

INTRODUCCIÓN A CLASES, OBJETOS Y MÉTODOS

UNIDAD 2: INTRODUCCIÓN A LA PROGRAMACIÓN ORIENTADA
A OBJETOS

SEMANA 3



Temario

- Conceptos en el paradigma orientado a objetos.
- Estructura de una Clase.
- Atributos y variables de instancia.
- Inicialización de objetos con constructores.
- Método recuperar(get) y establecer(set).
- Instanciación y rehusó.
- Tipos de referencia.



¿Qué es la abstracción?

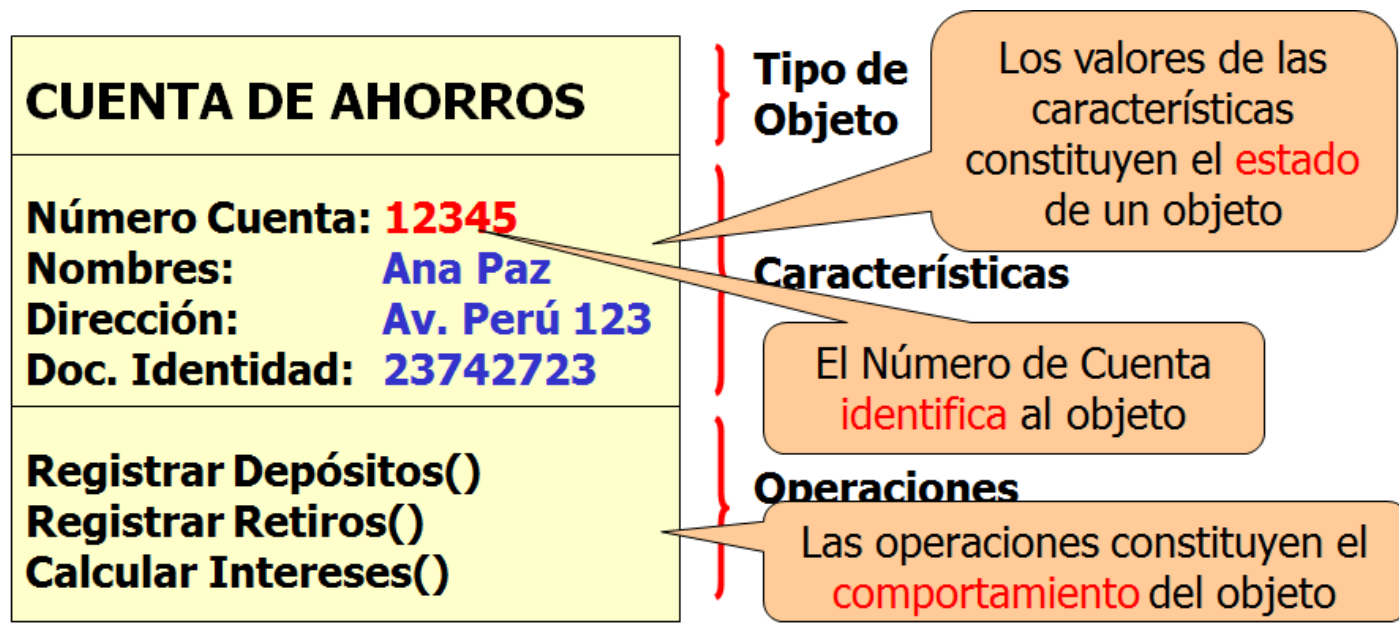
1. Conceptos en el paradigma orientado a objetos

