

Laboratorio Introducción A La Programacion Y Computacion 1



Junio 2021
Oliver Ernesto Sierra Pac

Oliver Ernesto Sierra Pac

Profesor

sierra.oe@cunoc.edu.gt

Twitter: @moroquin



Agenda

Unidad 5 - Introducción a java

- Generalidades

- Java virtual Machine

- Proceso de compilación y ejecución

- Estructura de una clase en java

- Método Main

- JShell

- Tipos de datos

- Declaración de variables

- Inicializando las variables

Generalidades



Acuerdos

- ▶ Silenciar el micrófono si no está hablando.
- ▶ Aportar con ideas o preguntas en cualquier momento.
- ▶ Cada sesión se graba.
- ▶ Encender la cámara es opcional.
- ▶ Receso de 10 minutos a cada hora.

Importante

- ▶ Programar es fácil
- ▶ Leer documentación es importante
- ▶ Practicar hasta dominar los conceptos.
- ▶ No buscar recetas
- ▶ Los conceptos básicos primero



Qué hacer después de bloquear 10 años

1

Introducción a java

Write once, run anywhere





Ciclos de 6 meses.
Java 11 LTS.

- ▶ Orientado a objetos
- ▶ Plataforma independiente



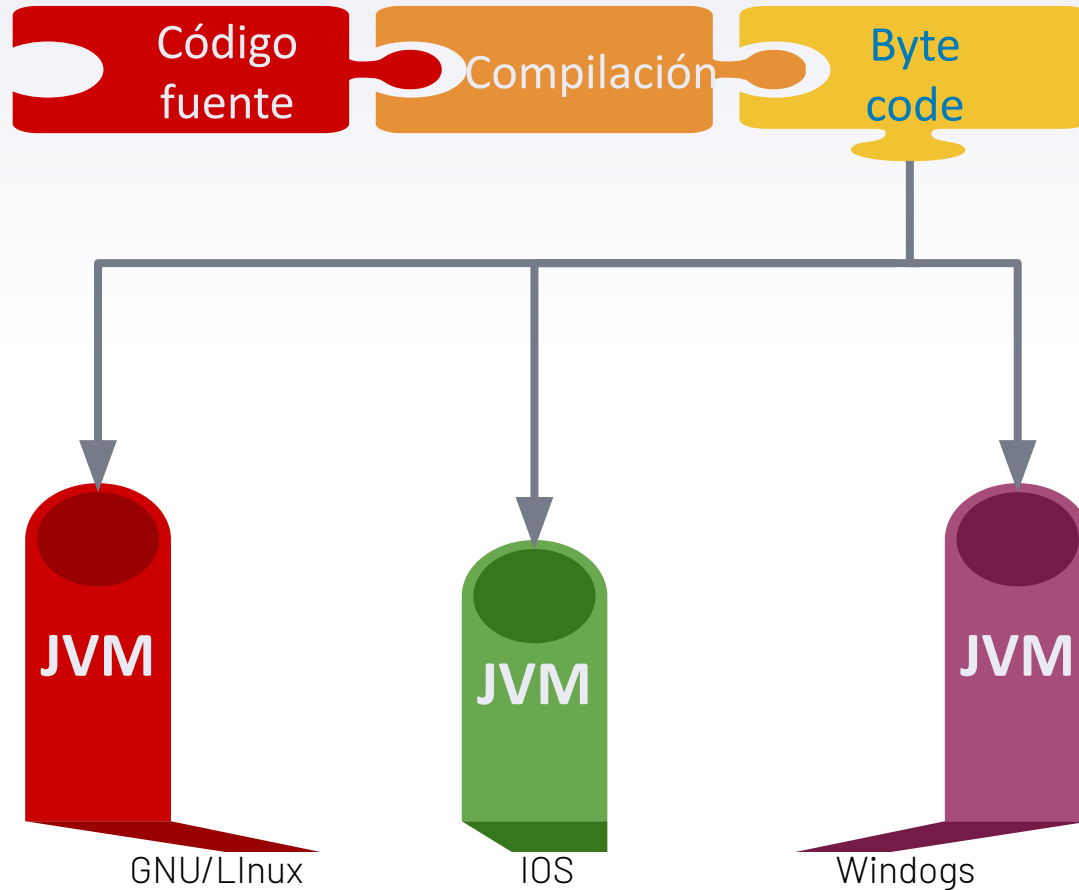
Java Editions

- ▶ Java Card - Smart card edition
- ▶ Java ME - Micro Edition
- ▶ Java SE - Standar Edition
- ▶ Java MP - MicroProfile
- ▶ Java EE - Enterprise Edition

Java SE - es la base en la cual las otras ediciones se basan.



