

# Programación Frontend y Backend

*BLOQUE JAVA*

Conceptos Básicos Java



01

## Conceptos Básicos



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial

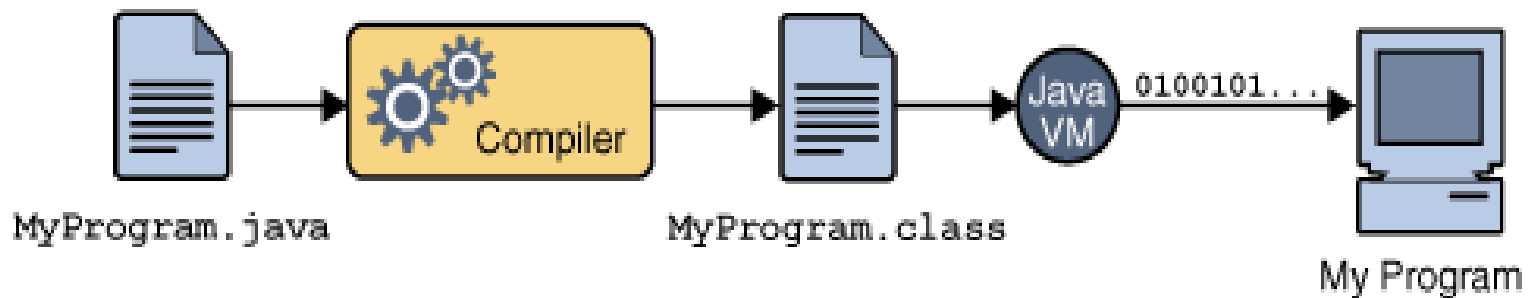


**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro

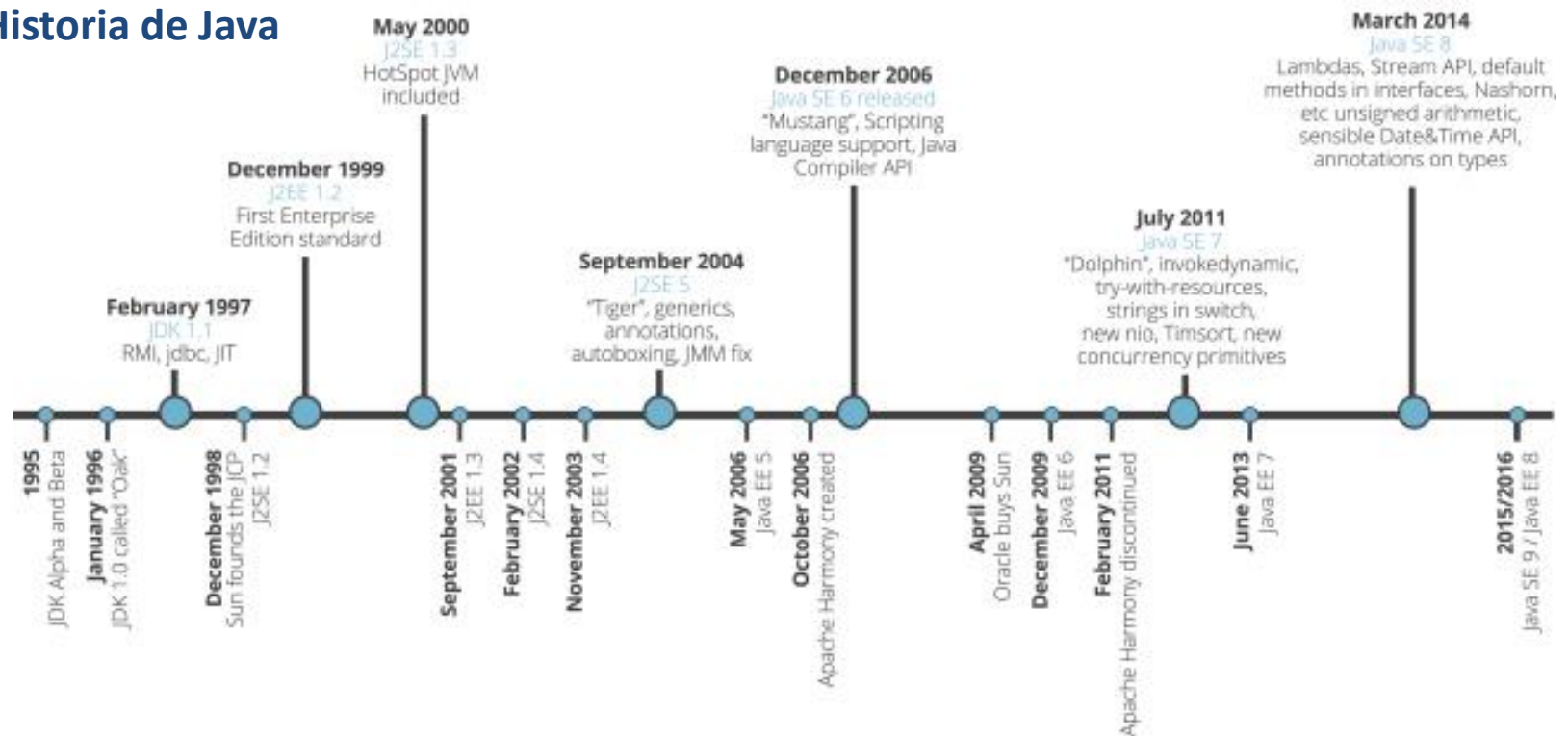


## ¿Qué es Java?

- Lenguaje de programación desarrollado por Sun Microsystems (posteriormente comprada por Oracle)
- Corre sobre una máquina virtual JVM (mismo código compatible en diferentes arquitecturas) que interpreta ficheros .class



## Historia de Java



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial



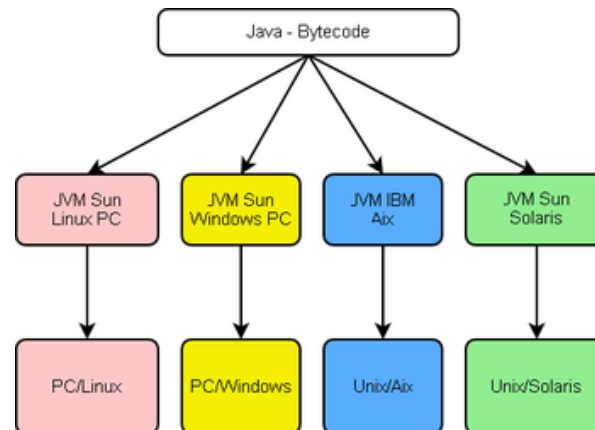
Unión Europea  
Fondo Social Europeo  
Iniciativa de Empleo Juvenil  
El FSE invierte en tu futuro



## JVM

### Definición

- Es la máquina virtual o programa que provee el entorno de ejecución donde java se ejecuta
- Interpreta ficheros .class
- Diferentes máquinas virtuales para diferentes arquitecturas (multiplataforma)



## Objetos y Clases

- Java es un lenguaje de programación orientado a objetos (POO) que soporta los siguientes conceptos
  - Polymorphism
  - Inheritance
  - Encapsulation
  - Abstraction
  - Classes
  - Objects
  - Instance
  - Method
  - Message Passing (o llamada a método, el nombre del método y sus parámetros, en términos generales, su signatura)