- 1. Objeto:** Una instancia de una clase que representa una entidad o concepto con atributos y comportamientos.
- 2. Clase:** Un plano que define la estructura y comportamiento de los objetos.
- 3. Encapsulación:** Ocultar detalles internos de un objeto y proporcionar una interfaz pública.
- 4. Herencia:** Permitir que una clase herede propiedades y métodos de otra clase.
- 5. Polimorfismo:** Capacidad de objetos de diferentes clases para responder a la misma interfaz de manera diferente.
- 6. Abstracción:** Simplificar la representación de un objeto, centrándose en los aspectos relevantes.
- 7. Instanciación:** Crear un objeto a partir de una clase.
- 8. Mensaje:** Comunicación entre objetos a través de solicitudes de acción o información.
- 9. Clase abstracta:** Una clase que puede contener métodos concretos y abstractos, pero no puede ser instanciada.
- 10. Interfaz:** Define métodos que deben ser implementados por las clases que la utilizan.
- 11. Composición:** Un objeto puede estar compuesto por otros objetos como parte de su estructura.

2. Estructura de una Clase

• 2.1. Estructura de un objeto

- Definición Formal (Grady Booch)
 - Un objeto es una entidad que tiene un **estado**, un **comportamiento** y una **identidad**.



2. Estructura de una Clase (cont..)

- 2.1. Estructura de un objeto

- 2.1.1. concepto de objetos...(estado)

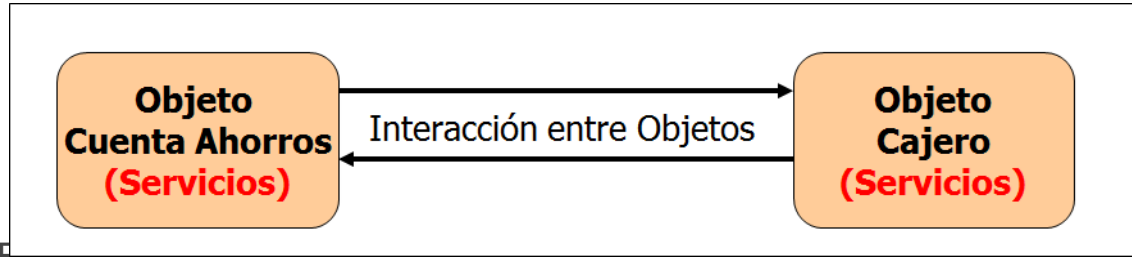
- Todos los objetos tienen características, propiedades o atributos. Por ejemplo, el objeto Cuenta de Ahorro tiene como atributos: **Número Cuenta, Nombres, Dirección, Documento de Identidad.**
 - Los **atributos** normalmente son estáticos, no cambian con el tiempo; mientras que los **valores** asociados a estos atributos son dinámicos, pueden cambiar con el tiempo.
 - El **estado** de un objeto está dado por los valores actuales de cada una de sus propiedades o atributos.

2. Estructura de una Clase (cont..)

- 2.1. Estructura de un objeto

- 2.1.1. concepto de objetos...(Comportamiento)

- Todos los objetos interactúan unos con otros. Esto significa que ningún objeto existe de forma aislada.
 - La interacción se efectúa en función a los **servicios** que ofrece cada objeto. Un servicio es alguna función (comportamiento) que puede efectuar un objeto.



- El comportamiento es como actúa y reacciona un objeto, en función a la interacción que efectúa sobre otros objetos o que efectúan otros objetos sobre él.

2. Estructura de una Clase (cont..)

- 2.1. Estructura de un objeto

- 2.1.1. concepto de objetos...(Identidad)

- Un objeto en el mundo real es único y se diferencia de los otros objetos incluso de la misma clase.
 - Es posible identificar entre todas las propiedades del objeto, una propiedad que lo distingue de todos los demás objetos, esta propiedad es la que identifica el objeto.

CUENTA DE AHORROS	
Número Cuenta:	12345
Nombres:	Ana Paz
Dirección:	Av. Perú 123
Doc. Identidad:	23742723

No puede haber dos cuentas de ahorros con el mismo Número. En consecuencia, el Número de Cuenta **identifica** unívocamente al objeto

- La **identidad** es aquella propiedad o atributo de un objeto que los distingue de todos los demás objetos.

