



Types Of Method With Respect To Inheritance

Table Of Content

[Table Of Content](#)

[Inherited Method](#)

[Overridden Method](#)

[Specialized Method](#)

Inherited Method

- Any method which is **acquired by the child class from the parent class** is called as inherited method.

▼ Inherited Method

```
package _1Java_Codes_From_Basics._15inheritanceInJava._3typesOfFunctionsWithRespec
tToInheritance;

class plane
{
    String name; // instance variable
    void takeOff()
    {
        System.out.println("plane is taking off");
    }
    void fly()
    {
        System.out.println("plane is flying"); // we are inheriting the method fly
    }
    void land()
    {
        System.out.println("plane is landing");
    }
}

class cargoPlane extends plane
{
    void fly()
    {
        super.name = "Cargo Plane";
    }
}
```

```

        System.out.println(super.name+" Flies at 931 km/h"); // inherited method
    }
    void cargoplane()
    {
        System.out.println(super.name+" Is Used Carry Goods");
    }
}

class passengerPlane extends plane
{
    void fly()// inherited method
    {
        super.name = "Passenger Plane";
        System.out.println(super.name+" Flies At 880 - 926 km/h");
    }
    void passengersPlane()
    {
        System.out.println(super.name+" Is Used Carry Peoples");
    }
}
class fighterJet extends plane
{
    void fly()// inherited method
    {
        super.name = "Fighter Jet";
        System.out.println(super.name+" Flies At 7,200 km/h");
    }
    void fighterPlane()
    {
        System.out.println(super.name+" Is Used Carry Arms");
    }
}

public class _1inheritedMethodEx1
{
    public static void main(String[] args)
    {
        //create a object where we use extends
        cargoPlane type1 = new cargoPlane();
        passengerPlane type2 = new passengerPlane();
        fighterJet type3 = new fighterJet();

        type1.takeOff();
        type1.fly();
        type1.cargoplane();
        type1.land();

        System.out.println("-----");
        type1.takeOff();
        type2.fly();
        type2.passengersPlane();
        type1.land();

        System.out.println("-----");
        type1.takeOff();
        type3.fly();
        type3.fighterPlane();
    }
}

```

```

        type1.land();
    }
}
//output
plane is taking off
Cargo Plane Flies at 931 km/h
Cargo Plane Is Used Carry Goods
plane is landing
-----
plane is taking off
Passenger Plane Flies At 880 - 926 km/h
Passenger Plane Is Used Carry Peoples
plane is landing
-----
plane is taking off
Fighter Jet Flies At 7,200 km/h
Fighter Jet Is Used Carry Arms
plane is landing

```

Overridden Method

- **any changes performed on inherited method** is called as overridden methods
- ▼ **Overridden Method**

```

package _1Java_Codes_From_Basics._15inheritanceInJava._3typesOfFunctionsWithRespec
tToInheritance;
class Parent
{
    void marry()
    {
        System.out.println("marry @ age of 26");           // overriding meth
od happening here in child class
    }
}
class Child extends Parent
{
    void marry()
    {
        System.out.println("marry @ age of 30");           // overriding method w.r.t child class
    }
    void job()
    {
        System.out.println("Job is Mandatory");
    }
}
public class _2overridingMethodEx2
{
    public static void main(String[] args)
    {
        Child c = new Child();
        c.marry();
        c.job();
    }
}

```

```

        }
    }
//output
marry @ age of 30
Job is Mandatory

```

Specialized Method

- methods **which are unique to child class** is called as specialized methods

▼ Specialized Method

```

package _1Java_Codes_From_Basics._15inheritanceInJava._3typesOfFunctionsWithRespec
tToInheritance;
class parentClass
{
    void marry()
    {
        System.out.println("marry @ age of 26"); // overriding meth
od happening here in child class
    }
}
class childClass extends parentClass
{
    //overridden method
    void marry()
    {
        System.out.println("marry @ age of 30"); // overriding method w.r.t child class
    }

    //unique method/specialized method in child class
    void job(); // specialized method 1
    void decision(); // specialized method 2
}
public class _3specializedMethodEx3
{
    public static void main(String[] args)
    {
        childClass c = new childClass();
        c.marry();
        c.job();
        c.decision();
    }
}
//output

```

marry @ age of 30

Job is Mandatory

Yes