## Objetos y Clases

### Objeto

- Un objeto es una instancia de una clase
- La instancia se guarda en memoria (heap) y una referencia apunta a ella (stack). Lo veremos más adelante.
- Los objetos tiene estados y comportamientos (muy similar al mundo real)
  - Estados: nombre, color, tamaño
  - Comportamientos: ladrar, correr, saltar



## Objetos y Clases

### Clase

- Es el molde que usan los objetos para crearse (copia original, contrato, etc...)
- Las clases y java (como en unix) son sensibles a Mayúsculas y Minúsculas (no es lo mismo CarPart que carPart o carpart)

```
1  public class Cube {  
2  
3      private int length;  
4      private int breadth;  
5      private int height;  
6  
7      public int getVolume() {  
8          return (length * breadth * height);  
9      }  
10 }
```



## Objetos y Clases

Instanciar una clase

```
public class CubeFactory {  
    Cube cube = new Cube();  
}
```

### JRE / JDK

#### JRE

- Contiene la máquina virtual Java que interpreta los “bytecodes” de los programas.
- Contiene el conjunto de librerías y paquetes que todo programa Java utiliza y que vienen con la máquina virtual Java por defecto.
- Se utiliza en entornos de sólo ejecución, en entornos finales.

### JRE / JDK

#### JDK

- Contiene JRE
- Contiene utilidad de compilación (*javac*).
- Otras utilidades, ejemplos y documentación.

