



El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación

Programación Orientada a Objetos

Mision
TIC2022



El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación

Misión
TIC 2022

Importancia de la Programación Orientada a Objetos (POO)

La POO no sólo es muy útil para elaborar sistemas de cómputo complejos, sino que también es la base para el desarrollo de aplicaciones web. Los empleos para los programadores web están muy bien remunerados hoy en día, debido al predominio que tiene la Internet.

Entonces, podemos decir que el estudio de la POO es de gran interés para todos los que cursan carreras relacionadas con el mundo de las computadoras. Un buen programador debe tener la capacidad de abstraer problemas y solucionarlos mediante la computadora. Los ejemplos y ejercicios del presente libro fomentan el desarrollo de esta habilidad.

Los casos de estudio que se incluyen favorecen el desarrollo de las habilidades analíticas y de comprensión que el estudiante requiere para trabajar con el paradigma orientado a objetos.





El paradigma de la programación orientada a objetos

Un paradigma es una metodología que intenta unificar y simplificar la manera en que se resuelve un cierto grupo de problemas. En el contexto de la programación, un paradigma es un conjunto de principios y métodos que sirven para resolver los problemas a los que se enfrentan los desarrolladores de software al construir sistemas grandes y complejos. Existen varios paradigmas de programación; los importantes son los siguientes:

- El **estructurado**. Se basa en estructuras de control de flujo de programa (por ejemplo, si/entonces/si no, para y mientras). No se hacen "saltos" de un lugar a otro dentro de una rutina. De esta manera, los programas son más fáciles de entender.
- El **funcional**. Se programa con funciones y sus llamados correspondientes. El código con funciones pequeñas queda muy claro y promueve la reutilización.
- El **orientado a objetos**. Los programas trabajan con base en unidades llamadas objetos, los cuales siguen una serie de principios que veremos más adelante.



Ventajas del paradigma orientado a objetos

- El paradigma orientado a objetos es útil cuando el sistema se modela de forma casi análoga a la realidad, porque así se simplifica el diseño de alto nivel.
- Esta analogía permite que los programadores tengan más claro cuál es el papel de cada porción del programa y de los datos, lo que facilita la creación y el mantenimiento del sistema.
- Además, se promueve la reutilización, pues las similitudes entre objetos se programan sólo una vez en forma abstracta y el programador concentra su esfuerzo en las diferencias concretas.
- En la POO podemos, por ejemplo, diseñar el código para un botón virtual genérico que detecta el "click" del mouse y llama a una función. El código del botón se reutiliza cada vez que queremos crear un botón, pero con características particulares para cada caso.
- También se mejora el desarrollo de software a gran escala. Los equipos de programadores trabajan sobre objetos diferentes y, posteriormente, se integra el trabajo de todos haciendo uso de las interfaces (la cara hacia afuera) de los objetos.



Ventajas del paradigma orientado a objetos

A continuación se comparan las características principales de la programación estructurada y de la POO. Cabe señalar que no es mejor una que la otra, sino que su utilidad depende del tipo de sistema que se desarrollará. Si queremos utilizar la POO en todos los casos, complicaríamos el problema en lugar de facilitar su solución

Programación estructurada	Programación orientada a objetos
Problemas de naturaleza algorítmica (dada cierta entrada se produce una cierta salida)	Desarrollo de aplicaciones web y otros sistemas que se presten al modelado de objetos
Se basa en estructura de datos	Se basa en objetos, que tienen un estado y un comportamiento
La información fluye a través de las estructuras y de llamados a funciones	Hay interacciones entre los objetos



Objetos del Mundo Real de un Sistema Académico



Selección de objetos

Selección de objetos del mundo real para un Sistema Académico informático



En inglés, object-oriented programming (oop), define los programas en términos de clases de objetos, los objetos son entidades que combinan estado (datos y valores), comportamiento (métodos o procedimientos) e identidad (nombre del objeto que lo diferencia del resto). Esta técnica es utilizada para crear aplicaciones informáticas extensas en base a objetos del mundo real denominados entidades.

Para automatizar un sistema académico de una institución educativa, los objetos del mundo real identificados son: escuela, profesores, personal administrativo y de servicio, estudiantes, representantes, aulas, inscripción, matrícula, plan de estudio, cursos, distributivo o planificación académica, horarios, útiles escolares, etc.

Algunos de los objetos son importantes en el sistema real, pero en el sistema informático pueden carecer de relevancia debido a la delimitación u objetivo de la automatización; por ejemplo los útiles escolares y el personal de servicio podrían no ser considerados en el sistema informático.



Los objetos

En la POO, un objeto es una entidad virtual (o entidad de software), con datos y funciones que simulan las propiedades del objeto. Los objetos con los que se construyen los programas se ven como si fueran máquinas, las cuales están formadas por un conjunto de elementos autónomos.

Las propiedades individuales de estos elementos y las relaciones entre sí definen el funcionamiento general de la máquina. Desde el punto de vista del mundo real, un objeto tiene dos propiedades esenciales: un estado y un comportamiento:

- **El estado.** Son los datos asociados con el objeto, los cuales indican su situación interna en un momento dado, por ejemplo: velocidad, calificación, color, capacidad, encendido/apagado, saldo, etc.
- **El comportamiento.** Es la manera en la que el objeto responde a estímulos del exterior, por ejemplo, lo que sucede cuando se oprime el botón "inicio", lo que sucede cuando se hace un retiro de una cuenta bancaria o cuando se oprime el botón "reiniciar" en un contador.