2. Estructura de una Clase (cont..)

- 2.2 Encapsulamiento

Objeto = Características + Comportamiento

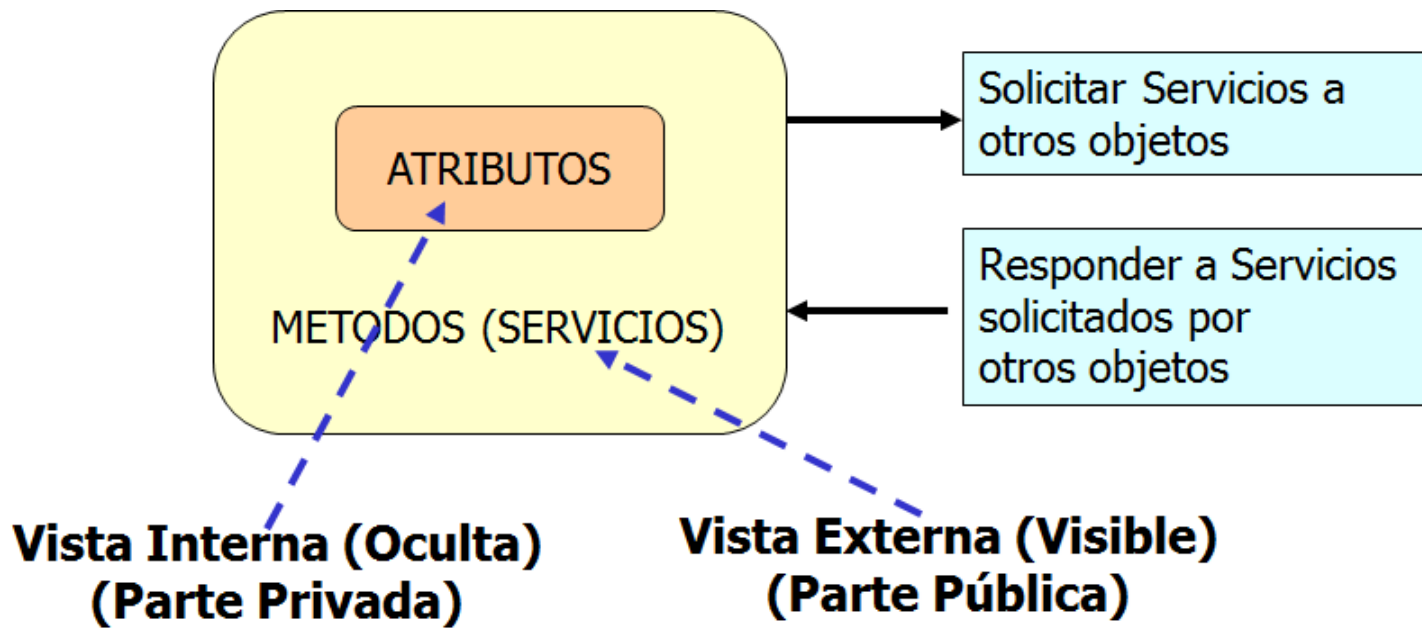
Objeto = Información + Proceso

Objeto = Atributos + Métodos

- Las características y el comportamiento de un objeto es modelado en un compartimiento (cápsula) único e indivisible.
- Por encapsulamiento debemos entender que un objeto debe ser considerado como “un todo”, como una unidad atómica indivisible, que engloba las características y el comportamiento del objeto.