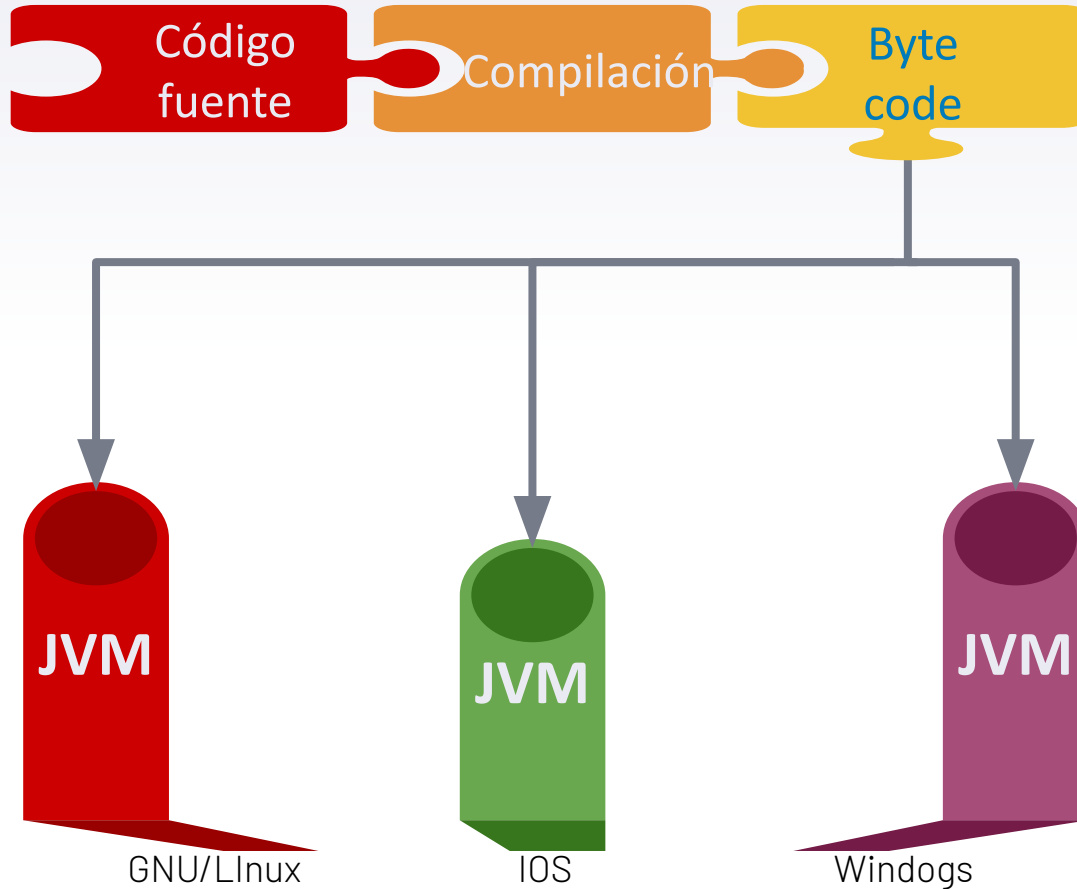
Funcionamiento JAVA



Se compila una vez, y funciona en cualquier jvm

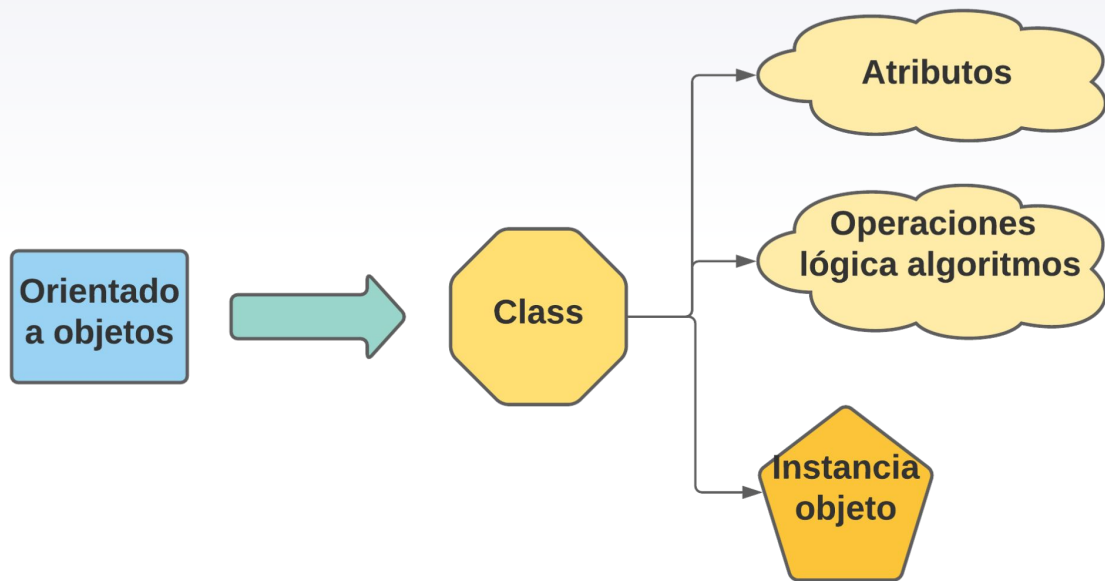
Portabilidad

Funcionamiento JAVA



JVM aísla la plataforma del código y el código de la plataforma (environment)

Organización de java



Para utilizar un objeto, se puede referenciar utilizando una variable de un tipo coherente.

En tiempo de ejecución, objetos hacen llamadas a operaciones entre ellos para ejecutar la lógica de un programa



Keywords reserved

if	implements	boolean	assert
else	extends	try	enum
continue	interface	catch	module
break	class	finally	requires
for	static	throw	transitive
do	final	throws	exports to
while	return	new	uses
switch	transient	this	provides
case	void	super	with
