

Semana 08

Métodos en Java

Concepto

Método en Java o Java Method es una colección de declaraciones que realizan alguna tarea específica y devuelven el resultado a la persona que llama. Un método Java puede realizar alguna tarea específica sin devolver nada. Los métodos en Java nos permiten reutilizar el código sin volver a escribir el código. En Java, cada método debe ser parte de alguna clase.

1. Un método es como una función, es decir, se utiliza para exponer el comportamiento de un objeto.
2. Es un conjunto de códigos que realizan una tarea particular.

Sintaxis

```
<access_modifier> <return_type> <method_name>( list_of_parameters)
{
    //body
}
```

Ventaja de los métodos

- Reutilización de código
- Optimización de código

Declaración de Método

En general, las declaraciones de métodos tienen 6 componentes:

1. Modificador de acceso

Define el tipo de acceso del método, es decir, desde dónde se puede acceder en su aplicación. En Java, hay 4 tipos de especificadores de acceso.

- **public**

- **protected**
- **private**
- **default**

2. Tipo de retorno

El tipo de datos del valor devuelto por el método o void si no devuelve un valor. Es Obligatorio en la sintaxis.

3. Nombre del Método

Las reglas para los nombres de campos también se aplican a los nombres de métodos, pero la convención es un poco diferente. Es obligatorio en la sintaxis.

4. Lista de parámetros

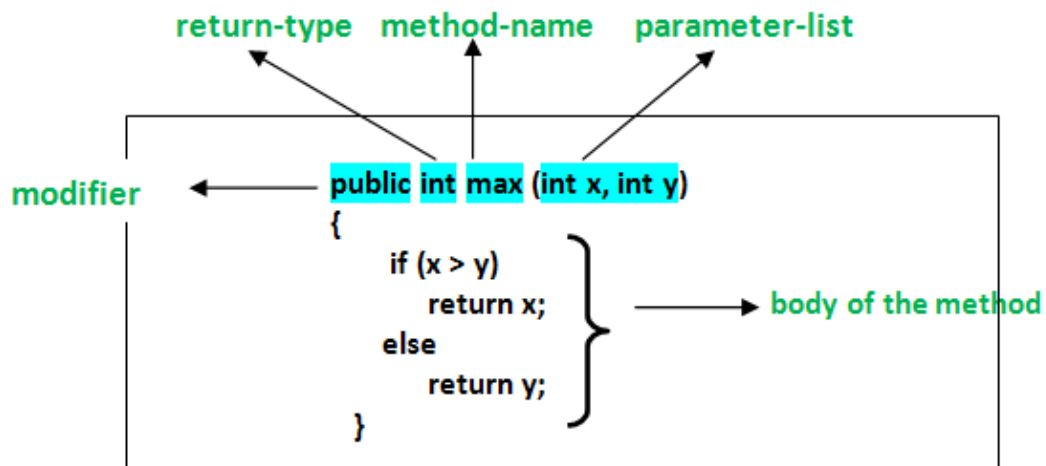
Se define una lista separada por comas de los parámetros de entrada, precedida por su tipo de datos, dentro del paréntesis adjunto. Si no hay parámetros, debe usar paréntesis vacíos (). Es opcional en la sintaxis.

5. Lista de excepciones

Las excepciones que espera que el método pueda generar, puede especificar estas excepciones. Es opcional en la sintaxis.

6. Cuerpo del método

Está encerrado entre las llaves. El código que necesita ejecutar para realizar las operaciones previstas. Es opcional en la sintaxis.



Tipos de métodos en Java

Hay dos tipos de métodos en Java:

1. Método de instancia

Accede a los datos de la instancia utilizando el nombre del objeto. Declarado dentro de una clase.

Sintaxis:

```
// Instance Method
void method_name(){
    body // instance area
}
```

2. Método estático

Acceda a los datos estáticos usando el nombre de la clase. Declarado dentro de la clase con palabra clave estática.

Sintaxis:

```
//Static Method
static void method_name(){
    body // static area
}
```

Firma del método

Consiste en el nombre del método y una lista de parámetros (número de parámetros, tipo de parámetros y orden de los parámetros). El tipo de devolución y las excepciones no se consideran parte de la misma.

Método Firma de la función anterior:

```
max(int x, int y) Number of parameters is 2, Type of parameter is int.
```

¿Cómo llamar a un método?

El nombre de un método suele ser una sola palabra que debe ser un verbo en minúsculas o varias palabras, que comienza con un verbo en minúsculas seguido de un adjetivo, sustantivo... Después de la primera palabra, la primera letra de cada palabra debe ser en mayúscula.

