

Ariel Alejandro

Declaración de Métodos y Método main en Java

Métodos en Java

Un **método** es un bloque de código que realiza una tarea específica y puede ser reutilizado varias veces en un programa.

- Los métodos **se definen dentro de una clase** y se pueden invocar tanto dentro de la clase como desde otras clases (dependiendo de la visibilidad del método).
- El uso de métodos ayuda a **estructurar un programa**, lo que facilita la legibilidad y el mantenimiento del código.



Estructura de un Método

a) Modificadores de Acceso

- Ejemplos: *public*, *private*, *protected*, *sin modificador (default)*:

b) Tipo de Retorno

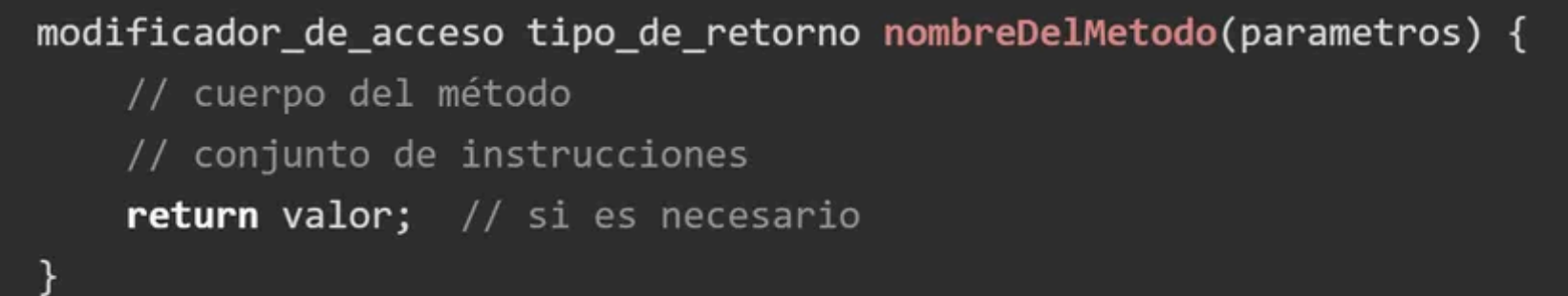
- Ejemplos: *int*, *double*, *String*, *void*, *etc*

c) Nombre del Método (notación camelCase)

- Ejemplo: *imprimirMensaje*

d) Parámetros

e) Cuerpo del Método



```
modificador_de_acceso tipo_de_retorno nombreDelMetodo(parametros) {  
    // cuerpo del método  
    // conjunto de instrucciones  
    return valor; // si es necesario  
}
```

Método main en Java

El método **main** es el punto de entrada de cualquier aplicación Java. Cuando se ejecuta un programa, la **Máquina Virtual de Java (JVM)** busca este método para comenzar la ejecución del programa.



```
public static void main(String[] args) {  
    // cuerpo del método  
}
```

Ejemplo de Programa Java con el Método main y Métodos Adicionales

```
private void executeLoad(long timeout, int usersCount) {  
    showDebugInfo(timeout);  
    Load.setPages(URL, parsingTimeout);  
    Load.setTimeout(timeout);  
    List<Load> threads = new ArrayList<>();  
    for (int i = 0; i < usersCount; i++) {  
        threads.add(new Load(this.URL));  
    }  
    logger.info( s: usersCount + " threads are created");  
    for (Load thread : threads) {  
        thread.start();  
    }  
    logger.info( s: "All threads are started");  
    progressInfo(timeout);  
    System.out.print(".....DONE\nProcessing with data...\n");  
}  
  
private void executeAvailability(long timeout, int usersCount) {  
}  
  
private void executeSimulation...
```

Referencias:

- Bloch, J. (2018). **Effective Java** (3rd Edition). Addison-Wesley. Sección **Item 47: Prefer Method References to Lambdas** (pp. 207-210), que explica el uso y la implementación adecuada de métodos en Java.
- Oracle. (n.d.). *The Java™ Tutorials: Methods*. Disponible en <https://docs.oracle.com/javase/tutorial/java/javaOO/methods.html>, específicamente en el apartado **Declaring Methods**, donde se detalla la sintaxis, tipos de retorno, y parámetros de métodos en Java.
- Sierra, K., & Bates, B. (2005). **Head First Java** (2nd Edition). O'Reilly Media. Capítulo 2: **A Trip to Objectville** (pp. 79-83), donde se abordan los fundamentos de la declaración de métodos y el uso del método main.