



## Introduction:

**Method Overloading** and **Method Overriding** help us use the *same method name* in Java but do *different work*. In **method overloading**, we stay *inside the same class* and write the method in different ways by changing what we pass in (like saying hello to no one, to a name, or many times). In **method overriding**, a *child class* changes how a method from its *parent class* works (like Vehicle moves, but Car drives, Boat sails, and Airplane flies). These features make our code easier to read, reuse, and understand—just like using one simple word (“move”) for many kinds of moving things.

### Quick reminders:

- **Overloading:** Same name, *different inputs*, same class.
- **Overriding:** Same name, *new work*, child class.

## 1. What is Method Overloading?

Method overloading is a feature in java that allows a class to have same name but different parameter list.

### Key Points:

Method must have same name but differ in:

- Number of parameters.
- Types of parameters.
- Order of parameter types.

### Example-1:

// Method with three integer parameters

```
class calculator {  
    void add (int a, int b) {  
        System.out.println(a + b);  
    }  
    void add(double a, double b) {  
        System.out.println(a + b);  
    }  
    void add (int a, int b, int c) {  
        System.out.println(a + b + c);  
    }  
    void add (String a, String b) {  
        System.out.println(a + b); // Adds two words like "Hello" + "World" = HelloWorld  
    }  
}
```

## Example-2:

// Method with library-books

```
class Library {  
    void readBook() {  
        System.out.println("Reading a book");  
    }  
    void readBook(String bookName) {  
        System.out.println("Reading the book: " + bookName);  
    }  
    void readBook(String bookName, String author) {  
        System.out.println("Reading " + bookName + " by " + author);  
    }  
    void readBook(String bookName, int pages) {  
        System.out.println("Reading " + bookName + " with " + pages + " pages");  
    }  
}
```

## 2. What is Method Overriding?

Method overloading is a feature in java that allows when a subclass (child class) has the same method as superclass (parent class).

### Key Points:

Method must have same name:

- Same method name.
- Same Input.
- New work done in child class.

## Example-1:

```
class Animal {  
    void makeSound() {  
        System.out.println("Animal makes a sound");  
    }  
}  
  
class Dog extends Animal {  
    void makeSound() {  
        System.out.println("Dog says: Woof Woof!");  
    }  
}  
  
class Cat extends Animal {  
    void makeSound() {  
        System.out.println("Cat says: Meow Meow!");  
    }  
}
```



### Example-2:

```
class Vehicle {  
    void move() {  
        System.out.println("Vehicle is moving");  
    }  
}  
  
class Car extends Vehicle {  
    void move() {  
        System.out.println("Car is moving on the road");  
    }  
}  
  
class Boat extends Vehicle {  
    void move() {  
        System.out.println("Boat is sailing on the water");  
    }  
}  
  
class Airplane extends Vehicle {  
    void move() {  
        System.out.println("Airplane is flying in the sky");  
    }  
}
```

### 3. Difference between Method Overloading & Method Overriding

Features	Method Overloading	Method Overriding
Meaning	Same method name, but different inputs	Same method name, same inputs, but new work
Class	Happens in the <b>same class</b>	Happens in <b>child class</b> (from parent class)
Purpose	To do similar tasks in different ways	To <b>change</b> the method's behavior
Inputs	Must be <b>different</b> (type or number)	Must be <b>same</b> as parent class method
Examples	add(int, int) and add(double, double)	Animal makes sound, Dog makes Woof!

## Exercises

### 1. Multiple Choice Questions (MCQs)

- Which concept allows using same method name with different parameter in the list?
  - ☐ a) Inheritance
  - ☐ b) Method overriding
  - ☐ c) Method overloading
  - ☐ d) Polymorphism
- What is require for method overloading occur?
  - ☐ a) Same class
  - ☐ b) Different method names
  - ☐ c) Inheritance
  - ☐ d) Different return types
- In method overloading, which of following can be different?
  - ☐ a) Method name
  - ☐ b) Class name
  - ☐ c) Number of type of parameters
  - ☐ d) None
- Method overriding is done between?
  - ☐ a) Two methods in the same class
  - ☐ b) A parent class and child class
  - ☐ c) Two unrelated class
  - ☐ d) None
- Which method is executed when a child class overrides a method and calls it?
  - ☐ a) Parent class method
  - ☐ b) Child class method
  - ☐ c) Both
  - ☐ d) None

### 2. Fill in the Blanks

- Method Overloading occurs in the \_\_\_\_\_ class.
- Method Overriding requires \_\_\_\_\_ between classes.
- In Method Overloading, the method name is the same, but the \_\_\_\_\_ are different.
- Overriding means a child class changes the method from its \_\_\_\_\_ class.
- Method Overriding uses the keyword \_\_\_\_\_ if the parent method is not allowed to change.

### 3. true or False

- Method Overloading can happen between a parent and child class.
- Method Overriding means giving new work to the same method in a child class.
- Overloaded methods must have the same number and type of parameters.
- In overriding, the child method must have the same return type as the parent method
- Both Overloading and overriding use the same method name

## Home Task

### Create a new project in BlueJ.

- Write a Java class Printer with three print() methods: one with no parameters, one with a String, and one with an int.
- Create a parent class Shape with a method display(), and a child class Circle that overrides the display() method.
- Write a Java class Calculator with overloaded multiply() methods: one for two integers, one for three integers, and one for two doubles.
- Create a class Animal with a method sound(). Then create a class Cow that overrides the sound() method to print "Moo Moo".