

Asignatura

Programación

## UNIDAD 2

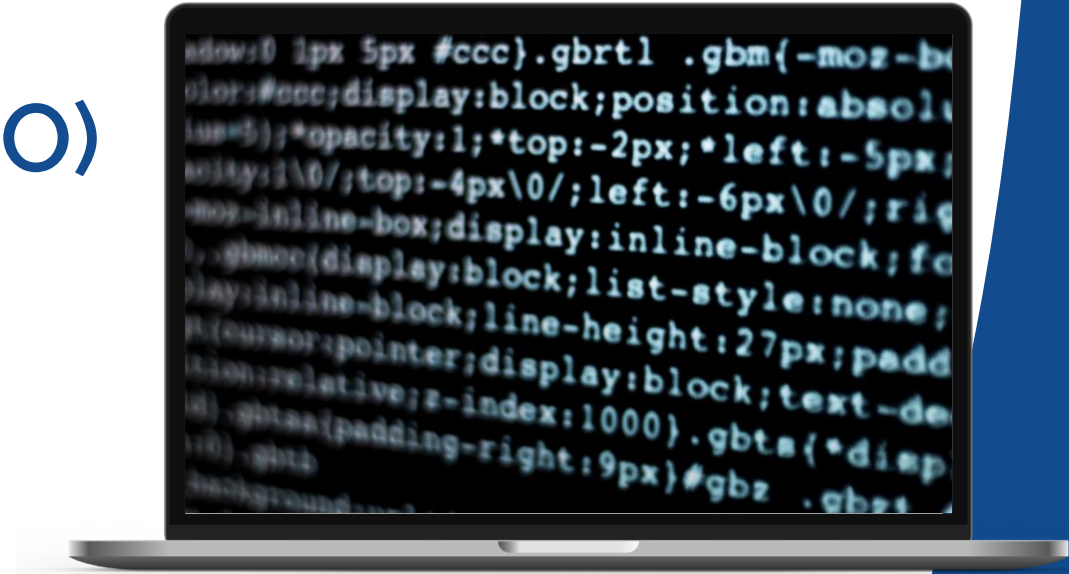
Programación orientada a objetos



UNIVERSAE  
Instituto Superior de FP

# La programación orientada a objetos (POO)

- Es un paradigma de la programación
- Se basa en conceptos de objetos en lugar de funciones y lógica
- Adecuado para programas grandes, complejos y de constante mantenimiento.
- El mas empleado
- Los lenguajes de programación mas conocidos de este paradigma son: Java y C#



## UML (Lenguaje unificado de modelado)

- No es un lenguaje de programación
- Permite definir diferentes tipos de diagramas
- Ideal para hacer diagramas de programación orientada a objetos

# Características. Clases y objetos

## Clase

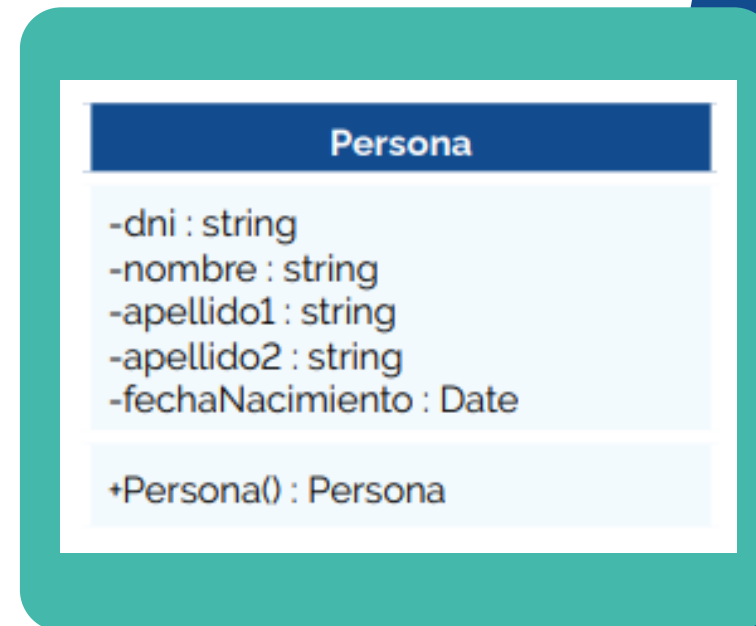
- Elemento principal de la POO
- Representación abstracta de un objeto real.
- Esta compuesta por:
  - Campos y propiedades
  - Constructores y destructores
  - Métodos y funciones

