WEEK 5:

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:

- a) Accept deposit from customer and update the balance.
- b) Display the balance.
- c) Compute and deposit interest
- d) Permit withdrawal and update the balance

Check for the minimum balance, impose penalty if necessary and update the balance.

Source Code:

```
import java.util.Scanner;
class Account {
     String customerName;
     int accountNumber;
     String accountType;
     double balance;
    public Account(String customerName, int accountNumber, String accountType) {
        this.customerName = customerName;
        this.accountNumber = accountNumber;
        this.accountType = accountType;
        this.balance = 0.0;
    public void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            System.out.println("Amount deposited: " + amount);
            System.out.println("Updated balance: " + balance);
        } 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 interestRate) {
        super(customerName, accountNumber, "Savings");
        this.interestRate = interestRate;
    public void computeAndDepositInterest() {
        double interest = balance * (interestRate / 100);
        balance += interest;
        System.out.println("Interest added: " + interest);
        System.out.println("Updated balance: " + balance);
    public void withdraw(double amount) {
        if (amount <= balance) {</pre>
            balance -= amount;
            System.out.println("Amount withdrawn: " + amount);
            System.out.println("Updated balance: " + balance);
        } else {
            System.out.println("Insufficient balance!");
class CurAcct extends Account {
  double minimumBalance;
   double serviceCharge;
    public CurAcct(String customerName, int accountNumber, double minimumBalance,
double serviceCharge) {
        super(customerName, accountNumber, "Current");
        this.minimumBalance = minimumBalance;
        this.serviceCharge = serviceCharge;
```

```
public void withdraw(double amount) {
        if (amount <= balance) {</pre>
            balance -= amount;
            System.out.println("Amount withdrawn: " + amount);
            if (balance < minimumBalance) {</pre>
                imposePenalty();
            System.out.println("Updated balance: " + balance);
            System.out.println("Insufficient balance!");
    private void imposePenalty() {
        balance -= serviceCharge;
        System.out.println("Balance fell below minimum. Service charge imposed: "
+ serviceCharge);
public class Bank {
   public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Choose account type:\n1. Savings Account\n2. Current
Account");
        int choice = scanner.nextInt();
        scanner.nextLine();
        System.out.println("Enter customer name: ");
        String name = scanner.nextLine();
        System.out.println("Enter account number: ");
        int accNum = scanner.nextInt();
        if (choice == 1) {
            System.out.println("Enter interest rate for savings account: ");
            double interestRate = scanner.nextDouble();
            SavAcct savAccount = new SavAcct(name, accNum, interestRate);
            System.out.println("Enter amount to deposit: ");
            double deposit = scanner.nextDouble();
            savAccount.deposit(deposit);
            savAccount.computeAndDepositInterest();
            System.out.println("Enter amount to withdraw: ");
```

```
double withdrawAmount = scanner.nextDouble();
            savAccount.withdraw(withdrawAmount);
        } else if (choice == 2) {
            System.out.println("Enter minimum balance for current account: ");
            double minBalance = scanner.nextDouble();
            System.out.println("Enter service charge for falling below minimum
balance: ");
            double serviceCharge = scanner.nextDouble();
            CurAcct curAccount = new CurAcct(name, accNum, minBalance,
serviceCharge);
            System.out.println("Enter amount to deposit: ");
            double deposit = scanner.nextDouble();
            curAccount.deposit(deposit);
            System.out.println("Enter amount to withdraw: ");
            double withdrawAmount = scanner.nextDouble();
            curAccount.withdraw(withdrawAmount);
        } else {
            System.out.println("Invalid account type selected.");
        scanner.close();
```

Output:

```
Choose account type:
1. Savings Account
2. Current Account
Enter customer name:
sagar
Enter account number:
1234
Enter interest rate for savings account:
Enter amount to deposit:
5000
Amount deposited: 5000.0
Updated balance: 5000.0
Interest added: 150.0
Updated balance: 5150.0
Enter amount to withdraw:
4800
Amount withdrawn: 4800.0
Updated balance: 350.0
```

```
Choose account type:
1. Savings Account
2. Current Account
Enter customer name:
chetan
Enter account number:
9876
Enter minimum balance for current account:
Enter service charge for falling below minimum balance:
150
Enter amount to deposit:
6000
Amount deposited: 6000.0
Updated balance: 6000.0
Enter amount to withdraw:
5200
Amount withdrawn: 5200.0
Balance fell below minimum. Service charge imposed: 150.0
Updated balance: 650.0
```

OBSERVATION:

