1. **Write a Java program to implement the concept of inheritance.**

```
class employee
{
	private int eid;
	private String name;
	private float sal;
	employee()
	{
		this.eid = 1;
		this.name = "Akash Chatterjee";
		this.sal = 50000;
	}
	void print()
	{
		System.out.println("The id of the employee is ="+this.eid);
		System.out.println("The name of the employee is ="+this.name);
		System.out.println("The salary of the employee is ="+this.sal);
	}
}
class programmer extends employee
{
	private String develops;
	programmer()
	{
		develops = "Python";
		System.out.println("Develops ="+this.develops);
	}
}
public class Week3
{
	public static void main(String[] args)
	{
		programmer ob = new programmer();
		ob.print();
	}
}
```

2. **Write a Java program to show method overloading.**

```
class box
{
	private int length;
	private float breadth;
	box(int length,float breadth)
	{
		this.length = length;
		this.breadth = breadth;
	}
	void area()
	{ System.out.println("the area is ="+((float)this.length*this.breadth));
}
	void area(float len)
	{ System.out.println("the area is ="+((float)len*this.breadth)); }
	void area(int br)
	{ System.out.println("the area is ="+((float)this.length*br)); }
```

```
}
public class Week3
{
	public static void main(String[] args)
	{
		box ob = new box(5,(float)9.0);
		ob.area();
		ob.area((float)2.0);
		ob.area(4);
	}
}
```

3. **Write a Java program to show method overriding.**

```
class employee
{
	private int eid;
	private String name;
	private float sal;
	employee()
	{
		this.eid = 1;
		this.name = "Akash Chatterjee";
		this.sal = 50000;
	}
	void print(String val)
	{
		System.out.println("The method wants to print="+val);
	}
}
class programmer extends employee
{
	private String develops;
	programmer()
	{
		develops = "Python";
	}
	void print(String val) //overridden method
	{
		System.out.println("Develops ="+val);
	}
}
public class Week3
{
	public static void main(String[] args)
	{
		programmer ob = new programmer();
		ob.print("Python"); //this method hides the parent method
	}
}
```

4. **Write a Java program to show method hiding.**

```
class employee
{
	private int eid;
	private String name;
```

```
        private float sal;
        employee()
        {
                this.eid = 1;
                this.name = "Akash Chatterjee";
                this.sal = 50000;
        }
        void print(String val)
        {
                System.out.println("The method wants to print="+val);
        }
}
class programmer extends employee
{
        private String develops;
        programmer()
        {
                develops = "Python";
        }
        void print(String val) //overridden method
        {
                System.out.println("Develops ="+val);
        }
}
public class Week3
{
        public static void main(String[] args)
        {
                programmer ob = new programmer();
                ob.print("Python"); //this method hides the parent method
        }
}
```

