

# java overloading

Topics : [JAVA](#)

Written on [April 10, 2023](#)

Java method overloading is a feature that allows a class to have multiple methods with the same name, but with different parameters. Overloading provides a way to create more readable and reusable code by giving the same method name to different methods that perform different tasks.

When a method is overloaded, the Java compiler determines which method to call based on the number, types, and order of the arguments passed in.

Here's an example of method overloading in Java:

```
public class MyClass {
    public void print(String message) {
        System.out.println(message);
    }
    public void print(int number) {
        System.out.println(number);
    }
    public void print(String message, int number) {
        System.out.println(message + " " + number);
    }
    public void print(int number, String message) {
        System.out.println(number + " " + message);
    }
}

public class Main {
    public static void main(String[] args) {
        MyClass myObj = new MyClass();
        myObj.print("Hello");
        myObj.print(42);
        myObj.print("The answer is:", 42);
        myObj.print(42, "is the answer.");
    }
}
```

In this example, the `MyClass` class has four different `print()` methods with different parameters. The `print()` methods either take a `String`, an `int`, or a combination of both as parameters.

In the `main()` method, we create an instance of the `MyClass` class and call each of the `print()` methods with different parameters. The Java compiler selects the appropriate `print()` method based on the number and types of the arguments passed in.

Depending on the arguments passed in, the output of the program would be:

```
Hello  
42  
The answer is: 42  
42 is the answer.
```

© Copyright **Aryatechno**. All Rights Reserved. Written tutorials and materials by [Aryatechno](#)

ARYATECHNO