default	byte	instanceof	opens to
private	short	native	true
protected	int	synchronized	false
public	long	volatile	null
import	char	goto	var
package	float	const	
abstract	double	strictfp	

- Desde 1.0
- No se utiliza
- Agregado en 1.2
- Agregado en 1.4
- Agregado en 5.0
- Agregado en 9.0
- Reservado valores literales
- Agregado en 10

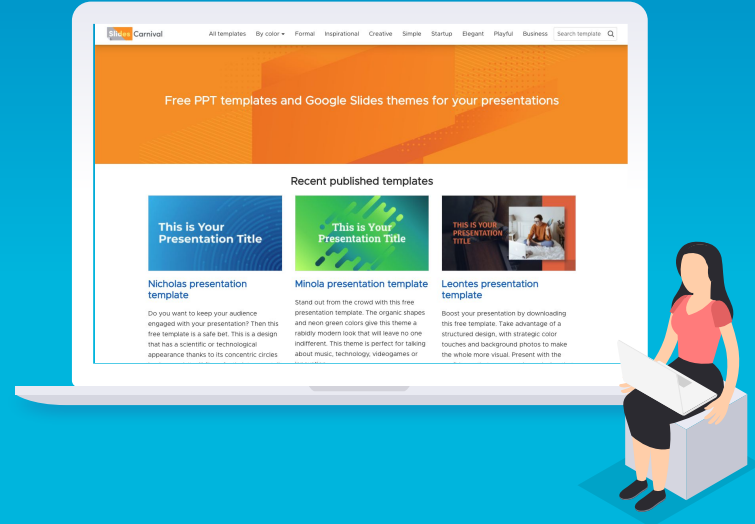



► Convenciones de nombres

- ▶ Case sensitive
- ▶ Packages es contrario al nombre de la compañía
- ▶ Clase debe ser un sustantivo (uppercase)
- ▶ Métodos deben ser un verbo
- ▶ Variables (lowercase)
- ▶ Constantes (mayúscula y separación con _)
- ▶ Nombres
 - ▶ No iniciar con números, _, \$
 - ▶ _ genera una advertencia del compilador en java 8, y error en los posteriores

Sintaxis básica

- ▶ Instrucciones terminan con ;
- ▶ Bloques de código deben encerrarse en { }
- ▶ Indentación es una buena práctica
- ▶ Si no se define un package, default
- ▶ Packages representan folders
- ▶ Package y nombre de la clase debe ser único
- ▶ Para acceder otros paquetes debe importar





```
public class Principal{  
    public static void main(String[] args){  
        System.out.println("Hola Mundo");  
    }  
}
```

`javac -cp /project/classes -d /project/classes /project/source.java`

-classPath

-cp

Clases
necesarias para
la compilación



-d

Resultado de la
compilación



Path

Ruta del código fuente a
compilar

`javac -cp /build org/.../Principal`

-classPath

-cp

De la compilación



Documentación

```
//Comentario de una línea
```

```
/*
```

```
    comentario multi línea
```

```
*/
```

```
/**
```

```
 * javadoc
```

```
 */
```

Tipos primitivos

- ▶ Numeros (valor por defecto 0 ó 0L)
 - ▷ byte
 - ▷ 8 bits, -128,127
 - ▷ short
 - ▷ 16 bits, -32768, 32767
 - ▷ int
 - ▷ 32 bits , -2147483648, 2147483647
 - ▷ long(L)
 - ▷ 64 bits, -9×10^{-18} , -9×10^{18}

▶ Tipos primitivos

- ▶ Boolean
 - ▶ true
 - ▶ false

Tipos primitivos

- ▶ Flotantes
 - ▶ Float (F)
 - ▶ 32 bits 1.4^{-45} , 3.40282335^{38}
 - ▶ double
 - ▶ 64 bits 4.9^{-324} , 1.7976931348623157^{308}

Tipos primitivos

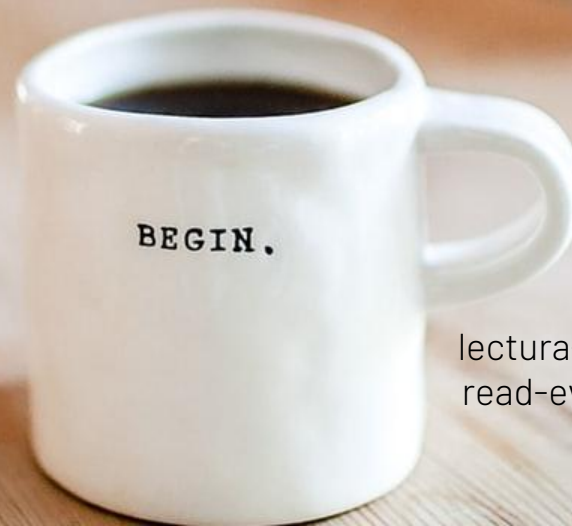
- ▶ Carácter
 - ▷ char
 - ▷ 16 bits 0 - 65535
 - ▷ Por defecto: `'\u0000'`
 - ▷ `'A'`
 - ▷ `;\101'`
 - ▷ `'\t'`



```
public class Principal{
```

```
    public static void main(String[] args){  
        // <type> <variableName> = <value>;  
        // <type> <variableName> ;  
        // <variableName> = <value> ;  
        // Debe inicializarse antes de usarse  
        int t = 0;  
    }
```

```
}
```



Es un ciclo interactivo de lectura-evaluación-impresión (REPL read-evaluate-print loop) de línea de comando

Tarea

Investigación

- ▶ Java operators
 - ▷ Todos los operadores
 - ▷ Incremento y decremento según la posición en que se utilice
- ▶ Flow control
 - ▷ `== !=`
- ▶ Short circuit evaluation
 - ▷ `&& ||`
- ▶ Clase Scanner
 - ▷ Métodos de ingreso de datos, `int`, `string` . . .

¡Gracias!

¿Preguntas?

