Inheritance

Thursday, August 21, 2025 5:24 PM

Inheritance in **Java** is one of the four pillars of Object-Oriented Programming (OOP).

Definition:

Inheritance in Java is the mechanism by which one class (**child/subclass/derived class**) acquires the properties and behaviors (fields and methods) of another class (**parent/superclass/base class**).
It allows **code reusability**, method overriding (runtime polymorphism), and helps in building a hierarchical relationship between classes.

Key Points:

- The keyword **extends** is used for class inheritance.
- The subclass inherits all **non-private fields and methods** of the superclass.
- The subclass can **add new fields/methods** or **override existing methods**.
- Java supports **single inheritance with classes** (a class can extend only one class), but multiple inheritance is possible using **interfaces**.

```java
public class inheritance {  new *
    public static void main(String[] args) {  new *

        // object declaration
        Calc calculator1 = new Calc() ;
        AdvCalc calculator2 = new AdvCalc() ;

        // output
        System.out.println("1 + 2 = " + calculator2.add( a: 1, b: 2));
        System.out.println("1 - 2 = " + calculator2.sub( a: 1, b: 2));
        System.out.println("1 / 2 = " + calculator2.div( a: 1, b: 2));
        System.out.println("1 * 2 = " + calculator2.mult( a: 1, b: 2));

    }
}
```

// output

```
1 + 2 = 3
1 - 2 = -1
1 / 2 = 0
1 * 2 = 2
```

```java
// class 1
class Calc{  3 usages  1 inheritor  new *

    // addition
    public int add(int a , int b){  1 usage  new *
        return a+b ;
    }

    // subtraction
    public int sub(int a , int b){  1 usage  new *
        return a-b ;
    }
}

// class 2 as an inheritance of class 1
class AdvCalc extends Calc{  2 usages  new *

    // division
    public int div(int a , int b){  1 usage  new *
        return a/b ;
    }

    // multiplication
    public int mult(int a , int b){  1 usage  new *
        return a*b ;
    }

}
```