

# Method Overriding in Java

If subclass (child class) has the same method as declared in the parent class, it is known as **method overriding in Java**.

## Usage of Java Method Overriding

- Method overriding is used to provide the specific implementation of a method which is already provided by its superclass.
- Method overriding is used for runtime polymorphism.

## Rules for Java Method Overriding

1. The method must have the same name as in the parent class
2. The method must have the same parameter as in the parent class.
3. There must be an IS-A relationship (inheritance).

**Let's understand the problem that we may face in the program if we don't use method overriding.**

//Java Program to demonstrate why we need method overriding Here, we are calling the method of parent class with child

```
class Vehicle
{
    void run()
    {
        System.out.println("Vehicle is running");
    }
}

class Bike extends Vehicle
{
    public static void main(String args[])
    {
        Bike obj = new Bike();
    }
}
```

```
obj.run();  
  
}  
  
}
```

## Example of method overriding

//Java Program to illustrate the use of Java Method Overriding Creating a parent class.

```
class Vehicle  
{  
    void run()  
{  
    System.out.println("Vehicle is running");  
}  
}  
  
class Bike2 extends Vehicle  
{  
    void run()  
{  
    System.out.println("Bike is running safely");  
}  
  
    public static void main(String args[])  
{  
        Bike2 obj = new Bike2();  
        obj.run();  
    }  
}
```

## A real example of Java Method Overriding

Consider a scenario where Bank is a class that provides functionality to get the rate of interest. However, the rate of interest varies according to banks. For example, SBI, ICICI and AXIS banks could provide 8%, 7%, and 9% rate of interest.

//Java Program to demonstrate the real scenario of Java Method Overriding

//where three classes are overriding the method of a parent class.

//Creating a parent class.

```
class Bank
{
int getRateOfInterest()
{
return 0;
}
}
.
class SBI extends Bank
{
int getRateOfInterest()
{
return 8;
}
}
class ICICI extends Bank
{
int getRateOfInterest()
{
```

```
return 7;
}
}
class AXIS extends Bank
{
int getRateOfInterest()
{
return 9;
}
}
class Test2
{
public static void main(String args[])
{
SBI s=new SBI();
ICICI i=new ICICI();
AXIS a=new AXIS();
System.out.println("SBI Rate of Interest: "+s.getRateOfInterest());
System.out.println("ICICI Rate of Interest: "+i.getRateOfInterest());
System.out.println("AXIS Rate of Interest: "+a.getRateOfInterest());
}
}
```