

# Programación POO

Manuel Molino Milla      Luis Molina Garzón

31 de octubre de 2014

## Índice

<b>1. Diagramas UML</b>	<b>1</b>
1.1. Ejercicio 0 . . . . .	1
<b>2. Creación de objetos</b>	<b>1</b>
2.1. Ejercicio 1 . . . . .	1
<b>3. Clases</b>	<b>2</b>
3.1. Ejercicio 2 . . . . .	2
<b>4. API de Java</b>	<b>2</b>
4.1. Ejercicio 3. API String . . . . .	2
4.2. Ejercicio 4 . . . . .	2
4.3. Ejercicio 5. API Math . . . . .	3
4.4. Ejercicio 6 . . . . .	3
<b>5. Miscelanea</b>	<b>3</b>
5.1. Ejercicio 7 . . . . .	3
5.2. Ejercicio 8 . . . . .	4
5.3. Ejercicio 9 . . . . .	4
5.4. Ejercicio 10 . . . . .	4

## 1. Diagramas UML

### 1.1. Ejercicio 0

Crea el diagrama UML de las siguientes clases:

- Cilindro.
- Cuenta bancaria.
- Reloj.

Puedes utilizar la aplicacion *dia* para realizar estos diagramas.  
Por otra parte utiliza los atributos y metodos que creas necesarios.

## 2. Creación de objetos

### 2.1. Ejercicio 1

Crea un programa que describa la clase *Coche* con los siguientes atributos:

1. Cantidad de combustible en el depósito.
2. Consumo del coche a los 100 km.

Y los siguientes métodos:

1. Un método para añadir combustible al depósito.
2. Conocer la cantidad de combustible que tiene el depósito.
3. Número de kilómetro que puede recorrer con el combustible que tiene en ese momento.

Crea dos objetos diferentes de la clase *Coche*. Añade 5 litros de gasolina a cada coche. Comprueba el funcionamiento de la clase creando una clase denominada *TestCoche*

NO uses constructores en este ejercicio.

## 3. Clases

### 3.1. Ejercicio 2

Crea una clase denominada *Libro*, que recoja los atributos y metodos que creas conveniente.

Comprueba su funcionamiento con una clase denominada *TestLibro*

## 4. API de Java

### 4.1. Ejercicio 3. API String

La clase *String* de Java se utiliza para el manejo de cadenas de texto. Busca información en la página oficial de Oracle y completa la siguiente información:

nombre método	valor retorno	parámetros	Breve descripción	Ejemplo
length	int	ninguno	Devuelve longitud cadena	amigo 5
substring				
substring				
concat				
replace				
toLowerCase				
toUpperCase				

## 4.2. Ejercicio 4

Crea una clase denominada Palabra que tenga como unica variable de instancia el contenido de la palabra y un constructor que asigne dicho contenido al atributo anterior. Comprueba su funcionamiento y posteriormente crea los siguientes metodos:

- Un metodo que devuelva la palabra en mayuscula.
- Un metodo que devuelva la palabra en minuscula.
- Un metodo que devuelva el numero de letras que tiene dicha palabra.
- Un metodo que reemplace las letras de la palabra. *Ejemplo palabra coco-drilo, parametros o u, valor devuelto cucudrilu*
- Un metodo que devuelva la primera letra de la palabra.
- Un metodo que devuelva la ultima letra de la palabra en mayuscula.

Comprueba el funcionamiento de dichos metodos.

## 4.3. Ejercicio 5. API Math

La clase Math de Java se utiliza para el manejo de funciones matemáticas. Busca información en la página oficial de Oracle y completa la siguiente información:

nombre método	valor retorno	parámetros	Breve descripción	Ejemplo
abs				
max				
min				
random				
round				
sqrt				
cbrt				

#### 4.4. Ejercicio 6

Crea una clase denominada *Matematicas* que tenga como unica variable de instancia un numero de tipo *double*. Utiliza *getter* y *setter* que accedan y asignen valor al numero respectivamente. Comprueba su funcionamiento y posteriormente crea los siguientes metodos:

- Un metodo que devuelva la raiz cuadrada de dicho número.
- Un metodo que devuelva la raiz cúbica de dicho número en valor absoluto.
- Un metodo que redondee el número de tipo *double* a entero. En el caso que sea un número negativo debe devolver su valor positivo redondeado.
- Un metodo que devuelva número aleatorios en el intervalo de 0 al numero de la clase redondeado. Ejemplo si el valor del numero es 2.3, su redondeo es 2, por tanto debe devolver aleatoriamente los números 0, 1 y 2.

Comprueba el funcionamiento tanto con números positivos como negativos.

### 5. Miscelanea

#### 5.1. Ejercicio 7

Busca informacion sobre que es NaN en Java. Propon ejemplos de su uso.

#### 5.2. Ejercicio 8

Queremos programa con el paradigma de POO una clase que resuelva ecuaciones de segundo grado. Usa los atributos y métodos que creas oportuno. Usa un constructor para crear objetos de esta clase.

Crea una clase denominada *TestEcuacionSegundoGrado* para comprobar su correcto funcionamiento.

#### 5.3. Ejercicio 9

Igual que antes, queremos un programa que implementa la clase *TrianguloRectangulo* usando los atributos que consideres oportuno y métodos para devolver el valor de la hipotenusa, el área del mismo, así como el perímetro de dicho triangulo.

Utiliza *getters* y *setters* y NO constructores. Comprueba el funcionamiento con una clase denominada *TestTrianguloRectangulo*.

Utiliza la API *javax.swing.JOptionPane* para solicitar los valores de los cate-tos de dicho triangulo y también para mostrar los datos de los métodos creados.

#### 5.4. Ejercicio 10

Crea un programa que lea la siguiente información:

- Nombre del empleado.
- Horas trabajada por semana.
- Sueldo bruto por hora.
- Deducción para hacienda.
- Deducción a la seguridad social.

El programa debe mostrar el sueldo bruto, las deducciones totales y el sueldo neto por semana.

Realiza el ejercicio usando cuadro de diálogos para solicitar y mostrar datos.