### JVM

#### Garbage Collector

- Por simplificar se encarga de liberar la memoria, eliminando las referencias no referenciadas.
- más detalles en <http://www.journaldev.com/2856/java-jvm-memory-model-memory-management-in-java>

## Palabras Reservadas

- La lista de palabras que no se deben utilizar bajo ninguna circunstancia como nombre de ninguna variable, método o clase.
- <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/keywords.html>

### Comentarios y documentación

`/* */` Comentar bloques

`//` Comentar líneas

`/** */` Indica documentación

02

## Empezando a Desarrollar



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial



**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro





## Estructura de un proyecto java

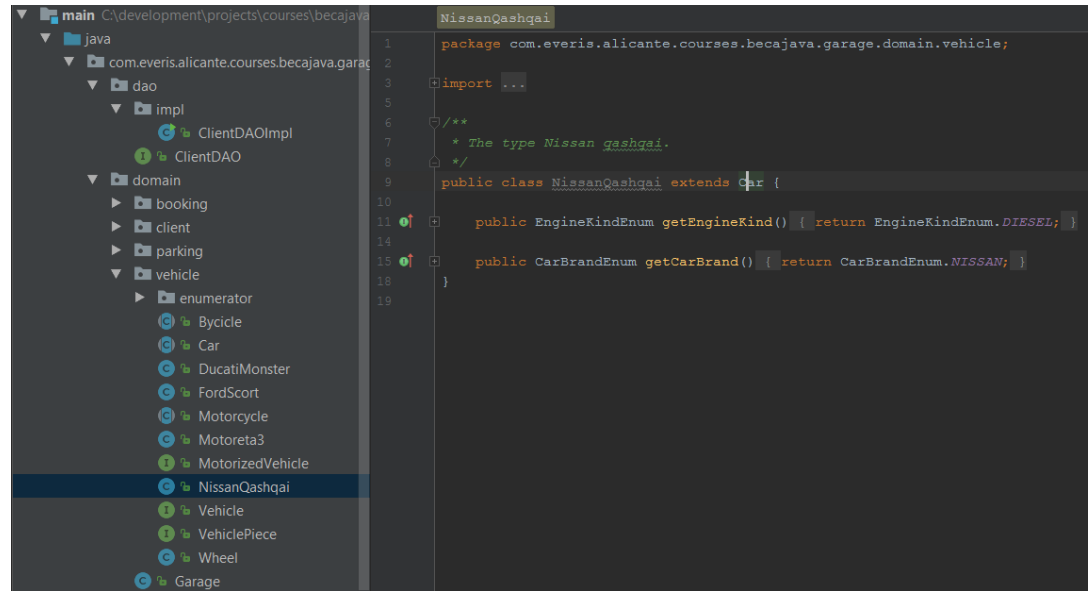
### Conceptos Básicos

- Las clases se guardan en ficheros .java que se organizan en carpetas
- Cada nivel de carpeta representa un nivel de paquete siendo el raíz la ausencia de paquete
- Los paquetes son determinantes desde el punto de vista de la visibilidad de las clases
  - Las clases en el mismo paquete son accesibles
  - Las clases en diferentes paquetes necesitan ser públicas
- Un conjunto de clases puede ser empaquetado en un contenedor denominado jar

```
C:\development\projects\courses\becajava07>jar cvf pepe.jar .
```

## Estructura de un proyecto java

### Paquete II



| Este equipo > OSDisk (C:) > development > projects > courses > becajava07 > garage > src > main > java > com > everis > alicante > courses > becajava > garage > domain > vehicle > |                       |                      |                     |        |
|---|-----------------------|----------------------|---------------------|--------|
|   | Nombre                | Fecha de modifica... | Tipo                | Tamaño |
|   | enumerator            | 03/07/2017 8:44      | Carpeta de archivos |        |
|   | Bicycle.java          | 29/06/2017 16:47     | IntelliJ IDEA       | 1 KB   |
|   | Car.java              | 29/06/2017 16:36     | IntelliJ IDEA       | 1 KB   |
|   | DucatiMonster.java    | 29/06/2017 16:37     | IntelliJ IDEA       | 1 KB   |
|   | FordScort.java        | 29/06/2017 16:36     | IntelliJ IDEA       | 1 KB   |
| 16  | Motorcycle.java       | 29/06/2017 16:37     | IntelliJ IDEA       | 1 KB   |
|   | Motoreta3.java        | 29/06/2017 16:47     | IntelliJ IDEA       | 1 KB   |
| on  | MotorizedVehicle.java | 29/06/2017 16:36     | IntelliJ IDEA       | 1 KB   |
|   | NissanQashqai.java    | 29/06/2017 16:36     | IntelliJ IDEA       | 1 KB   |

## Estructura de un proyecto java

### Paquete I – Convención

- Se escriben siempre con cualquier carácter del código ASCII en minúsculas, separando las palabras por un punto y sin espacios
- Empezando siempre por com, edu, gov, mil, net, org o uno de los códigos de dos letras establecidos por ISO-3166 <https://www.iso.org/obp/ui/#search>
- En caso de tener que utilizar palabras reservadas de java por convención se añade \_ al final del nombre. Por ejemplo com.int\_.example
- Los paquetes propios de java empiezan por java o javax.