5. Develop a Fava program to creade a claus want that maintains two hends of account for ets automers, one of them could savelings account and the others current account. The saveling account provides compound interest and with doount facilities but no cheque book facellity, The current account provider cheque book facility but no interest current account holder should also malitain a infiremum balance and of the balance falls below this level, a service charge is imposed. Create a clair Account that stores customer name, account number and type of account. From this durive the clauses cer-acct and sav-acet to make them more specific to their requirements. Include the necessary methord en order to achive the following tasks. @ Accept dyposite from curtomer and uptilate the bulance 6 D& play the balance @ compute and deposit interest @ Permit withdrawal and update the balance check for the minimum balance, 9 mpose penalty if necessary and updale to balance us Emport Lava. util. Scanner; class Account & stoling curtname; Ent acc-no; strong accetype, Public Account (swing ausname, ent account no, swing type curt-name = cusname; acc-no = account-no; ace-type = type-ace; balance = 0.0;

```
Public word deposit (double amount)
       if (amount so)
            balance = balance + account;
            System.out-println ("Around agos tod:" + amount);
            System out printle (" up dated balance: " + balance);
       eluatre
          System . Out. pointin( "Foralise ");
    4
   public void displaybalance ()
       System. Out . printle ("Balance"+ balance);
Clair Savacet extend Account
          double EnterestRate;
  Public SavAccoul (String Contemes Name)
       Super (custoname, acc-no);
       this of nevertrate = Enterestrate;
   3
  Public void DepositIntred()
   double Entired = balance + (Enterest Hale 100);
      balance = balane + Enterest;
      System.out. println (" Interest added: " + Enterest);
     System out . println ("up douted balance" " + balance);
```

```
withdraw (double amount)
   if (amount == balance)
      d
          balance = balance - amount;
           System - out-println ("Amount wathdraw", " + amount);
           Syntim - ocal - point in ("applicated balance" + balance);
      3 che
          system - out - pointful " Insufficient balance";
     8
down arrent
class curact extends accounted
 double menemum balance;
 double
           service charge;
 Public Curact (string contrame, ent acoro, double menimentalance,
         double Servicecharge)
      super (Cutonhame, accho)
      this. Se-svice charge;
 Public vold with draw (double amount)
    if (amount c = balance)
       balance e-amount?
       System out pointing " Amount withdrow" + amount);
       Tot balance a menemembalance)
        d imposepenally();
       System-out- println ("updated balance" + balance);
          u
System. och. Printik (" In sufficient balance");
    3
```

```
vard Proposedenally ()
      Porvale
         balance = balance - service charge;
         System - Old - println( "Balance is mener , service charge imposed" + services
Public clau Bank
    Public static void main (sning Dasgn)
      Scanner Scanner = new Scanner (Systemo in);
      System. ocd . println ("choose account type: in 1. Saveny acc in
      Put choice = Scanners-newtint();
      Scanner next (snec);
      System · och printin ("tribus customos name");
      string name = 9 canners . next line();
      System . out . privaln (" the account numbers");
      Ent acc-now = Scarner-next Ent ();
      El [chorce = =1)
        Systemoout. Proten (" tolor Entrect route for sowing acc");
         double Enterent Part = Scanner. next Double ();
         Sou Acet Savoccout = reco sau Ace (name, accour, interestrecte);
         system out printly (" cuts amount to deposite");
         double deposite = 9 comes. rest couble ();
          San Account. deposéte ( deposéte);
          San Accompations Depositiontret();
         System out opsint in (" entr or mount to with draw");
         double with document = scanner. next Double ();
         Sav Acc. with focus (with to an Account);
```

```
ette of (chorce == 2)
         System. och printin " tribs remembalane for correct as ",
( anax)
         double men balance = Scanner-next double ();
          Syntin · out · println (" Entr service charge ") ",
           double service charg = scanner, newdouble();
          currect currecount = new currect (mame, accnown, normbalance,
                                   servicechorge);
        System . Och println " fates amount to cerithdraw ");
         double withdrawAmound = scanners.nextdouble ()",
        Cur Acco withdrawl (withdrawhenoust);
       2
         elre
           System. oco-println (" Invalid");
         Scannoero clareco;
    4
    OUTPUT!
        choose account type:
          1. Savengs Account
          2. current Account
                                             culquishames
                                       enter
          Enter curtomer name:
                                        abc
                                        ontrace-no:
           ABC
                 acc-no %
           enter
           Enter Enterest rate for savings account:
                                        outs men balance for current acc
                                          1000
                                          outr service charge for falling
            Enter amount to diposite :
                                                   below hen bodane:
            Amount deposted: 2000
            1000
            opdated balance: 1000
                                          200
                                          onto amond to deposite.
            Interest added : 60
                                           500
             update balance: 1060
                                           pmoud deposite; 100:0
                                           sporter balance 500.0
            Amount with
             Enter amount to with draw ?
                                          Enter amound to withdraw : 100
                                          Balance for btelow nim . : 200
            Amount withdraw: 500.0
                                          updated balance = 200
            updated balance : 560.
```