Reglas para nombrar un método

- Al definir un método, recuerde que el nombre del método debe ser un verbo y debe comenzar con una letra minúscula.
- Si el nombre del método tiene más de dos palabras, el primer nombre debe ser un verbo seguido de un adjetivo o sustantivo.
- En el nombre de método de varias palabras, la primera letra de cada palabra debe estar en mayúsculas excepto la primera palabra. Por ejemplo, findSum, computeMax, setX y getX.

Generalmente, un método tiene un nombre único dentro de la clase en la que está definido, pero a veces un método puede tener el mismo nombre que otros nombres de métodos dentro de la misma clase, ya que Java permite la sobrecarga de métodos.

Llamada de métodos

El método necesita ser llamado para usar su funcionalidad. Puede haber tres situaciones cuando se llama a un método.

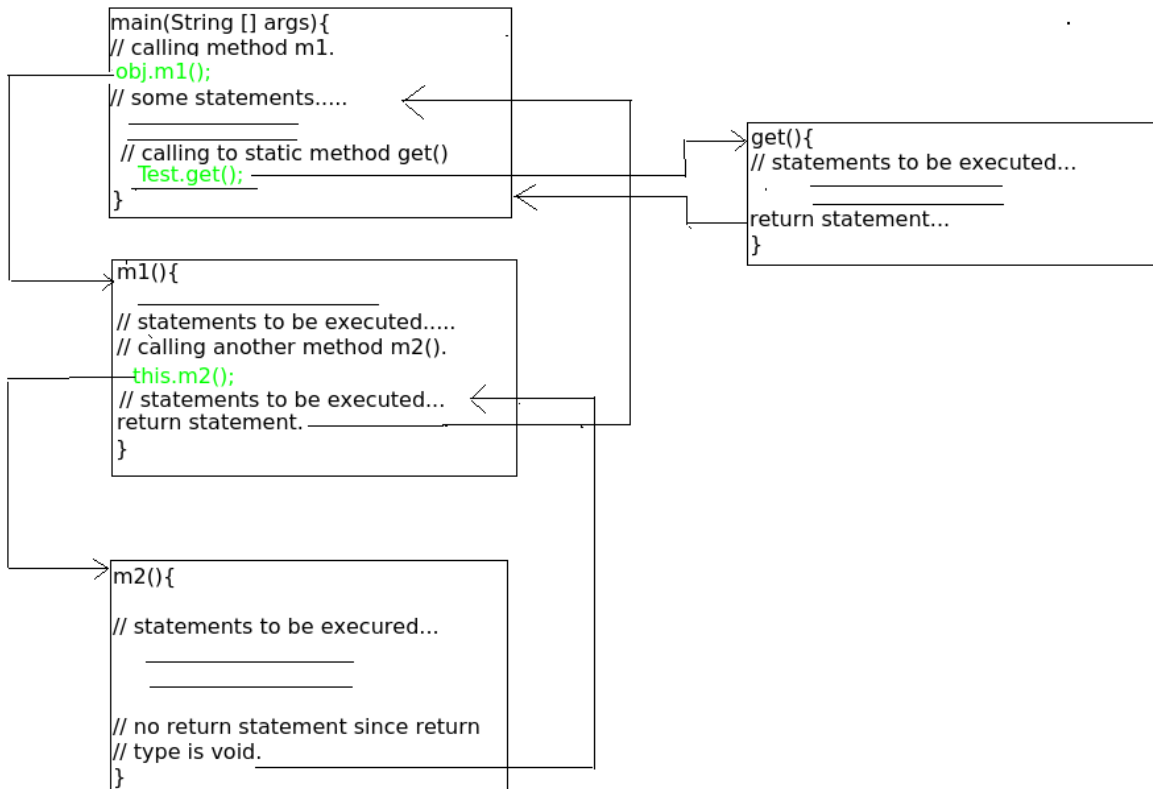
Un método vuelve al código que lo invocó cuando:

- Completa todas las sentencias del método.
- Llega a una declaración de devolución.

- Lanza una excepción.

Ejemplos

El flujo de control del programa anterior es el siguiente:



Asignación de memoria en las Llamadas a Métodos

Las llamadas a métodos se implementan a través de una pila. Cada vez que se llama a un método, se crea un marco de pila dentro del área de la pila y, después de eso, los argumentos pasados y las variables locales y el valor que devolverá este método llamado se almacenan en este marco de pila y cuando finaliza la ejecución del método llamado, el marco de pila asignado se eliminaría. Hay un registro de puntero de pila que rastrea la parte superior de la pila que se ajusta en consecuencia.