



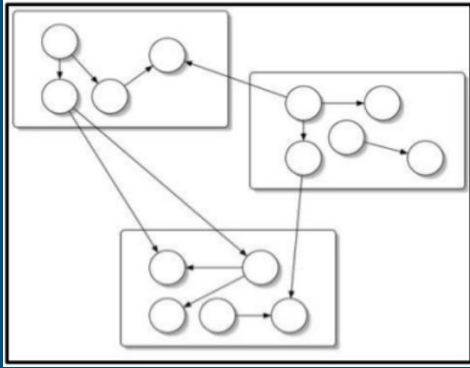
# Fundamentos de Java



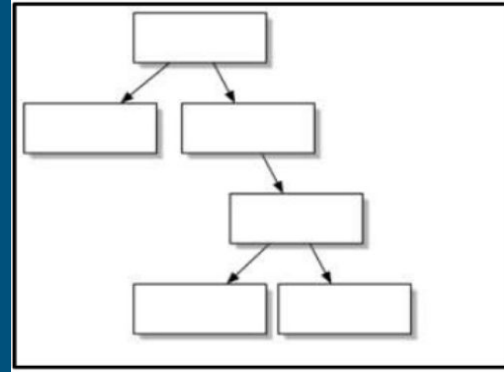
Módulo 1 - Clase #03  
Java Programmer SE 8



# Programación Orientada a Objetos



- Reusabilidad.
- Mantenibilidad.
- Modificabilidad.
- Fiabilidad.



- Facilidad de entendimiento.
- Menos esfuerzo en pruebas de algoritmo.
- Programas más sencillos y rápidos.
- Aumento de productividad.

# Características de la POO

---

1. Abstracción
2. Encapsulamiento
3. Herencia
4. Polimorfismo
5. Modularidad
6. Principio de Ocultamiento
7. Recolección de Basura

# Introducción a las Clases y Objetos

---

## Clases:

- Es una plantilla:
  - Nombre
  - Atributos
  - Métodos:
    - Arranque
    - Operaciones
  - Constructor
  - Destructor
- El nombre del archivo debe terminar con extensión .java
- Es una abstracción de algún hecho o ente del mundo real.

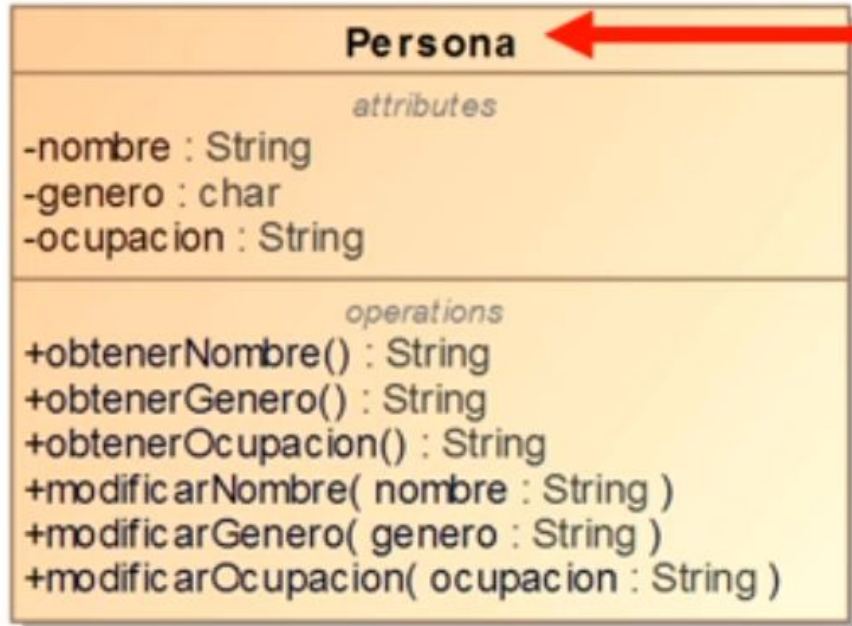
## Objetos:

- Es una instancia de una clase.
- Se compone de dos elementos:
  - Atributos
  - Métodos

# Diagrama de Tipos de Datos



# Diagrama General de una Clase



Nombre de la Clase

Atributos

Métodos

# Atributos

---

## Características:

- Son variables.
- Guardan datos.
- Deben ser declaradas, inicializadas y usadas tal como una variable o constante.

## Diferencia:

- Conceptual.
- Establece todas las características del objeto.
- Define los parámetros para que los desarrolladores identifiquen el objetivo de la clase.

# Métodos

---

## Características:

- Son funciones.
- Guardan subrutinas o algoritmos que cumplirán un objetivo específico.
- Deben ser declarados y se pueden usar desde cualquier ámbito siempre y cuando la clase se incluya en el mismo.

## Diferencias:

- Meramente conceptual.
- Los métodos establecen el comportamiento que un objeto debe tener.
- Los métodos deben definir los parámetros para que los desarrolladores puedan identificar con exactitud el objetivo o fin de esa clase java.



# Modificadores de Acceso

---

Modificador	Descripción
public	Los elementos pueden ser accedidos desde cualquier clase o instancia sin importar el paquete o procedencia de ésta.
private	Los elementos pueden ser accedidos únicamente por la misma clase.
protected	Permite acceso a los elementos desde la misma clase, clases del mismo paquete y clases que hereden de ella (incluso en diferentes paquetes).
default	Permite que tanto la propia clase como las clases del mismo paquete accedan a dichos componentes.

# Aplicando POO

---

## Trabajemos en conjunto:

- Nombre
- Atributos
- Métodos:
  - Arranque
  - Operaciones
- Constructor
- Destructor

# Ejercicio

---