2. Estructura de una Clase (cont..)

2.3 Representación de un objeto



2. Estructura de una Clase (cont...)

- 2.4. CLASES

- La **abstracción de clasificación** nos permite percibir que los objetos de la realidad están organizados como clases o tipos de objetos, caracterizados por propiedades comunes. Todo objeto pertenece a alguna clase de objetos.
- En la POO una clase es la implementación de un tipo de objetos. La POO identifica, implementa y utiliza tipos o clases de objetos del dominio del problema.
- **Una clase es como un molde que se utilizará para construir objetos.**

Clase = Implementación (*Atributos + Métodos*)
Comunes a todos los objetos de la clase

En la POO lo que realmente se implementa son las clases y no los objetos.
Los objetos se crean o construyen ("*tienen vida*") en tiempo de ejecución

2. Estructura de una Clase (cont...)

- **2.5. Ocultamiento de la información**

- Es el proceso de ocultar todos los detalles internos de un objeto, y poner a disposición de los otros objetos solamente la lista de servicios que éste ofrece.
- **Es ver un objeto con una vista interna y una vista externa.**
- La **vista interna** no es visible ni accesible para otros objetos, esconde la estructura de los atributos y protege el acceso directo a ellos desde fuera del objeto. También es parte de la vista interna la implementación de los métodos, es decir el “**como**” están hechos.
- La **vista externa** es lo que ven los otros objetos, es la parte visible y accesible del objeto, está constituido por la lista de métodos del objeto, con una explicación de “**que**” hacen o “**para que**” sirven.

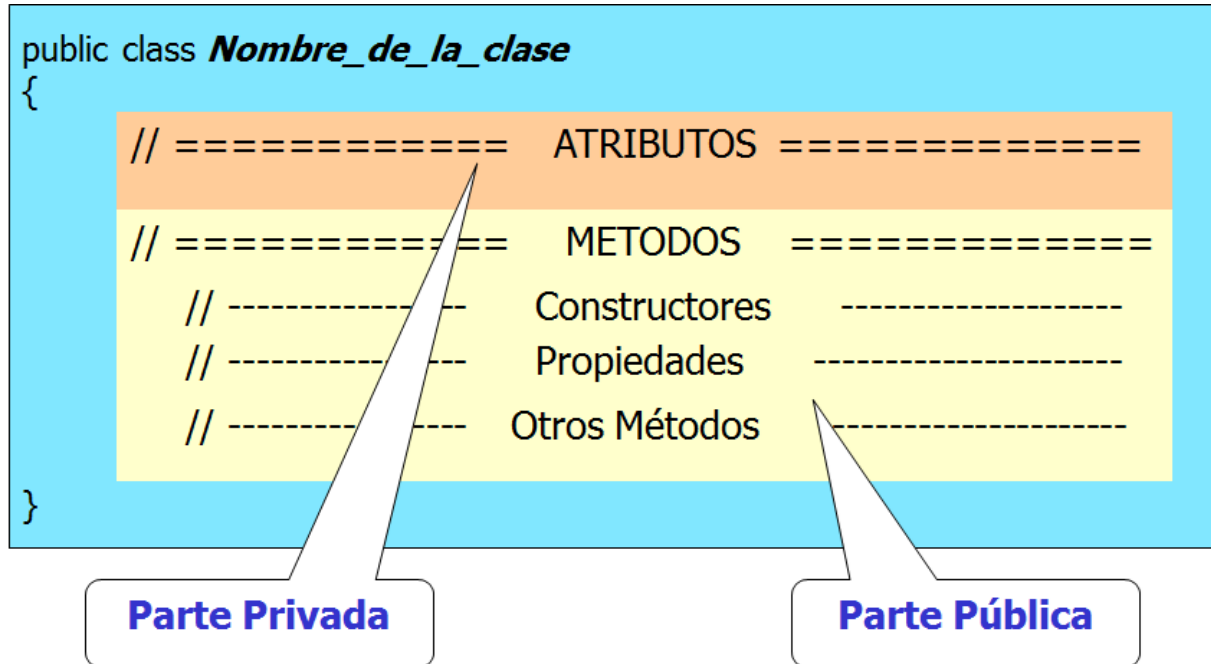
2. Estructura de una Clase (cont...)