Exemplos:

- Correctos
  - com.sun.eng
  - com.apple.quicktime.v2
  - edu.cmu.cs.bovik.cheese
- Incorrectos
  - com.sun\_eng (por la separación entre sun\_eng)
  - com.long.example (uso de palabra reservada)

## Estructura de un proyecto java

### Fichero .java

- Las clases se definen en ficheros con extensión .java

// Filename: NewApp.java

```
// PART 1: (OPTIONAL) package declaration
package com.company.project.fragilePackage;
```

```
// PART 2: (ZERO OR MORE) import declarations
import java.io.*;
import java.util.*;
```

```
// PART 3: (ZERO OR MORE) top-level class and interface declarations
public class NewApp { }

class AClass { }

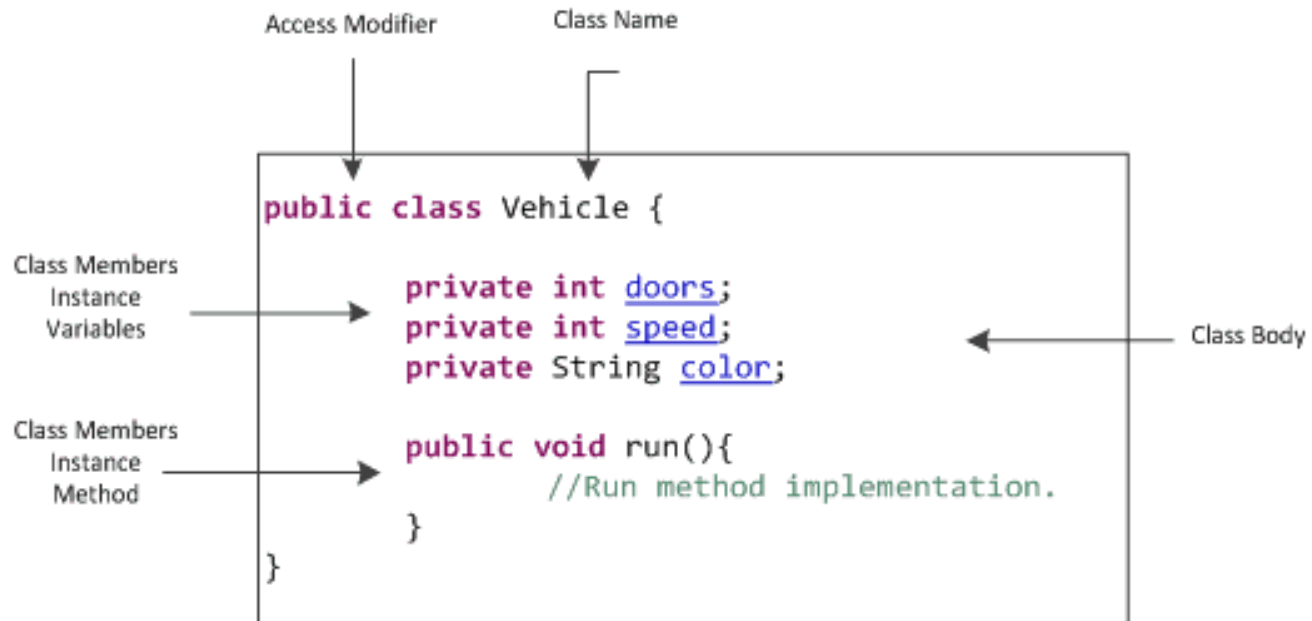
interface IOne { }

class BClass { }

interface ITwo { }
// ...
// end of file
```

## Clase java

### Body Básico



## Ejecutar una aplicación java

### Clase

```
public class AplicacionFlipante {  
    public static void main(final String[] args){  
        System.out.println("Eres guapísimo!!");  
    }  
}
```

- Compilar

```
C:\development\projects\courses\becajava07\garage\src\main\java>javac AplicacionFlipante.java
```

- Ejecutar

```
C:\development\projects\courses\becajava07\garage\src\main\java>java AplicacionFlipante  
Eres guapísimo!!
```