

Name	Hatim Yusuf Sawai
UID no.	2021300108
Experiment No.	6

AIM:	Programs to demonstrate inheritance and method overriding
<b>Program 1</b>	
<b>PROBLEM STATEMENT:</b>	<p>Consider a scenario where Bank is a class that provides functionality to get the rate of interest. However, the rate of interest varies according to banks. For example, SBI, ICICI and AXIS banks are given below.</p> <p>Aayush has deposited Rs. 10000 in SBI Bank, Rs. 12500 in ICICI Bank, and Rs. 20000 in AXIS bank respectively for a particular month. You need to print the money he will get by applying the rate of interest as per the bank and days.</p> <p>Create a class 'Bank' with a method 'get_rate_of_interest' which returns 2%.</p> <p>Make three subclasses named SBI_Bank, 'ICICI_Bank' and 'AXIS_bank' with a method with the same name 'get_rate_of_interest' which returns the rate of interest.</p> <p>Also, give the final amount Ayush will get from that particular bank by applying the rate of interest and period. Use Calendar Class to count the number of days and amount he will get after maturity with the date of Maturity, if he deposits today.</p> <p>Note:</p> <ol style="list-style-type: none"> <li>1. Use compound interest</li> <li>2. Get time period from the user</li> <li>3. Solve using method overriding</li> </ol>

**PROGRAM:**

```
import java.util.Scanner;
import java.util.Calendar;
import java.text.*;
import java.lang.Math;
class Bank {
    double time;
    int p;
    double amt=0;;
    double rate=0;
    Bank(double time,int p) {
        this.time = time;
        this.p = p;
    }
    void get_rate_of_interest() {
        rate = 2;
    }
}
class SBI_bank extends Bank {
    SBI_bank(double time,int p) {
        super(time, p);
    }
    void get_rate_of_interest() {
        if(time>=7 && time<=14) {rate = 3;}
        else if(time>=15 && time<=30) {rate = 3;}
        else if(time>=31 && time<=45) {rate = 3;}
        else if(time>=46 && time<=90) {rate = 4.05;}
        else if(time>=91 && time<=120) {rate = 4.10;}
        else if(time<=121 && time>=180) {rate = 4.10;}
    }
    void display() {
        amt = (1+rate/100);
        amt = Math.pow(amt, time/365);
        amt = p*amt;
        System.out.printf("Total amount on Maturity: $%.2f\n",amt);
    }
}
```

```

class AXIS_Bank extends Bank {
    AXIS_Bank(double time,int p) {
        super(time, p);
    }
    void get_rate_of_interest() {
        if(time>=7 && time<=14) {rate = 3.15;}
        else if(time>=15 && time<=30) {rate = 3.15;}
        else if(time>=31 && time<=45) {rate = 3.45;}
        else if(time>=46 && time<=90) {rate = 4.05;}
        else if(time>=91 && time<=120) {rate = 4.70;}
        else if(time<=121 && time>=180) {rate = 5;}
    }
    void display() {
        amt = (1+rate/100);
        amt = Math.pow(amt, time/365);
        amt = p*amt;
        System.out.printf("Total amount on Maturity: $%.2f\n",amt);
    }
}

class ICICI_Bank extends Bank {
    ICICI_Bank(double time,int p) {
        super(time, p);
    }
    void get_rate_of_interest() {
        if(time>=7 && time<=14) {rate = 3.10;}
        else if(time>=15 && time<=30) {rate = 3.20;}
        else if(time>=31 && time<=45) {rate = 3.50;}
        else if(time>=46 && time<=90) {rate = 4.50;}
        else if(time>=91 && time<=120) {rate = 4.70;}
        else if(time<=121 && time>=180) {rate = 4.90;}
    }
    void display() {
        amt = (1+rate/100);
        amt = Math.pow(amt, time/365);
        amt = p*amt;
        System.out.printf("Total amount on Maturity: $%.2f\n",amt);
    }
}

