

TEMA: DESARROLLO DE CLASES (PARTE I)

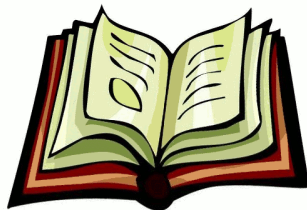
Taller de Programación.

Módulo: Programación Orientada a Objetos

Enunciado

- Generar una clase para representar libros. Un libro se caracteriza por: título, nombre del primer autor, nombre de la editorial, año de edición, ISBN, precio.
- El libro debe saber:
 - Devolver el valor de cada atributo.
 - Modificar el valor de cada atributo.
 - Calcular su precio final (incluyendo IVA).
 - Devolver su representación en formato String.

Ejemplo. “Java: A Beginner’s Guide por Herbert Schildt – 2014 - ISBN: 978-0071809252”



Libro
titulo, primerAutor, editorial, añoEdicion, ISBN, precio
<pre>String getTitulo() ... double getPrecio() void setTitulo(String unTitulo) ... void setPrecio(double unPrecio) double getPrecioFinal() String toString()</pre>

Definición de clases

- Sintaxis

```
public class NombreDeClase {  
    /* Declaración del estado del objeto*/  
    /* Declaración de constructor(es) */  
    /* Declaración de métodos que implementan acciones */  
}
```

Declaración del estado

- Sintaxis:

- Datos de tipos primitivos. `TipoPrimitivo nombreDato;`
- Referencias a otros objetos. `NombreDeClase nombreDato;`
- Anteponer a la declaración la palabra `private` para lograr encapsulamiento (ocultamiento de la información).

Las v.i.s **privadas** pueden ser accedidas sólo dentro de la clase que las declara

¿Y si fueran **públicas**?

- En la declaración del dato se puede dar un valor inicial (inicialización explícita).

Ejemplo

```
double precio;
```

```
String titulo;
```

```
private double precio;
```

```
private String titulo;
```

```
private double precio = 10.5;
```

```
private String titulo = "Java";
```

Declaración del estado. Ejemplo

```
public class Libro {  
  
    /* Declaración del estado */  
  
    private String titulo;  
    private String primerAutor;  
    private String editorial;  
    private int añoEdicion;  
    private String ISBN;  
    private double precio;  
  
    ...  
}
```

Al instanciar un libro, las v.i.s que no se inicializaron explícitamente tomarán un valor por defecto
(numéricos => 0; boolean => false;
char => "; objetos => null)

¿Qué debo hacer si quiero que mis libros tengan inicialmente año de edición 2015 y precio 100?

Declaración del comportamiento

- Sintaxis

```
public TipoRetorno nombreMetodo ( lista, parámetros, formales ) {  
    /* Declaración de variables locales al método */  
    /* Cuerpo del método */  
}
```

- **public** indica que el método forma parte de la interfaz. [¿Diferencia con private?](#)
- **TipoRetorno** tipo de dato primitivo | nombre de clase | void (*no se retornará dato*).
- **nombreMetodo** verbo seguido de palabras (convención de nombres).
- **Parámetros** datos de tipos primitivos u objetos, separados por coma, paso por valor
TipoPrimitivo nombreParametro, NombreClase nombreParametro
- **Declaración de variables** declaración igual que en *main*. ¿Ámbito de uso? ¿Tiempo de vida en memoria?
- **Cuerpo** código que puede utilizar/modificar el estado; devolver resultado con return

Definición de clases. Ejemplo.

```
public class Libro {  
    private String titulo;  
    private String primerAutor;  
    private String editorial;  
    private int añoEdicion;  
    private String ISBN;  
    private double precio;  
  
    public String getTitulo(){  
        return titulo;  
    }  
    public void setTitulo(String unTitulo){  
        titulo = unTitulo;  
    }  
    public double getPrecio(){  
        return precio;  
    }  
    ...  
}
```

Estado
(características)



Métodos
(acciones)



Generar una clase para representar libros. Un libro se caracteriza por: título, nombre del primer autor, nombre de la editorial, año de edición, ISBN, precio. El libro debe saber:

- Devolver el valor de cada atributo.
- Modificar el valor de cada atributo.
- Calcular su precio final (incluyendo IVA).
- Devolver su representación en formato String.

Libro.java

```
public void setPrecio(double unPrecio){  
    precio = unPrecio;  
}
```

```
public double getPrecioFinal(){  
    return precio + (precio*0.21);  
}
```

```
public String toString(){  
    String aux = titulo + " por " +  
        primerAutor + " - " +  
        añoEdicion + " - ISBN: " +  
        ISBN;  
  
    return aux;  
}
```

“aux” es variable local al método

Instanciación (creación de objeto)

- Declarar variable para mantener la referencia:

```
NombreDeClase miVariable;
```

- Enviar a la clase el mensaje de creación y guardar referencia:

```
miVariable= new NombreDeClase(valores, para, inicialización);
```

- Se puede unir los dos pasos anteriores:

```
NombreDeClase miVariable= new NombreDeClase(...);
```

Ejemplo (en main)

```
Libro miLibro;
```

```
miLibro = new Libro();
```

```
Libro miLibro = new Libro();
```

Secuencia de pasos en la instanciación (creación de objeto):

Reserva de Memoria. Las variables de instancia se inicializan a valores por defecto o explícitos.