Los objetos

Desde el punto de vista **computacional**:

- **Los atributos:** Son los datos que pertenecen al objeto y que representan el estado de éste, en función de los valores que tienen, por ejemplo:

```
double saldo;  
float calificacion;  
boolean on;
```

- **Los métodos:** Definen el comportamiento del objeto y son funciones que se pueden invocar desde otros objetos. Los métodos pueden modificar el estado del objeto cuando cambian el valor de alguno de los atributos, por ejemplo:

```
double obtenerSaldo();  
float calcularPromedio();  
boolean seOprimioOnOff();
```




Los objetos

Por ejemplo, a continuación, se muestra el objeto cuenta bancaria, cuyos datos contenidos son el nombre del cuentahabiente, sus apellidos, dirección, mail, saldo y el tipo de cuenta, en donde se especifica si es una cuenta de crédito, débito, ahorro, etc.

Además, el objeto cuenta bancaria también contiene algunas operaciones que se pueden ejecutar con la cuenta, en este ejemplo son consultar saldo, retiro, bonificación, actualizar datos y consultar datos.

Objeto: cuenta bancaria

Los datos del cuentahabiente	Operaciones que se pueden hacer con la cuenta
Nombre	Consultar datos
Apellidos	Retiro
Dirección	Bonificación
Correo electrónico	Actualizar datos
Saldo	Consultar datos
Tipo de cuenta	

A los datos del objeto se les denomina atributos y a las operaciones que se pueden ejecutar con el objeto se les llama métodos. Entonces, todos los objetos tienen un estado (conjunto de atributos) y un comportamiento (conjunto de métodos).



Los objetos lógicos

Objetos Lógicos. Son intangibles, creados por la necesidad del hombre para plasmar la información en forma textual o gráfica. Ejemplos:

- Elementos de interfaz gráficos de usuario: ventanas, botones, íconos, menús, etc.
- Estructura de datos: arreglos estáticos, estructuras, listas enlazadas, pilas, colas, árboles, etc.
- Tipos de datos definidos por el usuario: números complejos, fechas, puntos de un plano, etc.

Por ejemplo, los objetos lógicos de un sistema académico son: cartilla de notas, acta de notas, ficha de inscripción, hoja de matrícula, plan de estudios, cursos, distributivos, horarios, etc.



Procesos de abstracción e instanciación

Abstracción. El proceso de abstracción consiste en identificar o seleccionar entidades u objetos reales con similares características (atributos) y comportamiento; luego, se crean las clases que vienen a ser los modelos o plantillas donde se definen datos y métodos.

Instanciación. El proceso de instanciación implica crear objetos a partir de clases. Para instanciar objetos en la aplicación informática, es necesario contar con el modelo de clases. Los objetos informáticos son instancias (copia exacta) de clases con espacio en memoria para leer o escribir sus datos a través de sus métodos. Una vez que el objeto está creado, éste posee identidad, estado (datos con valores o atributos) y comportamiento (métodos que actúan sobre los propios datos).



Procesos de abstracción e instanciación

Partiendo de un conjunto de objetos del mundo real (estudiantes) con similares características y comportamiento, se crea una clase *Estudiante* con datos y métodos. Los datos vienen a ser nombres de variables que se han creado en base a los atributos o características de los objetos reales y los métodos son las operaciones que se pueden realizar con estos datos; luego, en base a la clase se puede crear o instanciar objetos con su identificación (E1), mientras que los datos con sus valores representan el estado, los métodos representan el comportamiento del objeto.





Consideraciones

Los tipos de datos abstractos llamados clases, encapsulan en una unidad la estructura de datos y los procedimientos que los manipularán. Las variables o instancias de la clase se denominan objetos, los cuales mediante un método envían mensajes a otros objetos para manipularlos de acuerdo a esos mensajes.

En relación con la programación tradicional, la programación orientada a objetos asocia nuevos fundamentos teóricos considerados como los elementos básicos, por ejemplo: objeto, clase, miembros de clase (datos y métodos), mensaje, herencia y otros términos asociados como son: abstracción, instanciación, reutilización, polimorfismo, encapsulamiento, ocultamiento y especificadores de acceso.

Las clases permiten fácilmente crear múltiples instancias u objetos sin tener que modificar o crear código. Cada objeto tiene su propio espacio en memoria para almacenar sus propios datos, los cuales se manipulan mediante los métodos.



El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación

Misión
TIC2022

Cuando se crea un programa, éste se organiza en paquetes de clases, donde cada clase consta de dos partes: la descripción del interfaz de la clase y la implementación de la misma:

- El interfaz describe qué hace la clase. Es la parte pública visible al exterior, esto es, los datos y los prototipos de métodos públicos.
- La implementación define cómo funciona la clase; es decir, el código para el funcionamiento de cada método y la parte no visible (privada o protegida).

Un programa en ejecución es una colección objetos que son instancias de clases. La ejecución comienza por un método principal de una clase estática que instancia objetos, los cuales interactúan con otros objetos mediante mensajes que ejecutan las acciones requeridas.



El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación

Mision
TIC2022

JAVA DEVELOPMENT KIT (JDK)

El Java Development Kit (JDK) es el conjunto de herramientas que nos permite trabajar en Java e incluye las herramientas y librerías para desarrollar aplicaciones y documentarlas.

El JDK contiene, entre otras cosas, el Java Runtime Environment (JRE) que consta de la JVM y de librerías.

Las principales herramientas que contiene el JDK son

- javac, compilador Java.
- java, Java Virtual Machine, para ejecutar aplicaciones Java.
- javadoc, para crear automáticamente documentación Java en HTML.
- jar, para compactar y descompactar archivos.