

Week 5

Name: Aaryan Prakash

USN: 1BM23CS006

Q) Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed.

Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

- i. Accept deposit from customer and update the balance.**
- ii. Display the balance.**
- iii. Compute and deposit interest**
- iv. Permit withdrawal and update the balance**
- v. Check for the minimum balance, impose penalty if necessary and update the balance.**

Code:

```
import java.util.Scanner;
```

```
class Account {  
    protected String customerName;  
    protected int accountNumber;  
    protected double balance;  
  
    public Account(String customerName, int accountNumber, double balance) {  
        this.customerName = customerName;  
        this.accountNumber = accountNumber;  
        this.balance = balance;  
    }  
  
    public void deposit(double amount) {  
        if (amount > 0) {  
            balance += amount;  
            System.out.println("Deposited: " + amount);  
        } else {
```

```

        System.out.println("Invalid deposit amount");
    }
}

public void displayBalance() {
    System.out.println("Balance: " + balance);
}
}

class SavAcct extends Account {
    private double interestRate;

    public SavAcct(String customerName, int accountNumber, double balance, double
interestRate) {
        super(customerName, accountNumber, balance);
        this.interestRate = interestRate;
    }

    public void computeAndDepositInterest() {
        double interest = balance * (interestRate / 100);
        balance += interest;
        System.out.println("Interest added: " + interest);
    }

    public void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
            System.out.println("Withdrawn: " + amount);
        } else {
            System.out.println("Insufficient balance for withdrawal");
        }
    }
}

class CurAcct extends Account {
    private double minimumBalance;
    private double serviceCharge;

    public CurAcct(String customerName, int accountNumber, double balance, double
minimumBalance, double serviceCharge) {
        super(customerName, accountNumber, balance);
        this.minimumBalance = minimumBalance;
        this.serviceCharge = serviceCharge;
    }

    public void withdraw(double amount) {
        if (amount <= balance) {

```

```

        balance -= amount;
        System.out.println("Withdrawn: " + amount);

        if (balance < minimumBalance) {
            balance -= serviceCharge;
            System.out.println("Service charge imposed: " + serviceCharge);
        }
    } else {
        System.out.println("Insufficient balance for withdrawal");
    }
}
}

public class Bank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Create a savings account
        SavAcct savAcc = new SavAcct("Alice", 12345, 1000, 5);

        // Create a current account
        CurAcct curAcc = new CurAcct("Bob", 67890, 2000, 500, 50);

        System.out.println("Choose Account Type:\n1. Savings Account\n2. Current Account");
        int choice = sc.nextInt();

        switch (choice) {
            case 1:
                System.out.println("Savings Account Selected");
                savAcc.deposit(500);
                savAcc.computeAndDepositInterest();
                savAcc.withdraw(300);
                savAcc.displayBalance();
                break;

            case 2:
                System.out.println("Current Account Selected");
                curAcc.deposit(500);
                curAcc.withdraw(1800);
                curAcc.displayBalance();
                break;

            default:
                System.out.println("Invalid choice");
        }
    }
}

```

```
        sc.close();
    }
}
```

Output:

```
Choose Account Type:
1. Savings Account
2. Current Account
1
Savings Account Selected
Deposited: 500.0
Interest added: 75.0
Withdrawn: 300.0
Balance: 1275.0
```

```
Choose Account Type:
1. Savings Account
2. Current Account
2
Current Account Selected
Deposited: 500.0
Withdrawn: 1800.0
Balance: 700.0
```