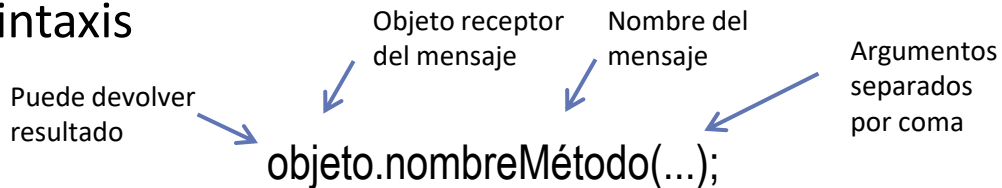
Ejecución del Constructor. Inicia las variables de instancia con los valores que enviamos en el mensaje de creación.



Asignación de la referencia a la variable.

Envío de mensaje al objeto

- Sintaxis



```
public class Demo01Libro {  
    public static void main(String[] args) {  
        Libro miLibro = new Libro();  
        miLibro.setTitulo("Java: A Beginner's Guide");  
        miLibro.setEditorial("Mcgraw-Hill");  
        miLibro.setAñoEdicion(2014);  
        miLibro.setPrimerAutor("Herbert Schildt");  
        miLibro.setISBN("978-0071809252");  
        miLibro.setPrecio(21.72);  
        System.out.println(miLibro.toString());  
        System.out.println(miLibro.getPrecioFinal());  
    }  
}
```

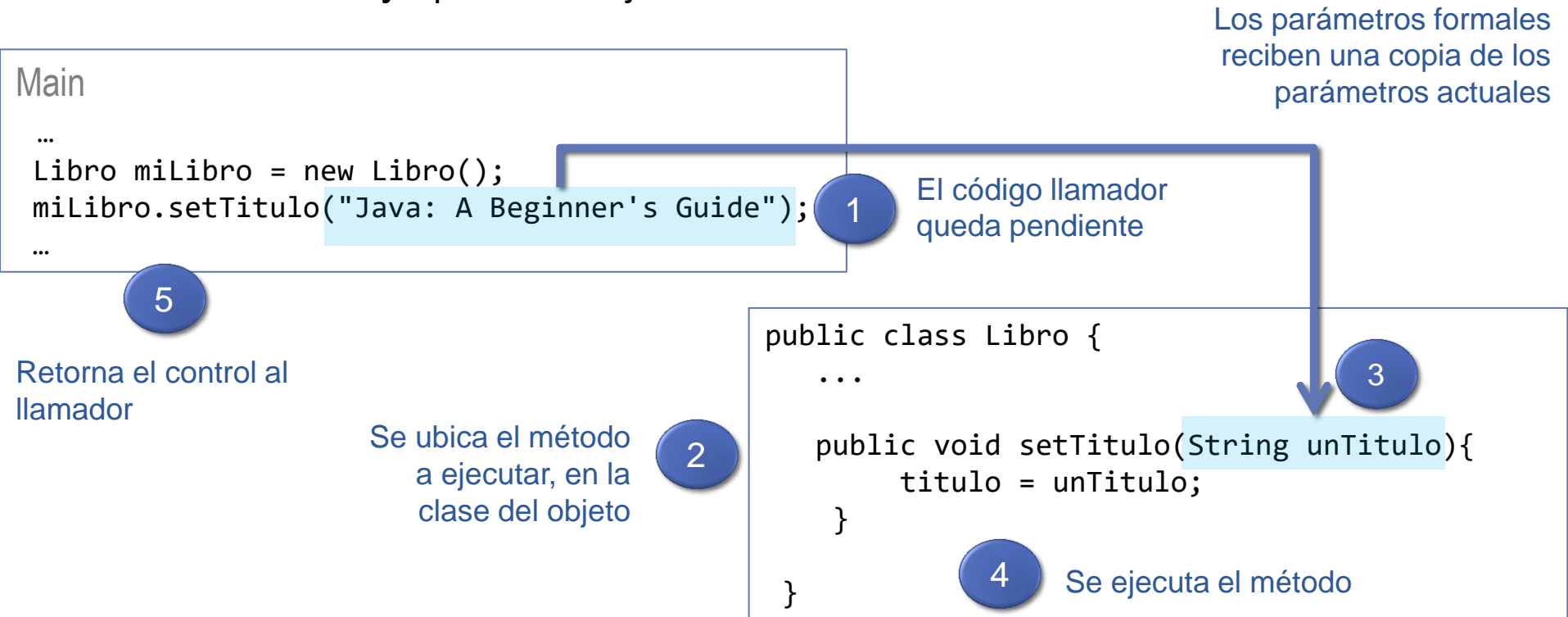
Demo01Libro.java

Output - Pruebas (run)

```
run:  
Java: A Beginner's Guide por Herbert Schildt - 2014 - ISBN: 978-0071809252  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Envío de mensaje al objeto

- **Envío de Mensaje:** provoca la ejecución del método



Envío de mensaje al objeto. Funcionamiento parámetro por valor.

- **Caso A)** Parámetro dato primitivo
 - **Parámetro formal** recibe una **copia** del parámetro actual (**dato primitivo**).
 - **Si se modifica el parámetro formal, no se altera el parámetro actual.**

Main

