

Declaración de Métodos y Método main en Java

Padlet: [Link](#)

1. Introducción a los Métodos en Java

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.

2. Estructura General de un Método en Java

La declaración de un método en Java sigue una sintaxis bien definida. A continuación, se presenta el formato básico:

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

a) Modificadores de Acceso Los modificadores de acceso definen la visibilidad del método dentro y fuera de su clase. Los más comunes son:

public: El método puede ser accedido desde cualquier clase. **private:** El método solo puede ser accedido dentro de la clase en la que está declarado. **protected:** El método puede ser accedido dentro del paquete y por las subclases. **sin modificador (default):** El método solo es accesible desde clases dentro del mismo paquete. b) Tipo de Retorno El tipo de retorno especifica qué tipo de valor devuelve el método. Puede ser cualquier tipo de datos (como `int`, `double`, `String`, etc.), o puede ser `void`, si el método no devuelve ningún valor.

Ejemplo de un método con retorno:

```
java Copy code public int suma(int a, int b) { return a + b; }
```

c) Nombre del Método El nombre del método debe ser significativo y seguir las convenciones de nomenclatura en Java, es decir, usar notación camelCase.

Ejemplo de un nombre de método correcto:

```
java Copy code public void imprimirMensaje() { System.out.println("Hola, Mundo"); }
```

d) Parámetros Los parámetros son variables que el método recibe como entrada. Los parámetros se definen con un tipo de dato seguido de un nombre. Un método puede recibir cero o más parámetros.

Ejemplo de un método con parámetros:

```
java Copy code public void saludar(String nombre) { System.out.println("Hola, " + nombre); }
```

e) Cuerpo del Método El cuerpo contiene el conjunto de instrucciones que el método ejecuta. Puede incluir declaraciones de variables, cálculos, estructuras de control (como condicionales y bucles), y llamadas a otros métodos.

3. 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.

La declaración estándar del método main es:

java Copy code public static void main(String[] args) { // cuerpo del método } a) Modificador public El método main es público porque debe ser accesible desde fuera de la clase para que la JVM pueda invocarlo.

b) Modificador static El método main es estático porque se ejecuta sin necesidad de crear una instancia de la clase. Esto permite que la JVM lo invoque directamente sin crear un objeto de la clase que lo contiene.

c) Tipo de Retorno void El tipo de retorno del método main es void, ya que no necesita devolver ningún valor a la JVM después de su ejecución.

d) Parámetro String[] args El parámetro String[] args es un arreglo de cadenas que permite que se le pasen argumentos desde la línea de comandos cuando se ejecuta el programa. Aunque no siempre se utilice, es fundamental para permitir que el programa reciba datos externos.

4. Ejemplo de Programa Java con el Método main y Métodos Adicionales

```
public class MiPrograma {  
  
    // Método principal  
    public static void main(String[] args) {  
        // Llamada a otros métodos  
        saludar();  
        int resultado = sumar(10, 5);  
        System.out.println("El resultado de la suma es: " + resultado);  
    }  
  
    // Método sin parámetros y sin retorno  
    public static void saludar() {  
        System.out.println("¡Hola, bienvenido a mi programa!");  
    }  
  
    // Método con parámetros y con retorno  
    public static int sumar(int a, int b) {  
        return a + b;  
    }  
}
```

Explicación del código: El programa contiene un método main que es el punto de inicio de la ejecución.

Dentro del método main, se invocan dos métodos adicionales: saludar y sumar. El método saludar no recibe parámetros ni retorna ningún valor; simplemente imprime un mensaje. El método sumar recibe dos enteros como parámetros y devuelve la suma de ellos.

5. Consideraciones sobre el Uso de Métodos Reutilización de código: Al definir métodos, el código se puede reutilizar varias veces sin tener que escribir el mismo bloque de instrucciones repetidamente.

Legibilidad: Los métodos permiten que el código sea más legible al dividir tareas complejas en subprocesos más manejables. Mantenimiento: Si es necesario cambiar una funcionalidad, solo es necesario modificar el método correspondiente, lo que facilita el mantenimiento del código.

6. Conclusión La declaración de métodos en Java es un concepto clave que permite organizar y estructurar los programas de manera eficiente. El método main es el punto de entrada para cualquier aplicación Java,

mientras que otros métodos adicionales pueden definirse para llevar a cabo tareas específicas dentro del programa. El uso adecuado de métodos mejora la modularidad, legibilidad y mantenibilidad del código.