

# Orientación a objetos

Fernando Cerezal López

24 Noviembre 2005

# Definición de objeto

## al diseñar

En cuanto a diseño, un objeto es algo cuyo estado se puede describir por una serie atributos y que puede realizar funciones.

## al ejecutar

En cuanto a funcionamiento, un objeto es una instancia de una clase, esto es, una copia de esa clase con unos valores en sus atributos.

# Definición de clase

## Clase

Para hacer objetos necesitamos unos moldes, estos moldes son las clases. Una clase es el esqueleto de todos los objetos de ese tipo que hagamos.

Ejemplo:

| Persona                                                                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre: String<br>+apellidos: String<br>+DNI: int                                                                                                    |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

# Estructura de una clase - Atributos

## Atributos

Son las características que definen el objeto, pueden ser simplemente datos (tipos primitivos) u otros objetos.

# Tipos primitivos

| Tipo    | Rango de valores                        | Capacidad   |
|---------|-----------------------------------------|-------------|
| byte    | [-128, 127]                             | 1 byte      |
| short   | [-32768, 32767]                         | 2 bytes     |
| int     | Enteros [-2147483648, 2147483647]       | 4 bytes     |
| long    | Enteros [ $-2^{63}$ , $2^{63} - 1$ ]    | 8 bytes     |
| float   | Racionales [ $10^{-46}$ , $10^{38}$ ]   | 4 bytes     |
| double  | Racionales [ $10^{-324}$ , $10^{308}$ ] | 8 bytes     |
| char    | Carácter                                | 1 ó 2 bytes |
| boolean | 1=true ó 0=false                        | 1 bit       |

## Nota

Si a una variable aún no se le ha asignado un valor, su valor es «null».

Si un método no devuelve nada se utiliza el tipo «void» (vacío).

# Estructura de una clase - Métodos

## Método

Es la serie de acciones a desarrollar para llevar a cabo una función.

## Sintaxis

En general, a un método se le invoca con:  
`nombreObjeto.nombreMétodo(parámetro1, parámetro2, ...)`  
y termina con un  
`return valorQueDevuelve`  
o si no devuelve nada  
`return`

Ejemplos con la clase persona:

- Un método que te dé el valor del nombre de la persona.
- Un método que calcule la letra del DNI.

# Relación clase-objeto

| Persona                                                                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre: String<br>+apellidos: String<br>+DNI: int                                                                                                    |
| +getNombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

# Relación clase-objeto

| Persona                                                                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre: String<br>+apellidos: String<br>+DNI: int                                                                                                    |
| +getNombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

| objetoPersona1                                                                                                                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre = Carmen<br>+apellidos = Pérez García<br>+DNI = 123456789                                                                                     |
| +getNombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

# Relación clase-objeto

| Persona                                                                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre: String<br>+apellidos: String<br>+DNI: int                                                                                                    |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

| objetoPersona1                                                                                                                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre = Carmen<br>+apellidos = Pérez García<br>+DNI = 123456789                                                                                     |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

| objetoPersona2                                                                                                                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre = Pedro<br>+apellidos = López Rodriguez<br>+DNI = 987654321                                                                                   |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

# Relación clase-objeto

| Persona                                                                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre: String<br>+apellidos: String<br>+DNI: int                                                                                                    |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

| objetoPersona1                                                                                                                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre = Carmen<br>+apellidos = Pérez García<br>+DNI = 123456789                                                                                     |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

| objetoPersona2                                                                                                                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre = Pedro<br>+apellidos = López Rodriguez<br>+DNI = 987654321                                                                                   |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

| objetoPersona3                                                                                                                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| +nombre = Aurora<br>+apellidos = Jimenez Fernandez<br>+DNI = 567894321                                                                                |
| +getnombre(): String<br>+getApellidos(): String<br>+getDNI(): int<br>+setNombre(nombre:String)<br>+setApellidos(apellidos:String)<br>+setDNI(dni:int) |

# Un método especial: el constructor

## Constructor

Es un método con el que se crean nuevos objetos. Invocándole le decimos al sistema que reserve memoria y cree una nueva variable de esa clase.

