



# *Programación Orientada a Objetos*

*Clase 4 - Smalltalk. Diseño de clases compuestas.*

*PARADIGMAS DE PROGRAMACION*

*UTN - La Plata*



## Definición de la clase Biblioteca y Aplicación

- 1) Especificación de la clase Biblioteca.
- 2) Aplicación.
- 3) Implementación de la clase Biblioteca en Smalltalk.
- 4) Actividad 4.

# Especificación de la clase Biblioteca

## 1) Especificación de la clase Biblioteca

- **Clase Biblioteca**

**Subclase de:** Object

**Variables de instancia:** nombre libros

**Métodos de clase:**

**>>crearBiblioNom:**unNom

“Retorna una instancia de Biblioteca inicializada”

**Métodos de instancia:**

**>>iniBiblioNom:**unNom

“Inicializa una instancia de Biblioteca”

## Especificación de la clase Biblioteca

>>**verNom**

“Retorna el nombre de la Biblioteca”

>>**modiNom:unNom**

“Modifica el nombre de la Biblioteca”

>>**agregarLibro:unLibro**

“Agrega un libro a la Biblioteca”

>>**eliminarLibro:unLibro**

“Elimina un libro de la Biblioteca”

>>**existeLibro:unLibro**

“Retorna V si el libro está en la Biblioteca, F en caso contrario(cc)”

>> **esVacía**

“Retorna V si la Biblioteca no tiene libros, F en cc”

## Especificación de la clase Biblioteca

>>**buscarLibro**lsbn:unlsbn

“Retorna el libro de lsbn unlsbn si existe, nil en cc”

>> **todosLosLibros**

“Retorna los libros de la Biblioteca”

>> **cantidadLibros**

“Retorna la cantidad total de libros de la Biblioteca”

>>**recuperarLibro**: pos

“retorne el libro de la posición pos de la biblioteca”

**2) Aplicación:** Informar los títulos de los libros que no estén prestados

## Aplicación

|b n l is t a e col1 col2 |

n:= Prompter prompt:'Ingrese un nombre' default:''.

b:= Biblioteca crearBiblioNom:n.

1 to:3 do:[i] is:= Prompter prompt:'Ingrese un isbn'.

t:= Prompter prompt:'Ingrese un título'.

a:= Prompter prompt:'Ingrese un autor'.

e:= Prompter prompt:'Ingrese una editorial'.

l:= Libro crearLibroIsbn:i tit:t aut:a edit:e.

b agregarLibro:l].

col2:=OrderedCollection new.

1 to: (b cantidadLibros) do: [: i | lib:= b recuperarLibro:i. (lib verEstado=false) ifTrue:[ col2 add: lib] ].

col1:=b todosLosLibros.

1 to: (col1 size) do:[i]((col1 at:i) verEstado=false) ifTrue:[col2 add:(col1 at:i)] ].

1 to: (col2 size) do:[i](col2 at:i) verTitulo inspect].

## Implementación de la clase Biblioteca

### 3) Implementación de la clase Biblioteca

- **Clase Biblioteca**

**Subclase de:** Object

**Variables de instancia:** nombre libros

**Métodos de clase:**

**>>crearBiblioNom:unNom**

“Retorna una instancia de Biblioteca inicializada”

^(self new) iniBiblioNom:unNom.

## Implementación de la clase Biblioteca

### Métodos de instancia:

>>**iniBiblioNom**:unNom

“Inicializa una instancia de Biblioteca”

nombre:=unNom.

libros:= OrderedCollection new.

>>**verNom**

“Retorna el nombre de la Biblioteca”

^ nombre.

>>**modiNom**:unNom

“Modifica el nombre de la Biblioteca”

nombre:=unNom.



## Implementación de la clase Biblioteca

>>**agregarLibro**:unLibro

“Agrega un libro a la Biblioteca”

libros add:unLibro.

>>**eliminarLibro**:unLibro

“Elimina un libro de la Biblioteca”

libros remove:unLibro.

>>**existeLibro**:unLibro

“Retorna V si el libro está en la Biblioteca, F en caso contrario(cc)”

^ libros includes:unLibro.

>> **esVacia**

“Retorna V si la Biblioteca no tiene libros, F en cc”

^ libros isEmpty.

## Implementación de la clase Biblioteca

**>>cantidadLibros**

“retorna el total de libros de la biblioteca”

^ libros size

**>>recuperarLibro: pos**

“retorne el libro de la posición pos de la biblioteca”

^ libros at:pos.

## 4) Actividad 4

- 1) Implementar en el entorno las clases Libro y Biblioteca.
- 2) Implementar los siguientes métodos de la Clase Biblioteca.

>>**buscarLibroIsbn:unIsbn**

“Retorna el libro de Isbn unIsbn si existe, nil en cc”

>> **todosLosLibros**

“Retorna los libros de la Biblioteca”

- 3) Recuperar el libro de isbn 235, si es que no está prestado. Tener en cuenta la Aplicación de la filmina 6.

