

PROGRAMACIÓN ORIENTADA A OBJETOS

Atributos o propiedades => que son datos:

Es un paradigma de programación que organiza el software entorno a objetos, los cuales combinan datos (atributos) y comportamientos (métodos) en una misma entidad. Estos objetos se basan en clases, que funcionan, como moldes o plantillas que definen sus características y acciones.

Diferencia entre Programación estructuras y P.O.O.

| Características | Programación Estructurada | Programación Orientada a Objetos |
|--------------------------------|---|--|
| Paradigmas | Se basa en procedimiento y funciones que operan sobre datos. | Se basa en objetos que combinan datos y comportamientos. |
| Organización del Código | El programa se divide en módulos o funciones. | El programa se divide en clases y objetos. |
| Enfoque | Orientado ha procesos: las secuencias de instrucciones es lo principal. | Orientado ha identidades: los objetos del mundo real son los principales. |
| Datos | Los datos son globales o compartidos, y pueden ser modificados por cualquier función. | Los datos (atributos) están encapsulados dentro de los objetos y solo son accesibles mediante sus métodos. |
| Reutilización | Baja se requiere repetir el código o copiar funciones. | Alta, gracias a la herencia y polimorfismo. |
| Mantenimiento | Difícil por que un cambio en los datos puede afectar muchas funciones. | Más sencillo, porque los cambios se localizan en clases u objetos específicos. |

En resumen:

- La programación estructurada organiza el software en funciones y pasos secuenciales siendo más adecuada para programas pequeños y sencillos.
- La P.O.O. organiza el software en clases y objetos, lo cual facilita la modularidad, reutilización y escalabilidad, siendo la mas usada en sistemas modernos.

Ventajas de las P.O.O.

1. **Modularidad:** El código se organiza en clases y objetos, lo que facilita la comprensión y mantenimiento.
2. **Realización del código:** Gracias a la herencia y al polimorfismo, se pueden crear nuevas clases a partir de otras sin repetir código.
3. **Encapsulamiento:** Protege los datos internos de los objetos, permitiendo acceso solo a través de métodos definidos.
4. **Abstracción:** Permite representar entidades del mundo real en modelos computacionales, simplificando la complejidad.
5. **Escalabilidad:** Es adecuada para el desarrollo de sistemas grandes y complejos, porque facilita dividir el trabajo en módulos independientes.
6. **Mantenimiento más sencillo:** Los cambios se realizan en clases específicas sin afectar todo el sistema.

7. **Flexibilidad y extensibilidad:** Permite adaptar y ampliar sistemas de manera más rápida y ordenada.
8. **Productividad:** Facilita el trabajo en equipo y a que varios programadores pueden desarrollar distintas clases u objetos de forma más rápida y ordenada.

Aplicaciones de la P.O.O.

La P.O.O. se aplica en una amplia variedad de áreas de desarrollo de software, ya que facilita la construcción de sistemas modulares, reutilizables y escalables. Entre sus principales aplicaciones destacan.

1. **Desarrollo de aplicaciones de escritorio:** Programas como editores de texto, hojas de cálculo o software de gestión utilizan programación orientada a objetos para organizar ventanas, menú, vótanos y acciones como objetos.
2. **Aplicaciones web y móviles:** Frame Word modernos aplican, principios de P.O.O. para manejar usuarios, productos, pedidos, mensajes, etc.
3. **Sistemas de bases de datos:** La P.O.O. se usa en el diseño de base de datos orientados a objetos y O.R.M. (object - Relational Mapping), que permiten mapear tablas a objetos de manera directa.
4. **Video juegos:** Cada personaje, enemigo escenario o arma puede representarse como un objeto con atributos (vida, posición, color) y método (moverse, atacar, desaparecer)
5. **Inteligencia artificial y simulación:** Los agentes inteligentes, redes neuronales o simulaciones físicas se modelan como objetos que interactúan entre sí.
6. **Sistemas distribuidos y en la nube:** Los servicios se modelan como objetos que se comunican en diferentes nodos, mantenimiento modoloridad y escolaridad.
7. **Interfaces graficas de usuario:** Librerías como Tkinter(Python),Java FX(java) y otros organizan cada elemento gráfico (botones, cuadros de texto, menus) como objetos reutilizables.

En conclusión: P.O.O. se aplica en cualquier área donde se requiera organizar y modelar entidades complejas del mundo real, siendo el paradigma más utilizado en el desarrollo de software moderno.

Clases:

En P.O.O. una clase es una plantilla o molde que permiten crear objetos.

- define atributos (propiedades o características).
- define métodos (acciones o comportamientos).

Los objetos son instancias de una clase, es decir, representaciones concretas creadas a partir de esa plantilla.

En otras palabras:

- La clase es el molde o plantillas
- El objeto es el producto creado a partir de ese molde.

