**Java Lab Sheet 3**

**-Mohith LS  
AM.SC.U3CSC21036**

1. Create a class to print the area of square and rectangle with the same name but different parameters.

import java.util.Scanner;

public class L3Q1

{

    public static void main(String args[])

    {

        Scanner sc = new Scanner(System.in);

        Shape shape = new Shape();

        System.out.print("Enter side of square: ");

        System.out.println("Area of square: "+shape.area(sc.nextInt()));

        System.out.print("Enter length and breadth: ");

        System.out.println("Area of rectangle: "+shape.area(sc.nextInt(), sc.nextInt()));

    }

}

class Shape

{

    public int area(int a)

    {

        return a\*a;

    }

    public int area(int a, int b)

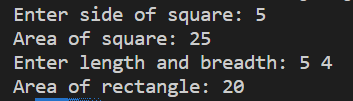
    {

        return a\*b;

    }

}

Output:



2. Create a class student with three data members... (Use array of 10 student objects)

import java.util.Scanner;

public class L3Q2

{

    public static void main(String args[])

    {

        Scanner sc = new Scanner(System.in);

        Student sarray[] = new Student[10];

        int choice=0;

        String temp\_name;

        int temp\_age;

        String temp\_addr;

        for (int j=0;j<10;j++)

        {

            sarray[j] = new Student();

        }

        while (choice!=4)

        {

            System.out.println("1.Set info (without address)\n2.Set info (with address)\n3.Print info\n4.Exit");

            System.out.print("Enter your choice: ");

            choice = sc.nextInt();

            sc.nextLine();

            for (int i=0;i<10;i++)

            {

                switch(choice)

                {

                    case 1:

                        System.out.print("Enter name: ");

                        temp\_name = sc.nextLine();

                        System.out.print("Enter age: ");

                        temp\_age = sc.nextInt();

                        sc.nextLine();

                        sarray[i].setInfo(temp\_name,temp\_age);

                        break;

                    case 2:

                        System.out.print("Enter name: ");

                        temp\_name = sc.nextLine();

                        System.out.print("Enter age: ");

                        temp\_age = sc.nextInt();

                        sc.nextLine();

                        System.out.print("Enter address: ");

                        temp\_addr = sc.nextLine();

                        sarray[i].setInfo(temp\_name,temp\_age,temp\_addr);

                        break;

                    case 3:

                        sarray[i].printInfo();

                        break;

                    default:

                        break;

                }

            }

        }

    }

}

class Student

{

    public String name;

    public int age;

    public String address;

    Student()

    {

        name = new String("unknown");

        age = 0;

        address = new String("not available");

    }

    public void setInfo(String name, int age)

    {

        this.name = name;

        this.age = age;

    }

    public void setInfo(String name, int age, String address)

    {

        this.name = name;

        this.age = age;

        this.address = address;

    }

    public void printInfo()

    {

        System.out.println("Name: "+name);

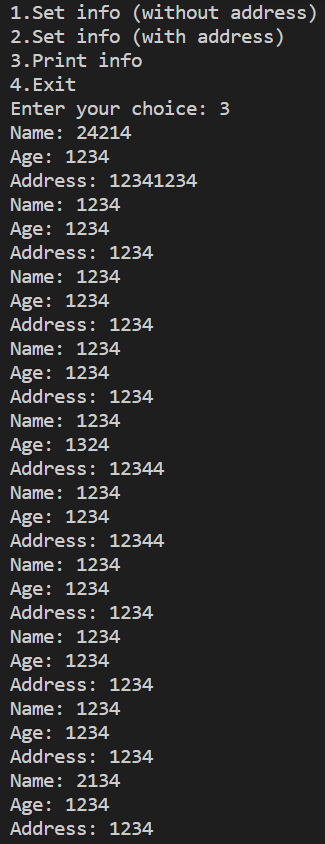
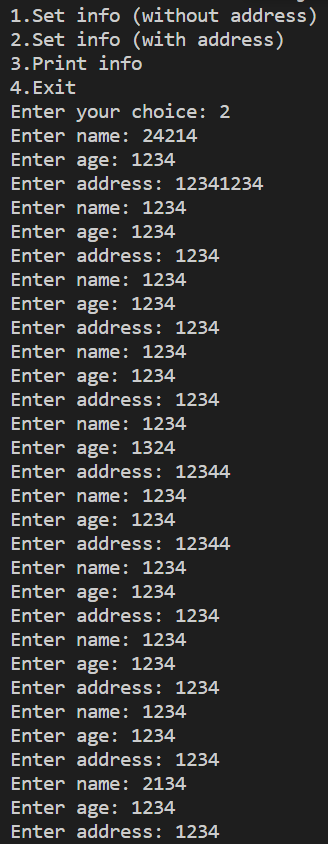
        System.out.println("Age: "+age);