## UML

- Este elemento se representa con un rectángulo y tres partes diferenciadas
- En la cabecera se pone el nombre de la clase
- En el medio se indican los atributos
- En la parte inferior los métodos

## Objeto

- Es la instanciación de la clase
- Usa la estructura de la clase
- Es la parte que tiene valores



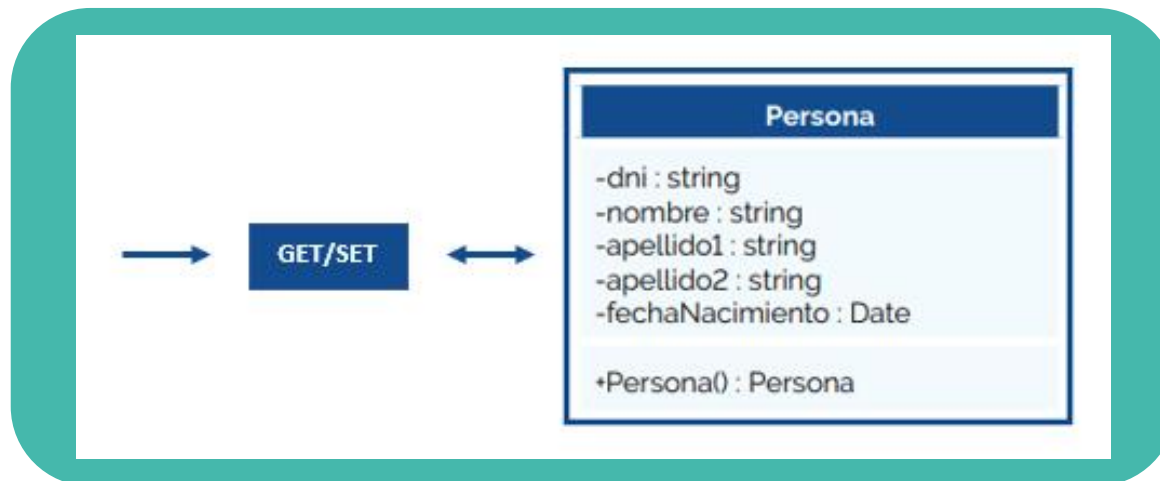
# Características. Encapsulación

## Definición

- Agrupa la información y función en una clase
- Se juega con la visibilidad
- Los datos son ocultos
- Solo se puede consultar y modificar datos mediante métodos

## Beneficios

- Oculta código para protegerlo
- Centraliza las llamadas a métodos mediante interfaces





# Características. Abstracción

Mecanismo que permite reducir la complejidad de los objetos minimizando la información a representar

## Ejemplo

Una persona al pulsar el botón de un ascensor. La persona no ve todo el mecanismo y sistema para hacer mover el ascensor.

# Características. Reutilización

La programación orientada a objetos tiene la característica de reusabilidad de código.

La reutilización de código permite:

- Reducir tamaño
- Reducir errores
- Facilita el mantenimiento





# Características. Visibilidad

- Determina como se puede los miembros de una clase
- Facilita la seguridad
- Gracias a la visibilidad se puede aplicar el encapsulamiento y la abstracción.



## Privada

- Se representa con el signo (-)
- Solo hay visibilidad desde la misma clase donde se declara.
- Desde fuera no se puede ver.