- **2.6. Interfaces**

- La interfaz es la vista externa del objeto. Es el conjunto de servicios que ofrece el objeto. Los otros objetos o el resto del programa pueden utilizar el objeto sólo mediante los servicios definidos en la interfaz.
- Se puede modificar la implementación interna de un objeto A, mientras no se cambie la interfaz del objeto A los otros objetos que interactúan con éste no se verán afectados.

2. Estructura de una Clase (cont...)

- Esquema general de implementación de clases en Java, que se utilizará como patrón.



3. Atributos y variables de instancia

- **Atributos:**

- Son características específicas de un objeto
 - Color de un automóvil, la cantidad de gasolina en su tanque o su velocidad actual.
- Los atributos describen las propiedades o características de un objeto.

- **Variables de Instancia:**

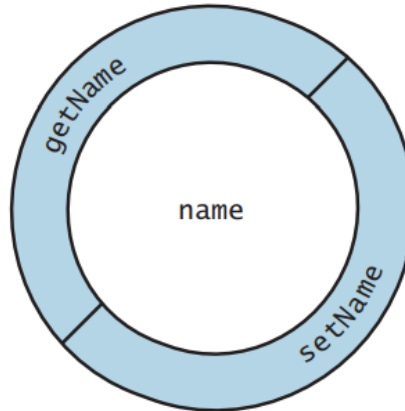
- Son variables que se utilizan para almacenar los atributos de un objeto.
- En otras palabras, son variables que pertenecen a una instancia particular de una clase (un objeto).
 - Por ejemplo, en un objeto de cuenta bancaria, la variable de instancia sería la que almacena el saldo de esa cuenta específica. Cada objeto tiene sus propias variables de instancia que almacenan sus atributos únicos.

4. Inicialización de objetos con constructores

- Cuando se crea un objeto de una clase, sus variables de instancia (como nombre en un objeto persona) a menudo se inicializan en el constructor de la clase.
- **Un constructor** es un método especial que se llama automáticamente cuando se crea un objeto de la clase. Puede aceptar parámetros para configurar las variables de instancia del objeto durante su creación.
 - Esto permite una inicialización personalizada de objetos, como establecer un por ejemplo un nombre al objeto persona cuando se crea el objeto.
- El uso de constructores es una práctica común para garantizar que los objetos se creen de manera coherente y con valores válidos.

5. Método recuperar(get) y establecer (set)

- **Método recuperar (get):** Un método público que se utiliza para obtener el valor de una variable privada en una clase. Proporciona acceso controlado a datos privados.
- **Método establecer (set):** Un método público que se utiliza para modificar el valor de una variable privada en una clase. Ofrece un mecanismo controlado para cambiar el estado de una instancia de clase.
- Ambos métodos son parte de la encapsulación, una práctica que protege y controla el acceso a los datos internos de un objeto.



6. Instanciación y rehusó

- La "**instanciación**" se refiere a la creación de objetos a partir de una clase, mientras que el "**reuso**" implica aprovechar y utilizar código existente en lugar de crearlo nuevamente, lo que mejora la eficiencia y la mantenibilidad del software.

7. Tipos de referencia

En programación, los tipos de datos se dividen en dos categorías principales: tipos primitivos y tipos de referencia.

- **Tipos Primitivos:**

- Representan valores simples como enteros, flotantes, caracteres y booleanos.
- Son eficientes y ocupan una cantidad fija de memoria.
- Se utilizan para datos simples y directos, como números y caracteres.

- **Tipos de Referencia:**

- Representan objetos y estructuras de datos más complejas.
- Almacenan referencias o direcciones de memoria en lugar de valores directos.
- Permiten acceder a atributos y métodos de objetos personalizados.
- Se utilizan para manejar datos complejos, como clases personalizadas, matrices y estructuras de datos.

Referencia

- Arturo Rozas Huacho, Algoritmos y Estructuras de Datos, 2002, Grupo Liebre, Cusco, Perú.
- Deitel, P. J., & Deitel, H. M. (2015). Java for programmers. Pearson education. Tenth Edition.