```

```

    }
}
public class finance {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Calendar cal = Calendar.getInstance();
        DateFormat date = DateFormat.getDateInstance();
        int choice, flag, p;
        double time;
        while(true) {
            System.out.print("Enter time period(days): ");
            while(true) {
                time = sc.nextDouble();
                if(time<=120) {break;}
                else {System.out.print("Max Time limit is: 120 Days! Please
Re-Enter time period: ");}
            }
            System.out.print("Enter the Amount: ");
            p = sc.nextInt();
            cal.add(Calendar.DATE,(int)time);
            System.out.println("Select 1 Bank:\n1 -> SBI Bank\n2 -> ICICI
Bank\n3 -> AXIS Bank");
            choice = sc.nextInt();
            switch(choice) {
                case 1:
                    SBI_bank b1 = new SBI_bank(time,p);
                    b1.get_rate_of_interest();
                    b1.display();
                    System.out.println("Your Amount will mature on:
"+date.format(cal.getTime()));
                    break;
                case 2:
                    ICICI_Bank b2 = new ICICI_Bank(time,p);
                    b2.get_rate_of_interest();
                    b2.display();
                    System.out.println("Your Amount will mature on:

```

```

"+date.format(cal.getTime()));
        break;
    case 3:
        AXIS_Bank b3 = new AXIS_Bank(time,p);
        b3.get_rate_of_interest();
        b3.display();
        System.out.println("Your Amount will mature on:
"+date.format(cal.getTime()));
        break;
    default:
        System.out.println("Invalid Bank!");
        break;
    }
    System.out.println("Do you want to continue?(yes=1/0=no)");
    flag = sc.nextInt();
    if(flag==0){
        break;
    }
}
sc.close();
}
}

```

## RESULT:

```

PS D:\Java Practicals\Experiment_6> cd "d:\Java Practicals\Expe
Enter time period(days): 85
Enter the Amount: 53000
Select 1 Bank:
1 -> SBI Bank
2 -> ICICI Bank
3 -> AXIS Bank
2
Total amount on Maturity: $53546.07
Your Amount will mature on: 20-Sep-2022
Do you want to continue?(yes=1/0=no)
1
Enter time period(days): 130
Max Time limit is: 120 Days! Please Re-Enter time period: 120
Enter the Amount: 69000
Select 1 Bank:
1 -> SBI Bank
2 -> ICICI Bank
3 -> AXIS Bank
3
Total amount on Maturity: $70049.80
Your Amount will mature on: 18-Jan-2023
Do you want to continue?(yes=1/0=no)
0
PS D:\Java Practicals\Experiment_6>

```

## Program 2

### PROBLEM STATEMENT:

Ankit works at ABC Company. He noticed that different roles(positions) have different salaries and bonuses.

The 1st Role is an 'Intern' which has  $\frac{3}{4}$ th of the base salary of an Employee.

Then there is 'Clerk' which has  $\frac{1}{2}$  of base salary.

And then there are 'Manager' who have twice the base salary of that of an employee.

Help him write a program in Java as follows.

Create a class 'Employee' which has a method named 'getSalary' which returns a base salary of Rs. 10,000. It also has methods named 'getBonus' which returns the bonus amount for that role(initially set to Rs. 0).

Make 3 subclasses for different roles which inherit the 'Employee' class and each has functions named 'getSalary' and 'getBonus'.(You can assume values for 'getBonus' method)

Display the output for all cases. Also print the total salary received by each Employee after getting the bonus.

Note : Solve using method overriding

### PROGRAM:

```
import java.util.*;
class Employee {
    double salary,bonus;
    Employee(double salary) {
        this.salary = salary;
    }
    void getSalary() {
        salary = 10000;
    }
    void getBonus() {
        bonus = 0;
    }
}
```

```

    }
    class Intern extends Employee {
        Intern(double salary) {
            super(salary);
        }
        void getSalary() {
            salary = salary*0.75;
        }
        void getBonus() {
            bonus = 690;
        }
    }
    class Clerk extends Employee {
        Clerk(double salary) {
            super(salary);
        }
        void getSalary() {
            salary = salary/2;
        }
        void getBonus() {
            bonus = 720;
        }
    }
    class Manager extends Employee {
        Manager(double salary) {
            super(salary);
        }
        void getSalary() {
            salary = 2*salary;
        }
        void getBonus() {
            bonus = 1000;
        }
    }
    public class ABC {
        public static void main(String[] args) {