5. **Create a general class ThreeDObject and derive the classes Box, Cube, Cylinder and Cone from it. The class ThreeDObject has methods wholeSurfaceArea ( ) and volume ( ). Override these two methods in each of the derived classes to calculate the volume and whole surface area of each type of three-dimensional objects. The dimensions of the objects are to be taken from the users and passed through the respective constructors of each derived class. Write a main method to test these classes.**

```
import java.util.*;
class ThreeDObject
{
        private float dim1;
        private float dim2;
        private float dim3;
        void wholeSurfaceArea()
        {
                System.out.println("The surface area is  =
"+(float)(dim1*dim2+dim2*dim3+dim3*dim1));
        }
        void volume()
        {
                System.out.println("The volume of the object =
"+(float)dim1*dim2*dim3);
        }
}
```

```
class Box extends ThreeDObject{
	Scanner sc = new Scanner (System.in);
	private float dim1;
	private float dim2;
	private float dim3;
	Box(){
		System.out.print("Enter the Dimension1 = ");
		dim1 = sc.nextFloat();
		System.out.print("Enter the Dimension2 = ");
		dim2 = sc.nextFloat();
		System.out.print("Enter the Dimension3 = ");
		dim3 = sc.nextFloat();
	}
	void wholeSurfaceArea(){
System.out.println("The surface area is  =
"+(float)(dim1*dim2+dim2*dim3+dim3*dim1));
	}
	void volume(){
	System.out.println("The volume of the object = "+(float)dim1*dim2*dim3);
	}
}
class Cube extends ThreeDObject{
	Scanner sc = new Scanner (System.in);
	private float dim1;
	private float dim2;
	private float dim3;
	Cube()
	{
		System.out.print("Enter the Dimension = ");
		dim1 = sc.nextFloat();
		dim2 =0;dim3=0;
	}
	void wholeSurfaceArea()
	{
	System.out.println("The surface area is  = "+(float)(3*dim1*dim1));
	}
	void volume()
	{
	System.out.println("The volume of the object = "+(float)dim1*dim1*dim1);
	}
}
class Cylinder extends ThreeDObject
{
	Scanner sc = new Scanner (System.in);
	private float dim1;
	private float dim2;
	private float dim3;
	Cylinder()
	{
		System.out.print("Enter the Radius = ");
		dim1 = sc.nextFloat();
		System.out.print("Enter the Height = ");
		dim2 = sc.nextFloat();
		dim3 = 0;
	}
	void wholeSurfaceArea()
	{
```

```
System.out.println("The surface area is  =
"+(float)(2*3.141f*dim1*dim1+2*3.141f*dim1*dim2));
	}
	void volume(){
System.out.println("The volume of the object = "+(float)3.141*dim1*dim1*dim2);
	}
}
class Cone extends ThreeDObject{
	Scanner sc = new Scanner (System.in);
	private float dim1;
	private float dim2;
	private float dim3;
	Cone()
	{
		System.out.print("Enter the Radius = ");
		dim1 = sc.nextFloat();
		System.out.print("Enter the Height = ");
		dim2 = sc.nextFloat();
		dim3 = 0;
	}
	void wholeSurfaceArea(){
		System.out.println("The surface area is  =
"+(float)(dim1*dim2+dim2*dim3+dim3*dim1));
	}
	void volume()
	{
		System.out.println("The volume of the object =
"+(float)(1/3)*3.141*dim1*dim1*dim2);
	}
}
public class Week3
{
	public static void main(String[] args)
	{
		Box ob = new Box();
		ob.wholeSurfaceArea();
		ob.volume();
		Cylinder ob1 = new Cylinder();
		ob1.wholeSurfaceArea();
		ob1.volume();
		Cube ob2 = new Cube();
		ob2.wholeSurfaceArea();
		ob2.volume();
		Cone ob3 = new Cone();
		ob3.wholeSurfaceArea();
		ob3.volume();
	}
}
```

6. **Write a program to create a class named Vehicle having protected instance variables regnNumber, speed, color, ownerName and a method showData ( ) to show "This is a vehicle class". Inherit the Vehicle class into subclasses named Bus and Car having individual private instance variables routeNumber in Bus and manufacturerName in Car and both of them having showData ( ) method showing all details of Bus and Car respectively with content of the super class's showData ( ) method.**

```
import java.util.*;
```

```
class Vehicle
{
	protected String regnno;
	protected float speed;
	protected String color;
	protected String name;
	void showdata()
	{
		System.out.println("This is a vehicle class");
	}
}
class Bus extends Vehicle
{
	Scanner sc = new Scanner (System.in);
	private String routeno;
	Bus()
	{
		System.out.print("Enter the routeno = ");
		routeno = sc.nextLine();
		System.out.print("Enter the registration no. = ");
		regnno = sc.nextLine();
		System.out.print("Enter the speed of bus = ");
		speed = sc.nextFloat();
		sc.nextLine();
		System.out.print("Enter the color of the bus = ");
		color = sc.nextLine();
		System.out.print("Enter the owner name of the bus = ");
		name = sc.nextLine();
	}
	void showdata()
	{
		super.showdata();
		System.out.println("The route no. of the bus = "+this.routeno);
		System.out.println("The registration no. of the bus = "+this.regnno);
		System.out.println("The Speed of the bus = "+this.speed);
		System.out.println("The color of the bus = "+this.color);
		System.out.println("The owner of the bus = "+this.name);
	}
}
class Car extends Vehicle
{
	Scanner sc = new Scanner (System.in);
	private String manufacturername;
	Car()
	{
		System.out.print("Enter the Manufacturer name = ");
		manufacturername = sc.nextLine();
		System.out.print("Enter the registration no. = ");
		regnno = sc.nextLine();
		System.out.print("Enter the speed of car = ");
		speed = sc.nextFloat();
		sc.nextLine();
		System.out.print("Enter the color of the car = ");
		color = sc.nextLine();
		System.out.print("Enter the owner name of the car = ");
		name = sc.nextLine();
	}
	void showdata()
```

```
	{
		super.showdata();
		System.out.println("The route no. of the car = "+this.manufacturername);
		System.out.println("The registration no. of the car = "+this.regnno);
		System.out.println("The Speed of the car= "+this.speed);
		System.out.println("The color of the car = "+this.color);
		System.out.println("The owner of the car = "+this.name);
	}
}
public class Week3
{
	public static void main(String[] args)
	{
		Car ob = new Car();
		ob.showdata();
		Bus obj = new Bus();
		obj.showdata();
	}
}
```