        System.out.println("Address: "+address);

    }

}

Output:



3. Assume that a bank maintains two kinds of accounts. Create a class account and two sub classes from it.

import java.util.Scanner;

class Account

{

    String cust\_name;

    long acc\_no;

    String acc\_type;

    int balance=0;

    public void credit(int amt)

    {

        balance+=amt;

    }

}

class CurrentAccount extends Account

{

    public void deposit(int amt)

    {

        credit(amt);

    }

    public void penalty()

    {

        if(balance<1000)

        {

            balance-=100;

        }

    }

    public void checkMinBalance()

    {

        if(balance<1000)

        {

            System.out.println("Minimum balance not maintained");

            penalty();

        }

        else

        {

            System.out.println("Minimum balance maintained");

        }

    }

    public int withdraw(int amt)

    {

        if(amt>balance)

        {

            System.out.println("Insufficient balance");

            return 0;

        }

        else

        {

            balance-=amt;

            return amt;

        }

    }

    public void checkBalance()

    {

        System.out.println("Balance: "+balance);

    }

}

class SavingsAccount extends Account

{

    public void deposit(int amt)

    {

        credit(amt);

    }

    public void addInterest()

    {

        credit(balance\*5/100);

    }

    public int withdraw(int amt)

    {

        if(amt>balance)

        {

            System.out.println("Insufficient balance");

            return 0;

        }

        else

        {

            balance-=amt;

            return amt;

        }

    }

    public void checkBalance()

    {

        System.out.println("Balance: "+balance);

    }

}

public class L3Q3

{

    public static void main(String args[])

    {

        CurrentAccount ca = new CurrentAccount();

        SavingsAccount sa = new SavingsAccount();

        Scanner sc = new Scanner(System.in);

        int ch;

        do

        {

            System.out.println("1. Deposit");

            System.out.println("2. Withdraw");

            System.out.println("3. Check Balance");

            System.out.println("4. Check Minimum Balance");

            System.out.println("5. Add Interest");

            System.out.println("6. Exit");

            System.out.print("Enter your choice: ");

            ch = sc.nextInt();

            switch(ch)

            {

                case 1:

                    System.out.print("Enter amount to deposit: ");

                    int amt = sc.nextInt();

                    ca.deposit(amt);

                    sa.deposit(amt);

                    break;

                case 2:

                    System.out.print("Enter amount to withdraw: ");

                    amt = sc.nextInt();

                    ca.withdraw(amt);

                    sa.withdraw(amt);

                    break;

                case 3:

                    ca.checkBalance();

                    sa.checkBalance();

                    break;

                case 4:

                    ca.checkMinBalance();

                    break;

                case 5:

                    sa.addInterest();

                    break;

                case 6:

                    break;

                default:

                    System.out.println("Invalid choice");