## Propiedades

- Se llama igual que la clase que lo contiene.
- En él se le dan los primeros valores a los atributos (inicializan), por tanto, al llamarle hay que pasarle todos los valores necesarios.
- Generalmente, si no se escribe ninguno, se utiliza el constructor por omisión, que no recibe parámetros.

# Un método especial: el constructor

## Constructor

Es un método con el que se crean nuevos objetos. Invocándole le decimos al sistema que reserve memoria y cree una nueva variable de esa clase.

## Propiedades

- Se llama igual que la clase que lo contiene.
- En él se le dan los primeros valores a los atributos (inicializan), por tanto, al llamarle hay que pasarle todos los valores necesarios.
- Generalmente, si no se escribe ninguno, se utiliza el constructor por omisión, que no recibe parámetros.

# Un método especial: el constructor

## Constructor

Es un método con el que se crean nuevos objetos. Invocándole le decimos al sistema que reserve memoria y cree una nueva variable de esa clase.

## Propiedades

- Se llama igual que la clase que lo contiene.
- En él se le dan los primeros valores a los atributos (inicializan), por tanto, al llamarle hay que pasarle todos los valores necesarios.
- Generalmente, si no se escribe ninguno, se utiliza el constructor por omisión, que no recibe parámetros.

# Un método especial: el constructor

## Constructor

Es un método con el que se crean nuevos objetos. Invocándole le decimos al sistema que reserve memoria y cree una nueva variable de esa clase.

## Propiedades

- Se llama igual que la clase que lo contiene.
- En él se le dan los primeros valores a los atributos (inicializan), por tanto, al llamarle hay que pasarle todos los valores necesarios.
- Generalmente, si no se escribe ninguno, se utiliza el constructor por omisión, que no recibe parámetros.

# Definición de paquete

## Paquete

Un paquete es un conjunto de clases, que por razones de funcionalidad se agrupan juntas.

## Ejemplos

- Todas las clases para entrada y salida de datos.
- Todas las que se utilicen para funciones matemáticas.
- etc. . .

- clases:  
Se puede utilizar desde cualquier otra clase.
- métodos:  
Se le puede invocar desde cualquier clase.
- atributos:  
Desde cualquier clase se puede acceder directamente al atributo.

# Modificadores - private

- clases:  
Solo las clases de su mismo paquete la pueden utilizar.
- métodos:  
Solo se puede acceder a él desde dentro de su clase.
- atributos:  
Solo se le puede invocar desde dentro de su clase.

# Modificadores - static

## Definición

Lo definido como «static» no se instancia, esto es, no se hace copia de ello y lo comparten todos los objetos de la misma clase. Así se crea un atributo de clase. Si es una clase, entonces no se hacen objetos de ella y todos sus métodos y atributos son «static».

Ejemplo:

| caja                                                                           |
|--------------------------------------------------------------------------------|
| +static identificadorEnvio: String<br>+identificadorCaja: String<br>+peso: int |
| +static getIdentificadorEnvio()<br>+getIdentificadorCaja()<br>+getPeso()       |

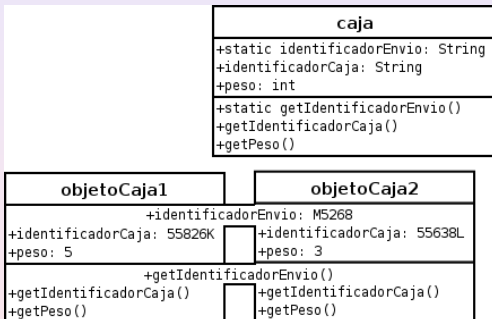
# Modificadores - static

Ejemplo:

|                                                                                                                                                                                                                           | <table><tr><th>caja</th></tr><tr><td>+static identificadorEnvio: String<br/>+identificadorCaja: String<br/>+peso: int</td></tr><tr><td>+static getIdentificadorEnvio()<br/>+getIdentificadorCaja()<br/>+getPeso()</td></tr></table> | caja                                                                 | +static identificadorEnvio: String<br>+identificadorCaja: String<br>+peso: int | +static getIdentificadorEnvio()<br>+getIdentificadorCaja()<br>+getPeso() |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|--------------------------------------------------------------------------------|--------------------------------------------------------------------------|
| caja                                                                                                                                                                                                                      |                                                                                                                                                                                                                                     |                                                                      |                                                                                |                                                                          |
| +static identificadorEnvio: String<br>+identificadorCaja: String<br>+peso: int                                                                                                                                            |                                                                                                                                                                                                                                     |                                                                      |                                                                                |                                                                          |
| +static getIdentificadorEnvio()<br>+getIdentificadorCaja()<br>+getPeso()                                                                                                                                                  |                                                                                                                                                                                                                                     |                                                                      |                                                                                |                                                                          |
| <table><tr><th>objetoCaja1</th></tr><tr><td>+identificadorEnvio: M5268<br/>+identificadorCaja: 55826K<br/>+peso: 5</td></tr><tr><td>+getIdentificadorEnvio()<br/>+getIdentificadorCaja()<br/>+getPeso()</td></tr></table> | objetoCaja1                                                                                                                                                                                                                         | +identificadorEnvio: M5268<br>+identificadorCaja: 55826K<br>+peso: 5 | +getIdentificadorEnvio()<br>+getIdentificadorCaja()<br>+getPeso()              |                                                                          |
| objetoCaja1                                                                                                                                                                                                               |                                                                                                                                                                                                                                     |                                                                      |                                                                                |                                                                          |
| +identificadorEnvio: M5268<br>+identificadorCaja: 55826K<br>+peso: 5                                                                                                                                                      |                                                                                                                                                                                                                                     |                                                                      |                                                                                |                                                                          |
| +getIdentificadorEnvio()<br>+getIdentificadorCaja()<br>+getPeso()                                                                                                                                                         |                                                                                                                                                                                                                                     |                                                                      |                                                                                |                                                                          |

# Modificadores - static

Ejemplo:



# Modificadores - static

Ejemplo:

| caja                                                                           |  |  |
|--------------------------------------------------------------------------------|--|--|
| +static identificadorEnvio: String<br>+identificadorCaja: String<br>+peso: int |  |  |
| +static getIdentificadorEnvio()<br>+getIdentificadorCaja()<br>+getPeso()       |  |  |

| objetoCaja1                            | objetoCaja2                                                          | objetoCaja3                            |
|----------------------------------------|----------------------------------------------------------------------|----------------------------------------|
| +identificadorCaja: 55826K<br>+peso: 5 | +identificadorEnvio: M5268<br>+identificadorCaja: 55638L<br>+peso: 3 | +identificadorCaja: 66852G<br>+peso: 4 |
| +getIdentificadorCaja()<br>+getPeso()  | +getIdentificadorEnvio()<br>+getIdentificadorCaja()<br>+getPeso()    | +getIdentificadorCaja()<br>+getPeso()  |

## Definición

Con la encapsulación se busca que cada objeto sea una caja negra de cara a los demás, y solo puedan saber datos de él a través de sus métodos públicos.

- Para conseguir esto se ponen todos los atributos como privados y se implementan unos métodos públicos que devuelvan y establezcan sus valores. De esta forma nos da igual como funcione una clase por dentro, solo sabremos que datos le pasamos y cuales nos devuelve.
- Por convención, los métodos para obtener los valores de los atributos son del tipo `getNombreAtributo()` y los métodos para establecer sus valores `setNombreAtributo(variable)`.

## Definición

La sobrecarga consiste en que puede haber varios métodos que se llamen igual, pero que se les pase distinto número de parámetros o de distinto tipo (si el lenguaje es tipado).

