

Научный руководитель: ст. преподаватель Королько О.В.
Белорусский национальный технический университет

La programación orientada a objetos (POO) es un enfoque en el que un programa se considera un conjunto de objetos que interactúan entre sí. Cada uno tiene propiedades y comportamientos. La programación orientada a objetos hace que escribir código sea más rápido y legible.

Antes de la programación orientada a objetos, se utilizaba otro enfoque en el desarrollo, el procedimental. Un programa se representaba como un conjunto de procedimientos y funciones, subprogramas que ejecutaban un bloque específico de código con los datos de entrada necesarios. La programación procedimental es adecuada para programas ligeros sin una estructura compleja. Pero si los bloques de código son grandes y hay cientos de funciones, tendrás que editar cada uno de ellos y elaborar una nueva lógica. El resultado puede ser un montón de código desordenado y mal legible: "código espagueti" o "fideos".

La ideología de la programación orientada a objetos (POO) se desarrolló para relacionar el comportamiento de un objeto concreto con su clase. A las personas les resulta más fácil percibir el mundo que les rodea como objetos que se prestan a una determinada clasificación

Una clase es una "plantilla" para un objeto que describe sus propiedades. Varios objetos similares, como los perfiles de distintos usuarios, tendrán la misma

estructura y, por tanto, pertenecerán a la misma clase. Cada objeto es una instancia de una clase.

Cada objeto del sistema tiene propiedades y comportamiento como cualquier objeto real. Por ejemplo, consideremos el objeto "máquina". Tiene propiedades (color, peso, coste) y comportamiento (el coche puede conducir, tocar la bocina, consumir combustible).

Un objeto es un conjunto de variables y funciones como en la programación funcional tradicional. Las variables y las funciones son sus propiedades.

Los atributos son variables, características específicas de un objeto, como el color del campo o el nombre del usuario.

Los métodos son funciones que se describen dentro de un objeto o clase. Se refieren a un objeto concreto y permiten interactuar con él o con otras partes del código.

Este enfoque ayuda a construir sistemas complejos de forma más sencilla y natural, ya que todo el ámbito se descompone en objetos y cada objeto está vagamente relacionado con otros objetos. La conectividad débil es el resultado de la adhesión a tres principios: encapsulación, herencia y polimorfismo.

La encapsulación consiste en ocultar el comportamiento de un objeto en su interior. El objeto "conductor" no necesita saber lo que ocurre en el objeto "coche" para poder conducir. Es un principio clave de la programación orientada a objetos.

Herencia. Hay objetos "humano" y "conductor". Obviamente, tienen algo en común. La herencia nos permite aislar esta cosa común en un objeto (en este caso, el más común es una persona), y definir al conductor como una persona, pero con propiedades y/o comportamientos adicionales. Por ejemplo, un conductor tiene carné de conducir y una persona puede no tenerlo.

El polimorfismo es la redefinición del comportamiento. Podemos considerar de nuevo "persona" y "conductor", pero ahora añadimos "peatón". Una persona sabe moverse de alguna manera, pero cómo exactamente depende de si es conductor o peatón. Es decir, un peatón y un conductor tienen un comportamiento similar, pero implementado de forma diferente: uno se mueve con los pies, el otro con un coche.

La programación orientada a objetos permite simplificar objetos complejos a partir de otros más pequeños y sencillos, de modo que cientos de desarrolladores pueden trabajar en un programa, cada uno ocupado con su propio bloque. La mayoría de los lenguajes de programación modernos están orientados a objetos y, una vez que se le coge el truco, se pueden aprender varios lenguajes a la vez.

La programación orientada a objetos se utiliza para:

- estructurar la información y evitar confusiones;
- definir exactamente cómo interactúan unos elementos con otros
- mejorar la manejabilidad del programa
- escalar el código más rápidamente para diferentes tareas;

- comprender mejor lo que se escribe;
- Mantener el software existente de forma más eficiente;
- Implementar cambios sin tener que reescribir todo el código.