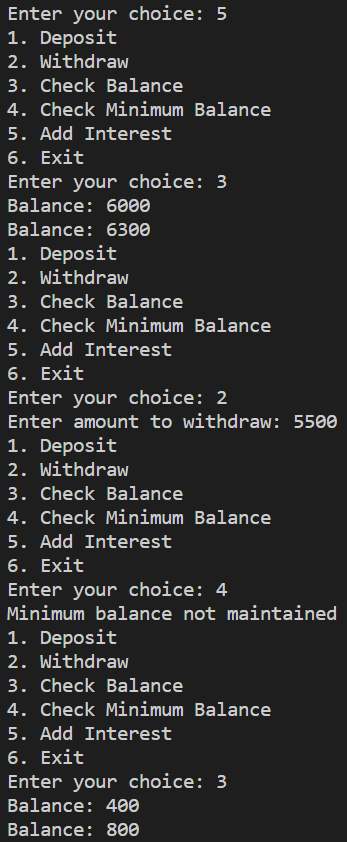
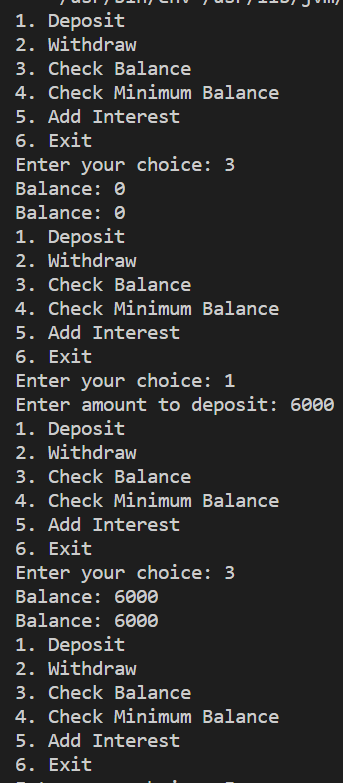
            }

        }while(ch!=6);

    }

}

Output:



4. Add constructors in the above class. Try to access base class constructor in the child class. import java.util.Scanner;

class Account

{

    String cust\_name;

    long acc\_no;

    String acc\_type;

    int balance=0;

    Account(){}

    Account(String cust\_name, long acc\_no, String acc\_type, int balance)

    {

        this.cust\_name = cust\_name;

        this.acc\_no = acc\_no;

        this.acc\_type = acc\_type;

        this.balance = balance;

    }

    public void credit(int amt)

    {

        balance+=amt;

    }

}

class CurrentAccount extends Account

{

    CurrentAccount(){}

    CurrentAccount(String cust\_name, long acc\_no, String acc\_type, int balance)

    {

        super(cust\_name, acc\_no, acc\_type, balance);

    }

    public void deposit(int amt)

    {

        credit(amt);

    }

    public void penalty()

    {

        if(balance<1000)

        {

            balance-=100;

        }

    }

    public void checkMinBalance()

    {

        if(balance<1000)

        {

            System.out.println("Minimum balance not maintained");

            penalty();

        }

        else

        {

            System.out.println("Minimum balance maintained");

        }

    }

    public int withdraw(int amt)

    {

        if(amt>balance)

        {

            System.out.println("Insufficient balance");

            return 0;

        }

        else

        {

            balance-=amt;

            return amt;

        }

    }

    public void checkBalance()

    {

        System.out.println("Balance: "+balance);

    }

}

class SavingsAccount extends Account

{

    SavingsAccount(){}

    SavingsAccount(String cust\_name, long acc\_no, String acc\_type, int balance)

    {

        super(cust\_name, acc\_no, acc\_type, balance);

    }

    public void deposit(int amt)

    {

        credit(amt);

    }

    public void addInterest()

    {

        credit(balance\*5/100);

    }

    public int withdraw(int amt)

    {

        if(amt>balance)

        {

            System.out.println("Insufficient balance");

            return 0;

        }

        else

        {

            balance-=amt;

            return amt;

        }

    }

    public void checkBalance()

    {

        System.out.println("Balance: "+balance);

    }

}

public class L3Q4

{

    public static void main(String args[])

    {

        CurrentAccount ca = new CurrentAccount("Mohith", 123456789, "Current", 1000);