```

```

Scanner sc = new Scanner(System.in);
System.out.print("Enter Base Salary: ");
double salary = sc.nextDouble();
Intern in = new Intern(salary);
in.getSalary();
in.getBonus();
System.out.printf("Salary of Intern: $%.2f",in.salary+in.bonus);
Clerk cr = new Clerk(salary);
cr.getSalary();
cr.getBonus();
System.out.printf("\nSalary of Clerk: $%.2f",cr.salary+cr.bonus);
Manager mg = new Manager(salary);
mg.getSalary();
mg.getBonus();
System.out.printf("\nSalary of Employee:
$%.2f",mg.salary+mg.bonus);
sc.close();
}
}

```

#### RESULT:

```

PS D:\Java Practicals\Experiment_6> cd
Enter Base Salary: 55000
Salary of Intern: $41940.00
Salary of Clerk: $28220.00
Salary of Employee: $111000.00
PS D:\Java Practicals\Experiment_6> █

```

### Program 3

#### PROBLEM STATEMENT:

Create a class named 'Shape' which has a method 'getArea', 'getPerimeter' and 'getSide' and all of them return 0. Make three subclasses for three different shapes - 'Circle', 'Triangle' and 'Pentagon'. These subclasses inherit the 'Shape' class and they also have 'getArea', 'getPerimeter' and 'getSide' methods.

Write a program for the above scenario and display the solution.

Note: Solve using method overriding



**PROGRAM:**

```
import java.util.*;
import java.lang.Math;
class Shape {
    Scanner sc = new Scanner(System.in);
    double area,perimeter;
    double radius,a,b,c,side;
    void getArea() {
        area = 0;
    }
    void getPerimeter() {
        perimeter = 0;
    }
    void getSide() {
        side = 0;
    }
}
class Circle extends Shape {
    void getArea() {
        area = Math.PI*radius*radius;
    }
    void getPerimeter() {
        perimeter = 2*Math.PI*radius;
    }
    void getSide() {
        System.out.println("Enter the radius of the circle: ");
        radius = sc.nextDouble();
    }
}
class Triangle extends Shape {
    void getArea() {
        double s = (a+b+c)/2;
        area = s*(s-a)*(s-b)*(s-c);
        area = Math.sqrt(area);
    }
    void getPerimeter() {
        perimeter = a+b+c;
    }
}
```

```

    }
    void getSide() {
        System.out.println("Enter 3 sides of the triangle: ");
        a = sc.nextDouble();
        b = sc.nextDouble();
        c = sc.nextDouble();
    }
}

class Pentagon extends Shape{
    void getArea() {
        area = 5*(5+2*Math.sqrt(5));
        area = Math.sqrt(area);
        area = 0.25*area*side*side;
    }
    void getPerimeter() {
        perimeter = 5*side;
    }
    void getSide() {
        System.out.println("Enter side of the pentagon: ");
        side = sc.nextDouble();
    }
}

public class TestShape {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int choice, flag;
        while(true) {
            System.out.println("Select 1 Shape:\n1 -> Circle\n2 ->
Triangle\n3 -> Pentagon");
            choice = sc.nextInt();
            switch(choice) {
                case 1:
                    Circle c = new Circle();
                    c.getSide();
                    c.getPerimeter();
                    c.getArea();

```

```

        System.out.printf("Perimeter of Circle: %.2f\nArea of
circle: %.2f",c.perimeter,c.area);
        break;
    case 2:
        Triangle t = new Triangle();
        t.getSide();
        t.getPerimeter();
        t.getArea();
        System.out.printf("Perimeter of Triangle: %.2f\nArea of
Triangle: %.2f",t.perimeter,t.area);
        break;
    case 3:
        Pentagon p = new Pentagon();
        p.getSide();
        p.getPerimeter();
        p.getArea();
        System.out.printf("Perimeter of Pentagon: %.2f\nArea of
Pentagon: %.2f",p.perimeter,p.area);
        break;
    default:
        System.out.println("Invalid Shape!");
        break;
    }
    System.out.println("\nDo you want to continue?(yes=1/0=no)");
    flag = sc.nextInt();
    if(flag==0){
        break;
    }
}
sc.close();
}
}