Ventajas de la programación orientada a objetos:

- Modularidad. El enfoque orientado a objetos hace que el código sea más estructurado y fácil de entender para el que no lo conoce. La encapsulación de objetos reduce los errores y acelera el desarrollo con más programadores, ya que cada uno puede trabajar de forma independiente.
- Flexibilidad. El código OOP es fácil de desarrollar, aumentar y cambiar. Esto lo proporciona la estructura modular independiente. Interactuar con objetos en lugar de con la lógica facilita la comprensión del código. No es necesario sumergirse en la forma en que está construido el software para modificarlo. Gracias al polimorfismo, el código puede adaptarse rápidamente a los requisitos de la tarea, sin tener que describir nuevos objetos y funciones.
- Ahorro de tiempo. Con la abstracción, el polimorfismo y la herencia no es necesario escribir el mismo código muchas veces. Esto acelera el desarrollo de nuevo software. Las interfaces y clases de la programación orientada a objetos pueden convertirse fácilmente en objetos tipo biblioteca que pueden reutilizarse en nuevos proyectos. La programación orientada a objetos también ahorra tiempo a la hora de mantener y modificar la aplicación.
- Seguridad. El programa es difícil de romper porque el código encapsulado no es accesible desde el exterior.

Desventajas de la programación orientada a objetos:

- Dificultad para empezar

Para utilizar la programación orientada a objetos, primero hay que aprender la teoría y los procedimientos.

Литература

1. Qué es la programación orientada a objetos [Электронный ресурс]. – Режим доступа: <https://intelequia.com/blog/post/qu%C3%A9-es-la-programaci%C3%B3n-orientada-a-objetos>. – Дата доступа: 02.04.2023.

2. Introducción a la programación orientada a objetos [Электронный ресурс]. – Режим доступа: https://www.cua.uam.mx/pdfs/revistas_electronicas/libros-electronicos/2016/2intro-oo-programacion_web.pdf. – Дата доступа: 02.04.2023.

3. Algoritmos y estructura de datos guías teórico-prácticas [Электронный ресурс]. – Режим доступа: <https://www.utnianos.com.ar/foro/tema-aporte->

FREE CUSTOMS ZONES AS A FEATURE OF CUSTOMS REGULATION IN THE FEZ

Королько Д.Д.

Научный руководитель: к. п. н., доцент Веремейчик О.В.

Белорусский национальный технический университет

In world trade practice, there are territorial formations within a state where preferential conditions for entrepreneurship are applied. The main goal for establishing such formations, which are called free economic zones, is to develop the economy throughout the country, individual regions or industries. These zones can take various organizational forms and be characterized by different purposes.

Free economic zones play a unique role in the legal framework of the Republic of Belarus as an essential institution of a market economy. Due to the special conditions for business, companies strive to become residents of the FEZ, that is, to place their factories and offices there.

A distinguishing characteristic of conducting business activities within the FEZ is the customs procedure of the free customs zone (Article 24 of the Law of the Republic of Belarus “On Free Economic Zones). This procedure allows local authorities to create free customs zones inside the FEZ, which are intended exclusively for its residents.

Free customs zones are specially designated areas within the borders of the FEZ that are equipped, designed, and managed in compliance with statutory regulations to facilitate the customs procedure of a free customs zone.

A free customs zone is a customs procedure under which goods are placed and used within the territory of the FEZ. Such goods are placed without payment of customs duties, taxes, as well as without the application of non-tariff measures in respect of foreign goods and without the application of prohibitions and restrictions in respect of goods of the EAEU.

The process of establishing and maintaining such zones is defined by the national legal framework. These areas can be created with or without facilities, and the national legislation provides for the necessary conditions for their functioning.

The territory of the free customs zone is subject to special requirement in respect of its organization. These requirements include the rules perimeter fencing, round-the-clock security (alarm system, video surveillance), marking on the ground, measures to maintain access and control.