7. **An educational institution maintains a database of its employees. The database is divided into a number of classes whose hierarchical relationships are shown below. Write all the classes and define the methods to create the database and retrieve individual information as and when needed.**
   Write a driver program to test the classes.

   Staff **(code, name)** — ***Teacher*** **(subject, publication) is a Staff**
   Officer **(grade) is a Staff** — ***Typist*** **(speed) is a Staff**
   RegularTypist **(remuneration) is a Typist** — ***CasualTypist*** **(daily wages) is a Typist.**

```
	import java.util.*;
	class Staff
	{
		protected String code;
		protected String name;
		void showdata()
		{
			System.out.println("This is employee is a staff");
		}
	}
	class Teacher extends Staff
	{
		Scanner sc = new Scanner (System.in);
		private String subject;
		private String publication;
		Teacher()
		{
			System.out.print("Enter the employee code = ");
			code = sc.nextLine();
			System.out.print("Enter the employee name = ");
			name = sc.nextLine();
			System.out.print("Enter the subject he/she teaches = ");
			subject = sc.nextLine();
			System.out.print("Enter the publication of the teacher = ");
			publication = sc.nextLine();
		}
```

```
	void showdata()
	{
		super.showdata();
		System.out.println("The employee code of the teacher =
"+this.code);
		System.out.println("The name of the teacher = "+this.name);
		System.out.println("The Subject of the teacher = "+this.subject);
		System.out.println("The publication of the teacher =
"+this.publication);
	}
}
class Officer extends Staff
{
	Scanner sc = new Scanner (System.in);
	private String grade;
	Officer()
	{
		System.out.print("Enter the employee code = ");
		code = sc.nextLine();
		System.out.print("Enter the employee name = ");
		name = sc.nextLine();
		System.out.print("Enter the grade of the officer = ");
		grade = sc.nextLine();
	}
	void showdata()
	{
		super.showdata();
		System.out.println("The employee code of the officer =
"+this.code);
		System.out.println("The name of the officer = "+this.name);
		System.out.println("The Grade of the officer = "+this.grade);
	}
}
class Typist extends Staff
{
	protected float speed;
	void showdata()
	{
		System.out.println("This is employee of Typist");
	}
}
class RegularTypist extends Typist
{
	Scanner sc = new Scanner (System.in);
	private float remu;
	RegularTypist()
	{
		System.out.print("Enter the employee code = ");
		code = sc.nextLine();
		System.out.print("Enter the employee name = ");
		name = sc.nextLine();
		System.out.print("Enter the typing speed of the Typist = ");
		speed = sc.nextFloat();
		System.out.print("Enter the remuneration of the Typist = ");
		remu = sc.nextFloat();
	}
	void showdata()
	{
```

```
            super.showdata();
            System.out.println("The employee code of the typist =
"+this.code);
            System.out.println("The name of the typist = "+this.name);
            System.out.println("The typing speed of the typist =
"+this.speed);
            System.out.println("The remunerisation of the typist =
"+this.remu);
      }
}
class CasualTypist extends Typist
{
      Scanner sc = new Scanner (System.in);
      private float dailywage;
      CasualTypist()
      {
            System.out.print("Enter the employee code = ");
            code = sc.nextLine();
            System.out.print("Enter the employee name = ");
            name = sc.nextLine();
            System.out.print("Enter the typing speed of the Typist = ");
            speed = sc.nextFloat();
            System.out.print("Enter the daily wage of the Typist = ");
            dailywage = sc.nextFloat();
      }
      void showdata()
      {
            super.showdata();
            System.out.println("The employee code of the typist =
"+this.code);
            System.out.println("The name of the typist = "+this.name);
            System.out.println("The typing speed of the typist =
"+this.speed);
            System.out.println("The daily wage of the typist =
"+this.dailywage);
      }
}
public class Week3
{
      public static void main(String[] args)
      {
            Teacher ob = new Teacher();
            ob.showdata();
      }
}
```