        SavingsAccount sa = new SavingsAccount("Jessica", 987654321, "Savings", 1000);

        Scanner sc = new Scanner(System.in);

        int ch;

        do

        {

            System.out.println("1. Deposit");

            System.out.println("2. Withdraw");

            System.out.println("3. Check Balance");

            System.out.println("4. Check Minimum Balance");

            System.out.println("5. Add Interest");

            System.out.println("6. Exit");

            System.out.print("Enter your choice: ");

            ch = sc.nextInt();

            switch(ch)

            {

                case 1:

                    System.out.print("Enter amount to deposit: ");

                    int amt = sc.nextInt();

                    ca.deposit(amt);

                    sa.deposit(amt);

                    break;

                case 2:

                    System.out.print("Enter amount to withdraw: ");

                    amt = sc.nextInt();

                    ca.withdraw(amt);

                    sa.withdraw(amt);

                    break;

                case 3:

                    ca.checkBalance();

                    sa.checkBalance();

                    break;

                case 4:

                    ca.checkMinBalance();

                    break;

                case 5:

                    sa.addInterest();

                    break;

                case 6:

                    break;

                default:

                    System.out.println("Invalid choice");

            }

        }while(ch!=6);

    }

}

Output:



5. An educational institute wishes to maintain a database.

import java.util.Scanner;

class Staff

{

    int code;

    String name;

    Staff(int code, String name)

    {

        this.code = code;

        this.name = name;

    }

    void display()

    {

        System.out.println("Code: "+code);

        System.out.println("Name: "+name);

    }

}

class Teacher extends Staff

{

    String subject;

    String publication;

    Teacher(int code, String name, String subject, String publication)

    {

        super(code, name);

        this.subject = subject;

        this.publication = publication;

    }

    void display()

    {

        System.out.println("Code: "+code);

        System.out.println("Name: "+name);

        System.out.println("Subject: "+subject);

        System.out.println("Publication: "+publication);

    }

}

class Officer extends Staff

{

    char grade;

    Officer(int code, String name, char grade)

    {

        super(code, name);

        this.grade = grade;

    }

    void display()

    {

        System.out.println("Code: "+code);

        System.out.println("Name: "+name);

        System.out.println("Grade: "+grade);

    }

}

class Typist extends Staff

{

    int speed;

    Typist(int code, String name, int speed)

    {

        super(code, name);

        this.speed = speed;

    }

    void display()

    {

        System.out.println("Code: "+code);

        System.out.println("Name: "+name);

        System.out.println("Speed: "+speed);

    }

}

class Regular extends Typist

{

    int salary;

    Regular(int code, String name, int speed, int salary)

    {

        super(code, name, speed);

        this.salary = salary;

    }

    void display()

    {

        super.display();

        System.out.println("Salary: "+salary);

    }

}

class Casual extends Typist

{

    int daily\_wages;

    Casual(int code, String name, int speed, int daily\_wages)

    {

        super(code, name, speed);

        this.daily\_wages = daily\_wages;

    }

    void display()

    {

        super.display();

        System.out.println("Daily Wages: "+daily\_wages);

    }

}

public class L3Q5

{

    public static void main(String args[])

    {

        Staff database[] = new Staff[10];

        database[0] = new Teacher(1, "A", "Maths", "Oxford");

        database[1] = new Teacher(2, "B", "Physics", "Pearson");

        database[2] = new Teacher(3, "C", "Chemistry", "Oxford");

        database[3] = new Officer(4, "D", 'A');

        database[4] = new Officer(5, "E", 'B');

        database[5] = new Regular(6, "F", 50, 10000);

        database[6] = new Regular(7, "G", 60, 12000);

        database[7] = new Regular(8, "H", 70, 14000);

        database[8] = new Casual(9, "I", 40, 500);

        database[9] = new Casual(10, "J", 45, 600);

        for (int i=0;i<10;i++)

        {

            System.out.println("Employee "+(i+1));

            database[i].display();

            System.out.println("\n");

        }

    }

}

Output:

