Object Oriented Programming

METHOD, CONSTRUCTOR

Class Component

A class in Java can contain:

- Fields
- Methods
- Constructors
- Blocks
- Nested class and interface

Method

- A method is a block of code or collection of statements or a set of code grouped together to perform a certain task or operation.
- It also provides the easy modification and readability of code, just by adding or removing a chunk of code.
- The method is executed only when we call or invoke it.

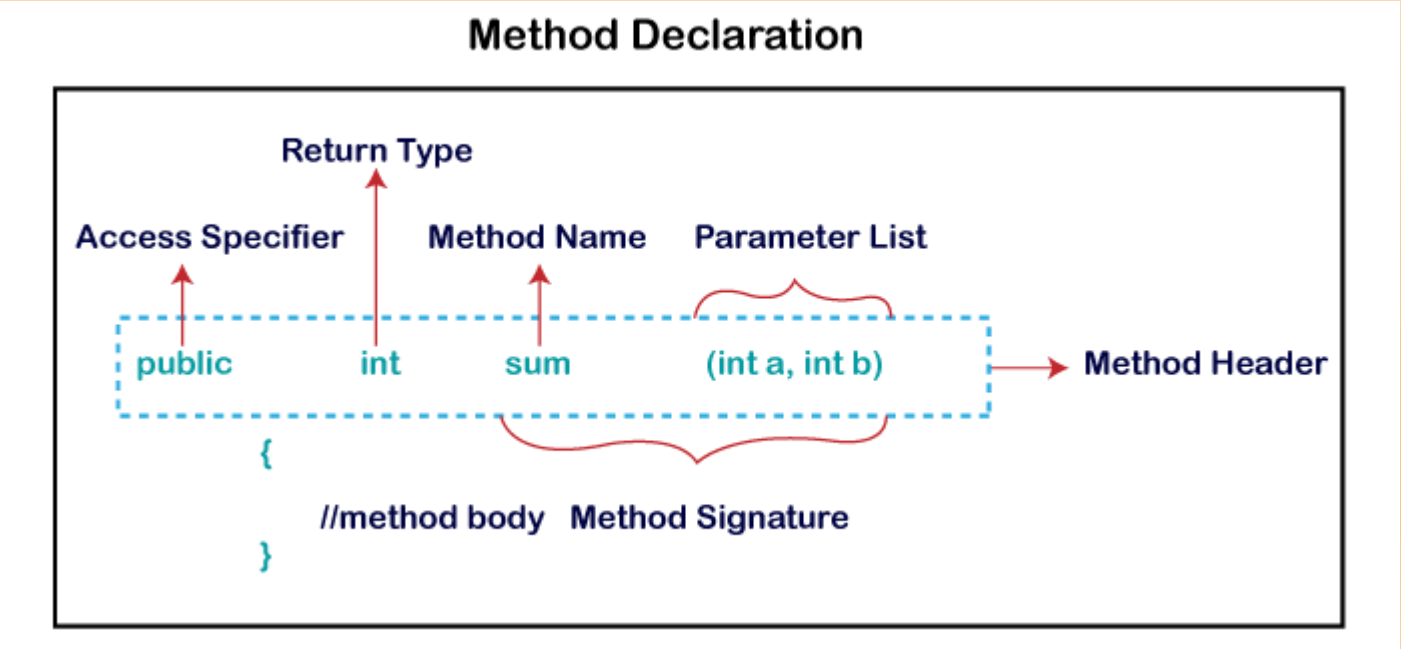
Method

Advantage of Method:

- Code Reusability
- Code Optimization

Method Declaration

The method declaration provides information about method attributes, such as visibility, return-type, name, and arguments.



Method Signature

- Every method has a method signature.
- It is a part of the method declaration.
- It includes the method name and parameter list.

Access Specifier

Access specifier or modifier is the access type of the method. It specifies the visibility of the method.

Java provides four types of access specifier:

Public: The method is accessible by all classes when we use public specifier in our application.

Private: When we use a private access specifier, the method is accessible only in the classes in which it is defined.

Protected: When we use protected access specifier, the method is accessible within the same package or subclasses in a different package.

Default: When we do not use any access specifier in the method declaration, Java uses default access specifier by default. It is visible only from the same package only.

Return Type

- Return type is a data type that the method returns.
- It may have a primitive data type, object, collection, void, etc.
- If the method does not return anything, we use void keyword.

Naming a Method

- Single-word method name: sum(), area()
- Multi-word method name: areaOfCircle(), stringComparision()

Types of Method

There are two types of methods in Java:

- Predefined Method
- User-defined Method

Predefined Method

```
public class Demo {

public static void main(String[] args)
{
// using the max() method of Math class

System.out.print("The maximum number is: " + Math.max(9,7));
}
}
```

User-defined Method

- The method written by the user or programmer is known as a user-defined method.
- These methods are modified according to the requirement.

How to Create a User-defined Method

```
//user defined method
public static void findEvenOdd(int num)
{
//method body

if(num%2==0){
	System.out.println(num+" is even");
}
else {
	System.out.println(num+" is odd");
}
```

How to Call or Invoke a User-defined Method

```
import java.util.Scanner;
public class EvenOdd
{
public static void main (String args[])
{
//creating Scanner class object
Scanner scan=new Scanner(System.in);
System.out.print("Enter the number: ");
//reading value from the user
int num=scan.nextInt();
//method calling
findEvenOdd(num);
}
```

```
//user defined method
public static void findEvenOdd(int num)
{
//method body
if(num%2==0)
System.out.println(num+" is even");
else
System.out.println(num+" is odd");
}
}
```

Static Method

- A method that has static keyword is known as static method.
- In other words, a method that belongs to a class rather than an instance of a class is known as a static method.
- The main advantage of a static method is that we can call it without creating an object.
- It can access static data members and also change the value of it.

Static Method

```
public class Display
{
public static void main(String[] args)
{
show();
}
static void show()
{
System.out.println("It is an example of static method.");
}
}
```

Instance Method

- The method of the class is known as an instance method.
- It is a non-static method defined in the class.
- Before calling or invoking the instance method, it is necessary to create an object of its class.

Instance Method

```
public class InstanceMethodExample
{
public static void main(String [] args)
{
//Creating an object of the class
InstanceMethodExample obj = new InstanceMethodExample();
//invoking instance method
System.out.println("The sum is: "+obj.add(12, 13));
}
```

```
int s;
//user defined method because we have not used static keyword

public int add(int a, int b)
{
s = a+b;
//returning the sum
return s;
}
}
```

Instance Method Types

There are two types of instance method:

- Accessor Method
- Mutator Method

Accessor Method

- The method(s) that reads the instance variable(s) is known as the accessor method.
- We can easily identify it because the method is prefixed with the word get. It is also known as getters.
- It returns the value of the private field. It is used to get the value of the private field.

Accessor Method (Example)

```
public int getId()
{
    return Id;
}
```

Mutator Method

- The method(s) read the instance variable(s) and also modify the values.
- We can easily identify it because the method is prefixed with the word set.
- It is also known as setters or modifiers.
- Does not return anything.
- Accepts a parameter of the same data type that depends on the field.
- It is used to set the value of the private field.

Mutator Method (Example)

```
public void setRoll(int roll)
{
this.roll = roll;
}
```

Thank You