8. **Create a base class Building that stores the number of floors of a building, number of rooms and it's total footage. Create a derived class House that inherits Building and also stores the number of bedrooms and bathrooms. Demonstrate the working of the classes.**

```
import java.util.*;
class Building
{
      protected int floors;
      protected int rooms;
      protected int footage;
}
```

```
class House extends Building
{
	private int bedrooms,bathrooms;
	Scanner sc = new Scanner(System.in);
	House()
	{
		System.out.println("Enetr no. of the floors =");
		this.floors = sc.nextInt();
		System.out.println("No. of the rooms =");
		this.rooms = sc.nextInt();
		System.out.println("No. of CCTV footage cameras =");
		this.footage = sc.nextInt();
		System.out.println("No. of the Bathrooms =");
		this.bathrooms = sc.nextInt();
		System.out.println("No. of the Bedrooms =");
		this.bedrooms = sc.nextInt();
	}
	void print()
	{
		System.out.println("No. of the floors ="+this.floors);
		System.out.println("No. of the rooms ="+this.rooms);
		System.out.println("No. of CCTV footage cameras ="+this.footage);
		System.out.println("No. of the Bathrooms ="+this.bathrooms);
		System.out.println("No. of the Bedrooms ="+this.bedrooms);
	}

}
public class week3
{
	public static void main(String[] args)
	{
		House ob = new House();
		ob.print();
	}
}
```

9. In the earlier program, create a second derived class Office that inherits Building and stores the number of telephones and tables. Now demonstrate the working of all three classes.

```
import java.util.*;
class Building
{
	protected int floors;
	protected int rooms;
	protected int footage;
	void print()
	{
		System.out.println("No. of the floors ="+this.floors);
		System.out.println("No. of the rooms ="+this.rooms);
		System.out.println("No. of CCTV footage cameras ="+this.footage);
	}
}
class House extends Building
{
	private int bedrooms,bathrooms;
```

```
        Scanner sc = new Scanner(System.in);
        House()
        {
              System.out.println("Enetr no. of the floors =");
              this.floors = sc.nextInt();
              System.out.println("No. of the rooms =");
              this.rooms = sc.nextInt();
              System.out.println("No. of CCTV footage cameras =");
              this.footage = sc.nextInt();
              System.out.println("No. of the Bathrooms =");
              this.bathrooms = sc.nextInt();
              System.out.println("No. of the Bedrooms =");
              this.bedrooms = sc.nextInt();
        }
        void print()
        {
              System.out.println("No. of the floors ="+this.floors);
              System.out.println("No. of the rooms ="+this.rooms);
              System.out.println("No. of CCTV footage cameras ="+this.footage);
              System.out.println("No. of the Bathrooms ="+this.bathrooms);
              System.out.println("No. of the Bedrooms ="+this.bedrooms);
        }

  }
  class Office extends Building
  {
        private int tables,telephones;
        Scanner sc = new Scanner(System.in);
        Office()
        {
              System.out.println("Enetr no. of the floors =");
              this.floors = sc.nextInt();
              System.out.println("No. of the rooms =");
              this.rooms = sc.nextInt();
              System.out.println("No. of CCTV footage cameras =");
              this.footage = sc.nextInt();
              System.out.println("No. of the Tables =");
              this.tables = sc.nextInt();
              System.out.println("No. of the Telephones =");
              this.telephones = sc.nextInt();
        }
        void print()
        {
              System.out.println("No. of the floors ="+this.floors);
              System.out.println("No. of the rooms ="+this.rooms);
              System.out.println("No. of CCTV footage cameras ="+this.footage);
              System.out.println("No. of the Tables ="+this.tables);
              System.out.println("No. of the Telephones ="+this.telephones);
        }

  }
  public class week3
  {
        public static void main(String[] args)
        {
              Building ob1 = new Building();
              ob1.print();
              House ob = new House();
```

```
            ob.print();
            Office ob2 = new Office();
            ob2.print();
      }
}
```