```
...  
Libro l1 = new Libro();  
l1.setTitulo("Java");  
l1.setPrecio(100);  
...  
double porcAumento=0.20;  
l1.subirPrecio(porcAumento);  
System.out.println(porcAumento);
```

¿Qué imprime?



```
public class Libro {  
    private String titulo;  
    ...  
    private double precio;  
    ...  
    public void subirPrecio(double porc){  
        porc = precio * porc;  
        precio = precio + porc;  
    }  
    /* Aumenta el precio en un  
    porcentaje recibido */  
}
```

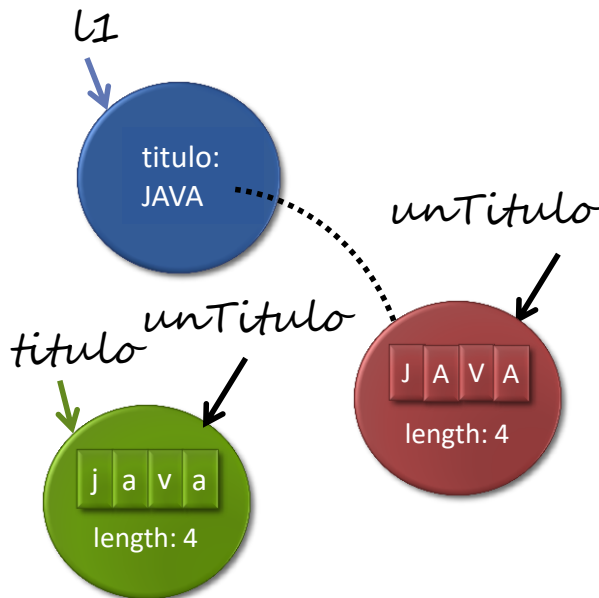
Envío de mensaje al objeto. Funcionamiento parámetro por valor.

- **Caso B.1) Parámetro objeto**
 - **Parámetro formal** recibe una **copia** del parámetro actual (*referencia al objeto*).
 - Si se modifica el parámetro formal, *no se altera el parámetro actual*.

Main

```
...  
Libro l1 = new Libro();  
String titulo= "java";  
l1.setTitulo(titulo);  
System.out.println(titulo);
```

¿Qué imprime?



```
public class Libro {  
    private String titulo;  
    ...  
    private double precio;  
    ...  
  
    public void setTitulo(String unTitulo){  
        unTitulo=unTitulo.toUpperCase();  
        titulo = unTitulo;  
    }  
  
    /* Establece el título al  
    recibido pero en MAYUSCULAS */  
}
```

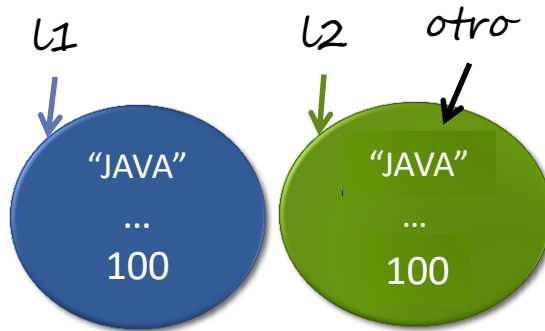
Envío de mensaje al objeto. Funcionamiento parámetro por valor.

- **Caso B.2) Parámetro objeto**

- **Parámetro formal** recibe una **copia** del parámetro actual (*referencia al objeto*).
- **Si se modifica el estado del parámetro formal, el cambio es visible desde el parámetro actual.**

Main

```
...
Libro l1 = new Libro();
l1.setTitulo("JAVA");
l1.setPrecio(100);
...
Libro l2 = new Libro();
l2.setTitulo("PASCAL");
l2.setPrecio(50);
...
l1.copiateEn(l2);
System.out.println(l2.getTitulo());
System.out.println(l2.getPrecio());
```



¿Qué imprime?

```
public class Libro {
    private String titulo;
    ...
    private double precio;
    ...

    public void copiateEn(Libro otro){
        otro.setTitulo(titulo);
        ...
        otro.setPrecio(precio);
    }

    /*El libro receptor del
    mensaje copia todos sus datos
    en el libro otro (parámetro)*/
}
```

IDE NetBeans



Crear Clase Java

- Click derecho sobre la carpeta contenedora
Ej: "tema 3"
- New → Java Class
- *Class Name*: Poner un nombre
- Finish

Aparecerá un archivo .java con el esqueleto de la clase