**Inheritance**:

class Animal{

   void eat(){

       System.out.println("The animal eats food.");

   }

}

class Dog extends Animal{

  void sound(){

      System.out.println("The dog barks.");

  }

}

class Main{

    public static void main(String[] args) {

        Dog obj = new Dog();

        obj.sound();

        obj.eat();

    }

}

**Super Keyword:**

class Vehicle{

  String type;

  Vehicle(String type){

    this.type = type;

  }

  void displayType(){

    System.out.println("Car Type:" + type);

  }

}

class Car extends Vehicle{

  String brand;

  Car(String brand, String type){

    super(type);

    this.brand = brand;

  }

  void displayDetails(){

    super.displayType();

    System.out.println("Car Brand" + brand);

  }

}

class Main{

    public static void main(String[] args) {

      Car car1 = new Car("4x4", "Revo");

      car1.displayDetails();

    }

}

**An example with same method name in parent and child class:**

class Person{

  String name;

  int age;

  Person(String name, int age){

    this.name = name;

    this.age = age;

  }

  void displayInfo(){

    System.out.println(name);

    System.out.println(age);

  }

}

class Student extends Person{

  int id;

  Student(String name, int age, int id){

    super(name,age);

    this.id = id;

  }

  void displayInfo(){

    super.displayInfo();

    System.out.println(id);

  }

}

class Main{

    public static void main(String[] args) {

      Student student1 = new Student("Ali", 21, 101);

      student1.displayInfo();

    }

}

**Single Inheritance:**

//Single Inheritence

class Parent{

    void displayParentMethod(){

        System.out.println("Parent method");

    }

}

class Child extends Parent{

    void displayChildMethod(){

        System.out.println("Child method");

    }

}

class Main{

    public static void main(String[] args) {

     Child obj = new Child();

     obj.displayChildMethod();

     obj.displayParentMethod();

    }

}

**Multilevel Inheritance:**

//MultiLevel Inheritence

class GrandParent{

    void displayGrandParentMethod(){

        System.out.println("Grand Parent method");

    }

}

class Parent extends GrandParent{

    void displayParentMethod(){

        System.out.println("Parent method");

    }

}

class Child extends Parent{

    void displayChildMethod(){

        System.out.println("Child method");

    }

}

class Main{

    public static void main(String[] args) {

     Child obj = new Child();

     obj.displayChildMethod();

     obj.displayParentMethod();

     obj.displayGrandParentMethod();

    }

}

**Hierarchy Inheritance:**

//Hierarchy Inheritence

class Parent{

    void displayParentMethod(){

        System.out.println(" Parent method");

    }

}

class Child1 extends Parent{

    void displayChild1Method(){

        System.out.println("Child1 method");

    }

}

class Child2 extends Parent{

    void displayChild2Method(){

        System.out.println("Child2 method");

    }

}

class Main{

    public static void main(String[] args) {

     Child2 obj = new Child2();

     obj.displayChild2Method();

     obj.displayParentMethod();

     Child1 obj1 = new Child1();

     obj1.displayChild1Method();

     obj1.displayParentMethod();

    }

}