```

**RESULT:**

```
PS D:\Java Practicals\Experiment_6> cd
e }
Select 1 Shape:
1 -> Circle
2 -> Triangle
3 -> Pentagon
1
Enter the radius of the circle:
25.7
Perimeter of Circle: 161.48
Area of circle: 2074.99
Do you want to continue?(yes=1/no=0)
1
Select 1 Shape:
1 -> Circle
2 -> Triangle
3 -> Pentagon
3
Enter side of the pentagon:
6.9
Perimeter of Pentagon: 34.50
Area of Pentagon: 81.91
Do you want to continue?(yes=1/no=0)
0
PS D:\Java Practicals\Experiment_6> |
```

**Program 4****PROBLEM  
STATEMENT:**

Imagine a publishing company that markets both book and audiocassette versions of its works. Create a class publication that stores the title (a string) and price (type float) of a publication. From this class derive two classes: book, which adds a page count (type int), and tape, which adds a playing time in minutes (type float). Each of these three classes should have a `getdata()` function to get its data from the user at the keyboard, and a `putdata()` function to display its data.

Add a class sales that holds an array of three floats so that it can record the dollar sales of a particular publication for the last three months. Include a `getdata()` function to get three sales amounts from the user, and a `putdata()` function to display the sales figures. Alter the book class so they are derived from both publication and sales. An object of class book should input and output sales data along with its

	other data. Write a main() function to create a book object and a tape object and exercise their input/output capabilities.
<b>PROGRAM:</b>	<pre> import java.util.*; class Publication {     Scanner sc = new Scanner(System.in);     String title;     float price;     void getData() {         System.out.print("Enter the title: ");         title = sc.nextLine();         System.out.print("Enter the price: ");         price = sc.nextInt();     }     void putData() {         System.out.printf("\nDetails:\nTitle: %s\nPrice: \$%.2f",title,price);     } } class Book extends Publication {     int pg_count;     void getData() {         super.getData();         System.out.print("Enter no. of pages: ");         pg_count = sc.nextInt();     }     void putData() {         super.putData();         System.out.printf("\nNo. of Pages: %d",pg_count);     } } class Tape extends Publication {     float mins;     void getData() {         System.out.println("\nTape:");         super.getData(); </pre>

```

        System.out.print("Enter time(in Mins): ");
        mins = sc.nextFloat();
    }
    void putData() {
        super.putData();
        System.out.printf("Time: %.1f mins",mins);
    }
}
class Sales extends Book {
    Scanner sc = new Scanner(System.in);
    int[] sale = new int[3];
    void getData() {
        System.out.println("Book:");
        super.getData();
        for(int i=0;i<3;i++) {
            System.out.printf("Enter Sales for Month %d: ",i+1);
            sale[i] = sc.nextInt();
        }
    }
    void putData() {
        super.putData();
        System.out.printf("\nSales:\nMonth 1: %d\nMonth 2: %d\nMonth
3: %d",sale[0],sale[1],sale[2]);
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Sales s = new Sales();
        Tape t = new Tape();
        s.getData();
        t.getData();
        s.putData();
        t.putData();
        sc.close();
    }
}

```

<b>RESULT:</b>	<pre> PS D:\Java Practicals\Experiment_6&gt; cd Book: Enter the title: Harry Potter Enter the price: 600 Enter no. of pages: 472 Enter Sales for Month 1: 6900 Enter Sales for Month 2: 7400 Enter Sales for Month 3: 4500  Tape: Enter the title: ek do teen Enter the price: 350 Enter time(in Mins): 3.45 Details: Title: Harry Potter Price: \$600.00 No. of Pages: 472 Sales: Month 1: 6900 Month 2: 7400 Month 3: 4500 Details: Title: ek do teen Price: \$350.00Time: 3.5 mins PS D:\Java Practicals\Experiment_6&gt; </pre>
<b>CONCLUSION:</b>	<p>In this experiment, we learned how to implement complex inheritance problems and use overridden methods to save memory and compile-time in our programs.</p>