

Objeto: Es una unidad o entidad individual e identificable que representa un elemento con un rol bien definido en el dominio del problema

Estado: Son los valores que asumen los atributos en un momento determinado

Comportamiento: Es el conjunto de métodos u operaciones que el objeto está habilitado para hacer. Algunos pueden estar relacionados con sus atributos y otros con objetos relacionados

Identidad: Es la característica que hace que un objeto se distinga del resto. No existen dos objetos idénticos. Pueden ser iguales. Son iguales cuando tienen los mismos valores en los atributos y mismos métodos. No son idénticos porque cada uno ocupa un lugar distinto en el espacio-tiempo. Es decir, la identidad depende de la dirección en memoria que ocupan los objetos

Clase: Describe un conjunto de características y comportamientos comunes para un grupo de objetos

Ejemplos de tipos de clases

Roles desempeñados por Personas	Humanos que llevan a cabo alguna función. Ejemplo: Madre, profesor, político.
Lugares	Denotan espacios geográficos o elementos para contención de casas. Ejemplo: País, Barrio, Estantería.
Cosas	Objetos físicos, o grupos de objetos, que son tangibles. Ejemplo: Vehículos, libros, sensores de temperatura
Roles desempeñados Organizaciones	Colecciones formalmente organizadas de personas, recursos, instalaciones y posibilidades que tienen una misión definida, cuya existencia es, en gran medida, independiente de los individuos. Ejemplo: Banco, Ministerio, Administradora de Tarjeta de Crédito
Conceptos	Principios o ideas no tangibles, utilizados para organizar o llevar cuenta de actividades de negocios y/o comunicaciones. Ejemplo: Materia, Carrera, Tratamiento Médico.
Eventos o Transacciones	Cosas que suceden, que habitualmente vinculan a clases de alguna de las otras categorías, en una fecha y/u horas concretas, o como pasos dentro de una secuencia ordenada. Ejemplo: Vuelo, Inscripción, Venta, Reserva, Donación.

Tipos de métodos

- Modificador (set()): Permiten modificar atributos del objeto y con ello su estado del objeto.
- Selector (get()): Permiten acceder a los atributos del objeto sin modificarlos
- Constructor (new()): Crea un objeto y/o inicializa su estado
- Destructor (delete()): Destruye el objeto

Navegabilidad: Es la visibilidad que tienen unas clases sobre otras

Paradigma: Es una forma de estructurar el pensamiento o entender ciertos aspectos de la realidad

Reutilización: Enfocarse en la generación de componentes reutilizables y formas de resolver problemas

Trazabilidad: Es un vínculo de cada elemento del software con el dominio del problema

Modelo: Es un conjunto de clases

Cohesión: Mide que tan bien armada está la clase en términos de los atributos y métodos que elegimos colocar para definirla

- Es una medida de relación interna

Cohesión óptima: También llamada cohesión funcional o cohesión alta. Cada clase debe modelar un concepto y sus atributos y métodos debe estar relacionados con ese concepto. Se habla de una cohesión alta cuando los objetos están muy relacionados entre sí

Acoplamiento: Medida de la interconexión o dependencia entre clases.

- Sin acoplamiento no hay sistema
- Un acoplamiento demasiado elevado tiene el riesgo de que pequeños cambios en una clase afecte a muchas otras (efecto dominó)
- Es una medida de relación externa

Funcionamiento en POO: Es la capacidad de los objetos de comunicarse con otros objetos para pedirle que realice determinadas acciones. Esta capacidad de colaboración permite desarrollar modelos mas complejos

Visibilidad: Es la capacidad que tiene un objeto para pedirle a otro que realice una determinada acción

- La visibilidad se logra con los atributos de referencia (punteros)
- Si un objeto no puede pedirle información a otro, es porque no tiene visibilidad sobre ese objeto

Inconsistencia: Es la duplicación o redundancia de información. Para evitar esto alojamos cada pieza de información en un único lugar, y si un objeto necesita esa pieza de información contenida en otro objeto, a través de los métodos utiliza el puntero como vehículo para pedírselo. El otro objeto también debe tener un método para responder

Sistema orientado a objetos: Es un conjunto de objetos que colaboran entre sí para lograr un objetivo que solos no podrían. Si está bien diseñado hay una relación directa entre los objetos y los conceptos del mundo real que estos representan

Polimorfismo: Capacidad de objetos de distintas clases de responder de forma distinta a una misma llamada de función miembro.

Clase abstracta: Clases que permiten heredar atributos y métodos, pero no están pensadas para producir objetos

Clase concreta: Permiten producir objetos

Modelo de objetos: Modelo teórico que organiza un conjunto de elementos que ayudan a evaluar el grado de adecuación al Paradigma de Objetos

Conceptos Esenciales

Abstracción: Es un mecanismo que nos permite manejar la complejidad, teniendo en cuenta las características que nos resultan más relevantes y descartando aquellas que tienen menor interés. Es subjetiva.

- Las abstracciones se ven reflejadas en las clases
- La subjetividad puede ser contraproducente ya que nos lleva a una situación de percepciones diferentes sobre una misma cosa

Encapsulamiento: Es el ocultamiento del estado y la implementación. Es útil para proteger la información del objeto de modificaciones externas

- Modularidad: Permite dividir el todo en partes que tengan un tamaño que nos permita manejarlas y comprenderlas conceptualmente. Empaqueta abstracciones en unidades discretas. Debe respetar la cohesión y acoplamiento. Entre módulos debe haber la menor cantidad de relaciones posibles y dentro del módulo los elementos deben estar relacionados entre sí
- Jerarquía o Herencia (generalización): Comportamiento que permite a una clase heredar métodos y atributos de otra clase. El objetivo es el reúso del comportamiento

Conceptos Secundarios

- Concurrencia: Permite a dos objetos actuar al mismo tiempo

- Tipificación: Hace un control estricto del tipo de dato. Caracterización precisa de propiedades estructurales o de comportamiento que comparten ciertas entidades.
- Persistencia: Conserva el estado de un objeto en el espacio-tiempo más allá de la ejecución. Se busca almacenar los objetos en una memoria no volátil