//package com;

// When Same function name perform different task is called polymorphism;

// 1)Method overloading 2)Method Override

//Method overloading: Same function name but differt parameters(either in number of arguments or thier datatypes).

/\*

public class DemoPolymorphism {

int sum(int x,int y)

{

return x+y;

}

int sum(int x,int y,int z)

{

return x+y+z;

}

int sum(String s1,String s2)

{

//return Integer.parseInt(s1+s2);

return Integer.parseInt(s1)+Integer.parseInt(s2);

}

String sum(String s1,String s2,String s3)

{

return s1+s2+s3;

}

public static void main(String[] args) {

DemoPolymorphism ob=new DemoPolymorphism();

System.out.println(ob.sum(5, 6));

System.out.println(ob.sum("4", "5"));

System.out.println(ob.sum(2, 4,5));

System.out.println(ob.sum("Hello","world","java"));

}

}

\*/

//Method Overriding : function has same signature methods in child and super class

class Vehicle

{

String vehicleType;;

void details()

{

System.out.println("Parent class memeber"+vehicleType);

}

}

class Car extends Vehicle

{

String vehicleType;

void details()

{

System.out.println("child class memeber"+vehicleType);

}

}

public class DemoPolymorphism {

public static void main(String[] args) {

Vehicle c=new Car();

c.details();

Vehicle v=new Vehicle();

v.details();

}

}

/\*

abstract class Shape

{

abstract void area();

}

class Circle extends Shape

{

void area()

{

System.out.println();

}

}

\*/