10. Write a Java program which creates a base class Num and contains an integer number along with a method shownum() which displays the number. Now create a derived class HexNum which inherits Num and overrides shownum() which displays the hexadecimal value of the number. Demonstrate the working of the classes.

```
import java.util.*;
import java.lang.*;
class Num
{
      protected int num;
      void shownum()
      {
            System.out.println("The number is ="+this.num);
      }
}
class Hexnum extends Num
{
      Scanner sc = new Scanner(System.in);
      Hexnum()
      {
            System.out.println("Eneter the number =");
            this.num = sc.nextInt();
      }
      void Hexcon()
      {
            int n=this.num,k=0;
            StringBuffer s = new StringBuffer();
            while(n!=0)
            {
                  if(n%16==10)
                  s.append('A');
                  else if (n%16==11)
                  s.append('B');
                  else if (n%16==12)
                  s.append('C');
                  else if (n%16==13)
                  s.append('D');
                  else if (n%16==14)
                  s.append('E');
                  else if (n%16==15)
                  s.append('F');
                  else
                  s.append(n%16);
                  n =(int) n/16;
            }
            System.out.println("The given no. in hexadecimal ="+s.reverse());
      }

}
public class week3
{
```

```
	public static void main(String[] args)
	{
		Num ob = new Num();
		ob.shownum();
		Hexnum ob1 = new Hexnum();
		ob1.Hexcon();
	}
}
```

11. Write a Java program which creates a base class Num and contains an integer number along with a method shownum() which displays the number. Now create a derived class OctNum which inherits Num and overrides shownum() which displays the octal value of the number. Demonstrate the working of the classes.

```
import java.util.*;
import java.lang.*;
class Num
{
	protected int num;
	void shownum()
	{
		System.out.println("The number is ="+this.num);
	}
}
class Octnum extends Num
{
	Scanner sc = new Scanner(System.in);
	Octnum()
	{
		System.out.println("Eneter the number =");
		this.num = sc.nextInt();
	}
	void Octcon()
	{
		int n=this.num;
		StringBuffer s = new StringBuffer();
		while(n!=0)
		{
			s.append(n%8);
			n =(int) n/8;
		}
		System.out.println("The given no. in Octal ="+s.reverse());
	}

}
public class week3
{
	public static void main(String[] args)
	{
		Num ob = new Num();
		ob.shownum();
		Octnum ob1 = new Octnum();
		ob1.Octcon();
	}
}
```

12. Combine Question number 10 and 11 and have all the three classes together. Now describe the working of all classes.

```
import java.util.*;
import java.lang.*;
class Num
{
	protected int num;
	void shownum()
	{
		System.out.println("The number is ="+this.num);
	}
}
class Hexnum extends Num
{
	Scanner sc = new Scanner(System.in);
	Hexnum()
	{
		System.out.println("Eneter the number =");
		this.num = sc.nextInt();
	}
	void Hexcon()
	{
		int n=this.num,k=0;
		StringBuffer s = new StringBuffer();
		while(n!=0)
		{
			if(n%16==10)
			s.append('A');
			else if (n%16==11)
			s.append('B');
			else if (n%16==12)
			s.append('C');
			else if (n%16==13)
			s.append('D');
			else if (n%16==14)
			s.append('E');
			else if (n%16==15)
			s.append('F');
			else
			s.append(n%16);
			n =(int) n/16;
		}
		System.out.println("The given no. in hexadecimal ="+s.reverse());
	}

}
class Octnum extends Num
{
	Scanner sc = new Scanner(System.in);
	Octnum()
	{
		System.out.println("Eneter the number =");
		this.num = sc.nextInt();
	}
	void Octcon()
	{
```

```
                int n=this.num;
                StringBuffer s = new StringBuffer();
                while(n!=0)
                {
                      s.append(n%8);
                      n =(int) n/8;
                }
                System.out.println("The given no. in Octal ="+s.reverse());
        }

}
public class week3
{
        public static void main(String[] args)
        {
                Num ob = new Num();
                ob.shownum();
                Octnum ob1 = new Octnum();
                ob1.Octcon();
                Hexnum ob2 = new Hexnum();
                Ob2.Hexcon();
        }
}
```

