```java
package com.tnsif.oopsconcept;

//Single Inheritance

class Animal{
 String name;


 //Constructor - it initialises the object
 Animal(String name){
  this.name = name;
 }

 void display() {
  System.out.println("Animal Name: "+name);
 }


}

class Dog extends Animal{
 String breed;

 Dog(String name, String breed){
  super(name);
  this.breed = breed;
 }

 void showDetails() {
  display();
  System.out.println("Dog Breed: "+breed);
 }

}
```

```java
//multilevel inheritance

class Grandparent{
 void gp() {
  System.out.println("Grand Parent Class");
 }
}

class Parent extends Grandparent{
 void p() {
  System.out.println("Parent Class");
 }
}

class Child extends Parent{
 void c() {
  System.out.println("Child Class");
 }
}

//multiple inheritance

class Parent1{
 void show() {
  System.out.println("Parent 1");
 }
}

class Parent2{
 void show2() {
  System.out.println("Parent 2");
 }
}

/*
```

```java
 * class child1 extends Parent1, Parent2{ void disp() {
 * System.out.println("Child"); } }
 */



//hierachical Inheritance

class Fruit{
 void eat() {
  System.out.println("I like to eat");
 }
}

class Fruits{
 void eat() {
  System.out.println("I like to eat");
 }
}

class Grapes extends Fruits{
 void eatGrapes() {
  System.out.println("I like to eat grapes");
 }
}

class Oranges extends Fruits{
 void eatOranges() {
  System.out.println("I like to eat oranges");
 }
}



public class InheritanceDemo {
```

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

    Dog d = new Dog("Shero","Street Dog");
    d.showDetails();

    Child c = new Child();
    c.c(); //method calling
    c.p();
    c.gp();

    Grapes g = new Grapes();
    g.eat();
    g.eatGrapes();

    Oranges or = new Oranges();
    or.eat();
    or.eatOranges();

}

}
```