## Pública

- Se representa con el signo (+)
- Son visibles desde cualquier parte.



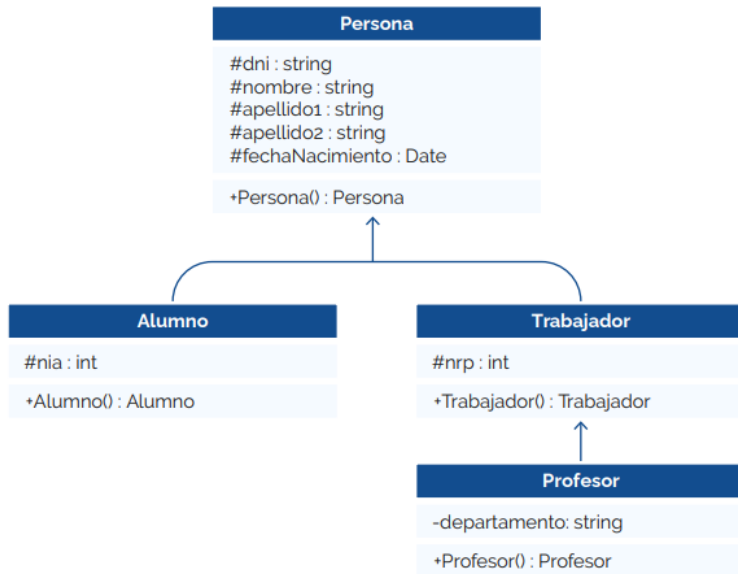
## Protegida

- Se representa con el signo (#)
- Tiene visibilidad en la misma clase y de cada clase que herede de ella.

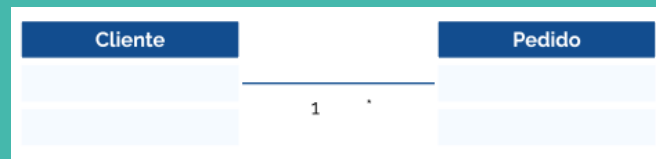


# Características. Relaciones

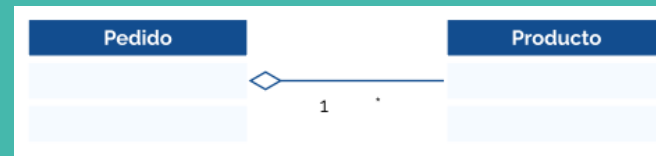
## Herencia



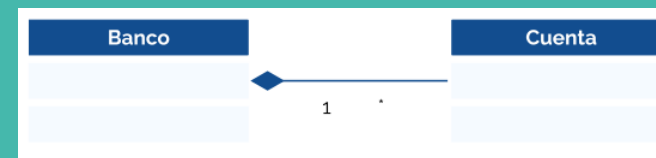
## Asociación



## Agregación



## Composición





# Características. Polimorfismo

## ¿Qué es?

- Los objetos pueden adoptar distintas formas
- Sucede en tiempo de ejecución
- Esta relacionado con la herencia

## Formas de polimorfismo

- **Asignación polimorfa** – Contiene en un objeto otras clases derivadas.
- **Ejecución polimorfa** – Invoca el mismo nombre del método sobre instancias de diferentes clase.





# Resumen

1. La programación orientada a objetos (POO)
2. Características
  - 3.1. Clase y objetos
  - 3.2. Encapsulación
  - 3.3. Abstracción
  - 3.4. Reutilización
  - 3.5. Visibilidad
  - 3.6. Relaciones (Herencia, Asociación, composición y agregación)
  - 3.7 Polimorfismo

The background is a solid blue color. Overlaid on this are several faint, light-blue geometric patterns. These include a grid of small squares that form larger, irregular shapes, and numerous small, light-blue arrows pointing in various directions. The overall effect is a sense of movement and digital connectivity.

# UNIVERSAE

— CHANGE YOUR WAY —