13. Create a base class Distance which stores the distance between two locations in miles and a method travelTime(). The method prints the time taken to cover the distance when the speed is 60 miles per hour. Now in a derived class DistanceMKS, override travelTime() so that it prints the time assuming the distance is in kilometers and the speed is 100 km per second. Demonstrate the working of the classes.

```
import java.util.*;
class Distance
{
        protected float loc1;
        Scanner sc = new Scanner(System.in);
        Distance()
        {
        System.out.println("Enter the distance between 2 location in Miles =");
                this.loc1 = sc.nextFloat();
        }
        void traveltime()
        {
                System.out.println("The travel time is ="+(this.loc1/60)+"hr");
        }
}
class DistanceMKS extends Distance
{
        private String develops;
        Scanner sc = new Scanner(System.in);
        DistanceMKS()
        {
        super.traveltime();
        System.out.println("Enter the distance between 2 location in Km =");
        this.loc1 = sc.nextFloat();
        }
        void traveltime()
```

```
        {
            System.out.println("The travel time is ="+(this.loc1/100)+"sec");
        }
    }
    public class week3
    {
        public static void main(String[] args)
        {
            DistanceMKS ob1 = new DistanceMKS();
            ob1.traveltime();
        }
    }
```

14. **Create a base class called "vehicle" that stores number of wheels and speed. Create the following derived classes – "car" that inherits "vehicle" and also stores number of passengers. "truck" that inherits "vehicle" and also stores the load limit. Write a main function to create objects of these two derived classes and display all the information about "car" and "truck". Also compare the speed of these two vehicles - car and truck and display which one is faster.**

```
import java.util.*;
class Vehicle
{
    protected int wheels;
    protected float speed;
    void showdata()
    {
        System.out.println("This is a vehicle class");
    }
}
class Truck extends Vehicle
{
    Scanner sc = new Scanner (System.in);
    private int load;
    float getspeedtr()
    { return this.speed; }
    Truck()
    {
        System.out.println("Enter the no. of wheels of the truck = ");
        wheels = sc.nextInt();
        System.out.println("Enter the speed of truck = ");
        speed = sc.nextFloat();
        System.out.println("Enter the load limit of the truck = ");
        load = sc.nextInt();
    }
    void showdata(){
        super.showdata();
        System.out.print("No. of wheels of the truck = "+this.wheels);
        System.out.print("The speed of truck = "+this.speed);
        System.out.print("The load limit of the truck = "+this.load);
    }
}
class Car extends Vehicle
{
```

```
	Scanner sc = new Scanner (System.in);
	private int passengers;
	Car()
	{
		System.out.println("Enter the no. of wheels of the car = ");
		wheels = sc.nextInt();
		System.out.println("Enter the speed of car = ");
		speed = sc.nextFloat();
		System.out.println("Enter the no. of seats of the car = ");
		passengers = sc.nextInt();
	}
	float getspeedc()
	{ return this.speed; }
	void showdata()
	{
		super.showdata();
		System.out.println("No. of wheels of the car = "+this.wheels);
		System.out.println("The speed of car = "+this.speed);
		System.out.println("The no. of seats of the car = "+this.passengers);
	}
}
public class week3
{
	public static void main(String[] args)
	{
		Car ob = new Car();
		ob.showdata();
		Truck obj = new Truck();
		obj.showdata();
		if(ob.getspeedc()<obj.getspeedtr())
		System.out.println("The truck is faster than car");
		else
		System.out.println("The car is faster than the truck");
	}
}
```

15. Write a Java program to explain "multilevel inheritance."

```
	import java.util.*;
	class Staff                                  //Level 0
	{
		protected String code;
		protected String name;
		void showdata()
		{
			System.out.println("This is employee is a staff");
		}
	}
	class Typist extends Staff                          //Level 1
	{
		protected float speed;
		void showdata()
		{
			System.out.println("This is employee of Typist");
		}
	}
	class RegularTypist extends Typist          //level 2
```

```
{
	Scanner sc = new Scanner (System.in);
	private float remu;
	RegularTypist()
	{
		System.out.print("Enter the employee code = ");
		code = sc.nextLine();
		System.out.print("Enter the employee name = ");
		name = sc.nextLine();
		System.out.print("Enter the typing speed of the Typist = ");
		speed = sc.nextFloat();
		System.out.print("Enter the remuneration of the Typist = ");
		remu = sc.nextFloat();
	}
	void showdata()
	{
		super.showdata();
		System.out.println("The employee code of the typist =
"+this.code);
		System.out.println("The name of the typist = "+this.name);
		System.out.println("The typing speed of the typist =
"+this.speed);
		System.out.println("The remunerisation of the typist =
"+this.remu);
	}
}
public class Week3
{
	public static void main(String[] args)
	{
		RegularTypist ob = new RegularTypist();
		ob.showdata();
	}
}
```