Ejemplos:

- metodoElQueSea(variable)
- metodoElQueSea(variable, variable2)

## Definición

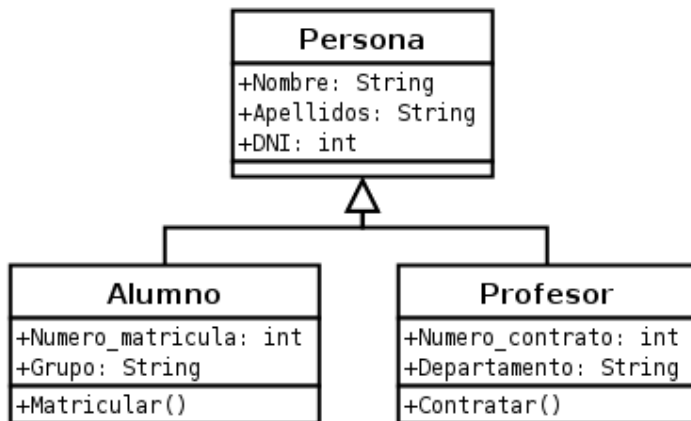
Una clase puede ser una extensión o particularización de otra, de forma que tenga todos sus atributos y métodos y añada algunos más. A esto se le denomina herencia.

Ejemplo:

| Alumno                 | Profesor              |
|------------------------|-----------------------|
| +Nombre: String        | +Nombre: String       |
| +Apellidos: String     | +Apellidos: String    |
| +DNI: int              | +DNI: int             |
| +Numero_matricula: int | +Numero_contrato: int |
| +Grupo: String         | +Departamento: String |
| +Matricular()          | +Contratar()          |

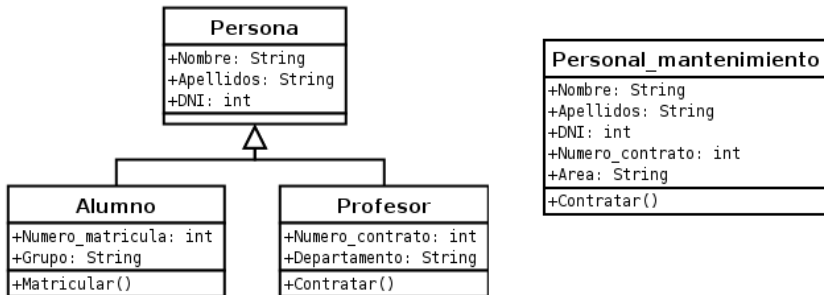
# Herencia

Ejemplo:

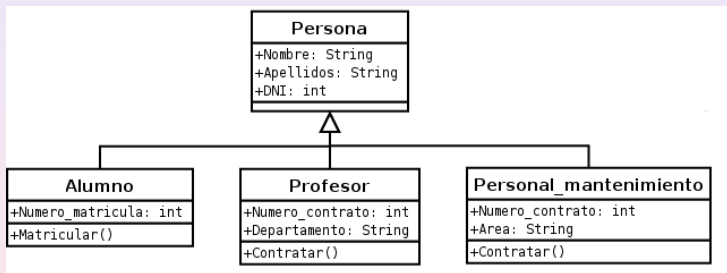


# Herencia

Ejemplo:

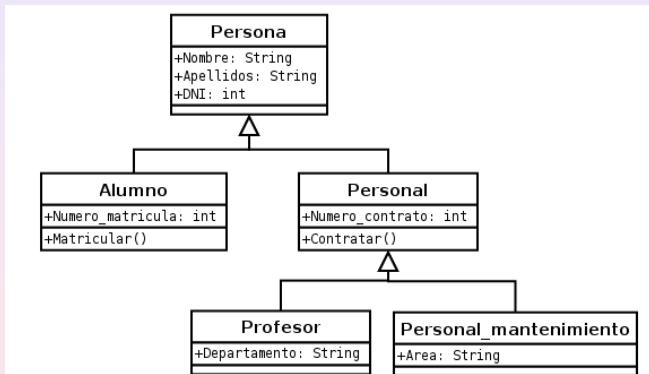


Ejemplo:



# Herencia

Ejemplo:



## Definición

Al igual que una clase puede tener varios métodos que se llamen igual, una clase que herede de otra puede sobrecargar los métodos de la clase que herede. El método que se ejecutará al llamarla se decidirá al ejecutarlo. A esto se le llama enlace dinámico.

## Definición

Se utiliza para transformar un tipo de dato en otro forzándolo.

¿?