Inheritance

Definition

- The process by which one class (**child/subclass**) acquires the **properties and methods** of another class (**parent/superclass**).
- Promotes **code reusability**.
- Achieved using the `extends` keyword.

Example

```
// Parent class
class Animal {
    void eat() {
        System.out.println("Animals eat food");
    }
}

// Child class
class Dog extends Animal {
    void bark() {
        System.out.println("Dog barks");
    }
}

public class InheritanceDemo {
    public static void main(String[] args) {
        Dog d = new Dog();
        d.eat();   // inherited method
        d.bark();  // own method
    }
}
```

Output:

```
Animals eat food
Dog barks
```

Polymorphism

Definition

Polymorphism means **many forms**. In Java, it allows one thing (method/operation) to behave differently in different contexts.
There are two types:

Compile-time Polymorphism (Method Overloading)

- Same method name with **different parameter lists**.
- Resolved at **compile-time**.

Example

```
class Calculator {
    int add(int a, int b) {
        return a + b;
    }
    double add(double a, double b) {
        return a + b;
    }
}

public class OverloadingDemo {
    public static void main(String[] args) {
        Calculator c = new Calculator();
        System.out.println(c.add(5, 10));       // int version
        System.out.println(c.add(3.5, 2.5));   // double version
    }
```

```
}
```

Output:

```
15
6.0
```

Runtime Polymorphism (Method Overriding)

- A **subclass provides its own implementation** of a method declared in the parent class.
- Achieved through **inheritance + overriding**.
- Resolved at **runtime**.

Example

```
class Animal {
    void sound() {
        System.out.println("Animal makes sound");
    }
}

class Dog extends Animal {
    void sound() {
        System.out.println("Dog barks");
    }
}

class Cat extends Animal {
    void sound() {
        System.out.println("Cat meows");
    }
}

public class OverridingDemo {
    public static void main(String[] args) {
```

```
        Animal a1 = new Dog(); // upcasting
        Animal a2 = new Cat();

        a1.sound();  // Dog's version
        a2.sound();  // Cat's version
    }
}
```

Output:

```
Dog barks
Cat meows
```

Difference Between Inheritance and Polymorphism

Feature	Inheritance	Polymorphism
Meaning	Reusing parent class code in child class.	Same method behaves differently based on context.
Achieved by	`extends` keyword.	Method Overloading & Overriding.
Type	Relationship between classes.	Behavior of methods.
Compile-time/Runtime	Exists in code structure.	Exists as compile-time (overloading) and runtime (overriding).
Example	Dog inherits Animal.	Dog's `sound()` vs Cat's `sound()`.