

Static & Final Methods with Overloading / Overriding Rules

1 Static Methods

- **Definition:** Class-level method, object না তৈরি করেও call করা যায়।
- **Syntax Example:**

```
class Example {  
    static void greet() {  
        System.out.println("Hello!");  
    }  
}  
  
public class Test {  
    public static void main(String[] args) {  
        Example.greet(); // Object creation লাগেনা  
    }  
}
```

- **Key Points:**
- Static method **class-level**, object-level নয়।
- **Override করা যায় না, কিন্তু overload করা যায়।**
- Static method এ `this` keyword use করা যায় না।

2 Final Methods

- **Definition:** Method যা **override করা যাবে না।**
- **Syntax Example:**

```
class Parent {  
    final void show() {  
        System.out.println("Final method");  
    }  
}  
  
class Child extends Parent {  
    // void show() { ... } // ❌ Error! Override not allowed  
}
```

- **Key Points:**
- Final method **override করা যায় না।**

- Final methods **overload** করা যায়।

3 Overloading Rules




- **Definition:** একই class এ method name same, but **different parameters**।
- Static এবং Final methods ****overload** করা যায়।

Example:

```
class Example {
    static void greet() { System.out.println("Hello!"); }
    static void greet(String name) { System.out.println("Hello " + name); }

    final void show() { System.out.println("Show1"); }
    final void show(int x) { System.out.println("Show2: " + x); }
}
```

4 Overriding Rules

Method Type	Overriding Allowed?
Static	 No → method hiding হয়
Final	 No → override করা যায় না
Normal (non-static, non-final)	 Yes

Notes: 1. Static method কে child class এ same signature use করলে **method hiding** হয়। 2. Final method parameter change করলে **overloading** হয়, overriding নয়।

5 Summary

- **Static** → Overload  Override 
- **Final** → Overload  Override 
- **Normal** → Overload  Override 
- Always remember: **Override = same signature + parent → child, Overload = same name + different parameters**

6 Example Usage

```
class Parent {
    final void show() { System.out.println("Final method"); }
    static void greet() { System.out.println("Static method"); }
}
```

```

class Child extends Parent {
    // void show() {} // ❌ Error! Cannot override final method
    static void greet(int x) { System.out.println("Static method overloaded: "
+ x); }
}

public class Test {
    public static void main(String[] args) {
        Parent.greet();           // Static method call
        Child.greet(100);         // Static method overloaded

        Child c = new Child();
        c.show();                 // Final method call
    }
}

```

Output:

```

Static method
Static method overloaded: 100
Final method

```