

Java: conceptos básicos

Alejandro Traver Fos



KEEPCODING

Tech School

¿Qué aprenderemos en este tema?

1. Funciones / métodos
2. Packages y ámbitos
3. Tipos básicos
4. Operadores básicos
5. Operaciones con Strings
6. If-else
7. Switch
8. Clases envoltorio
9. ArrayList e iteración



1 | Operadores básicos: **Funciones / métodos**

- public y static lo veremos más adelante
- El tipo de retorno
- Nombre de la función
- Parámetros de entrada separados por comas
- Llaves para delimitar el cuerpo de la función

```
public static void fun(int a) {  
}
```

2 | Operadores básicos: **packages** y **ámbitos**

- Normalmente en Java los ámbitos se delimitan mediante llaves `{}`
- Los packages son un mecanismo de Java para facilitar la gestión de los ámbitos
 - Estructura similar a directorios que permiten referenciar código de otros proyectos

3 | Operadores básicos: **Tipos básicos**

- Son los que empiezan por minúscula
 - `int` -> Número entero de 1B
 - `long` -> Número entero de larga longitud (2B)
 - `float` -> Número en coma flotante (4B)
 - `double` -> Número en coma flotante de doble precisión (8B)
 - `char` -> caracter
 - `boolean` -> `True` / `False`
 - Otros: `byte`, `short`

4 | Operadores básicos: **Operadores básicos**

- Operadores **aritméticos** (+, -, x, /)
- Operadores de **asignación** (=, +=)
- Operadores de **comparación**: == (igual a), != (diferente de), < (menor que), > (mayor que), <= (menor o igual que) y >= (mayor o igual que).
- Operadores **Lógicos**: && (AND lógico), || (OR lógico) y ! (NOT lógico).
- Operadores de **incremento y decremento** (++ , --)

- Se pueden construir directamente con las dobles comillas ""
- No son tipos básicos
- Operación básicas:
 - `length()`
 - `charAt()`
 - `substring()` y `contains()`
 - Concatenación: `+`
 - Doc:<https://docs.oracle.com/javase/8/docs/api/java/lang/String.html>

6 | Operadores básicos: **if-else** (I)

- La condición entre paréntesis y {} para separar los ámbitos de cada condición
- Se pueden anidar múltiples condiciones con if – else if

```
public static void main(String [] args) {  
    boolean goToCinema = true;  
  
    if (goToCinema) {  
        buyTickets();  
    } else {  
        goToSofa();  
    }  
}
```


6 | Operadores básicos: **if-else** (II)

- Operadores **ternarios**: condición
? valor_si_verdadero :
valor_si_falso.
- Operadores de **igualdad**
- Operadores de **concatenación**

Y... muchos más!



```
if (condition) {  
    return A;  
} else {  
    return B;  
}
```



```
return condition ? A : B;
```

7 | Operadores básicos: **switch**

- El switch-case es una estructura que permite gestionar múltiples opciones if - else if -else
- Si se cumple una condición se ejecuta y el resto no se evalúan
- Si no se ejecuta ninguna se ejecuta default

```
public static void main(String [] args) {  
    String option = "a";  
    String res = "";  
    switch (option) {  
        case "a":  
            res = "Go to cinema";  
            break;  
        case "b":  
            res = "Go to restaurant";  
            break;  
        default:  
            res = "Stay at home";  
            break;  
    }  
  
    System.out.print(res);  
}
```

8 | Operadores básicos: **Clases envoltorio**

- Son con mayúscula y tienen propiedades similares a los básicos
- Herramientas para cambiar de tipos
 - `toString()`
 - `valueOf()`

```
public static void main(String [] args) {  
    Integer a = Integer.valueOf(5);  
    System.out.print(a);  
}
```

- Para representar una lista, la estructura más básica es **ArrayList**
- Operaciones básicas:
 - `add()`
 - `get()`
 - `length()`
 - `set`
 - `size()`, `contains()`
 - ...

```
public static void main(String [] args) {  
    ArrayList<String> stringList = new ArrayList<>();  
  
    stringList.add("Uno");  
    stringList.add("Dos");  
    stringList.add("Tres");  
    stringList.add("Cuatro");  
    stringList.add("Cinco");  
  
    for (String element : stringList) {  
        System.out.println(element);  
    }  
}
```



KEEPCODING

Tech School

Madrid | Barcelona | Bogotá

Datos de contacto

Alejandro Traver Fos
alextrfos@gmail.com