9/1/24

LAB - 5 Bank Account

```
import java.util.Scanner;
class Account {
  String customerName;
  long accountNumber;
  String accounttype;
  double balance;

  public Account (String customerName,
    long accountNumber, String accounttype
    double balance)
  {
    this.CustomerName = customerName;
    this.AccountNumber = AccountNumber;
    this.accounttype = accounttype;
    this.Balance = balance;
  }
  public void deposit (double amount)
  {
    balance += amount;
    System.out.println("Deposit of $"
                + amount + "
                Successful updated
                balance $" +
                balance);
  }
  public void displayBalance()
  {
    System.out.println("Acc
                balance: $"
                + balance);
  }
}
```

```
class CurAccount extends Account{

    double minBalance;
    double ServiceCharge;

public CurAccount (String CustomerName,
    long ~~ban~~ accountNumber, double
    balance, double minbalance,
    double ServiceCharge)
{
    super(CustomerName, accountNumber,
    "Current", balance);
    This.minBalance = minBalance};
    this.servicecharge = ServiceCharge;
}
public void checkminBalance()
  {
    if(balance < minBalance)
      {
        balance - = Servicecharge;
      System.out.println("Min. balance
      maintained. Service charge of $"+
        servicecharge + imposed");
        displayBalance();
      }
  }
public void withdraw (double amount)
    if(amount > balance){
      System.out.println("Insufficient
      funds. Withdrawal failed");
}
  else{

    balance = - amount;
```

```
        System.out.println("Withdrawal of $" +
                amount + "Successful.
                Updated
                balance: $" + balance);

        checkMinBalance();
    }
}

class SavAccount extends Account{
    double interestRate;

    public SavAccount(String customerName,
    long accountNumber, double balance,
    double interestRate)
    {
        super(CustomerName, accountNumber,
        "Savings", balance);
        this.interestRate = interestRate;
    }
    public void computeInterest
                                    (){
        double interest = balance * (int
                                    Rate/
                                    100);
        balance += interest;
        System.out.println("Interested
                                    computed and
                                    deposited: $" +
                                    interest);
        displayBalance();
    }
}
```

```
public void withdraw (double amount
    if (amount > balance)
        System.out.println ("Insufficient
        funds. Withdrawal failed");
    }
    else {
        balance = - amount;
        System.out.println ("Withdrawal of
            $" + amount + " successful.
                                updated
                                balance: $" +
                                balance);

            checkMinBalance();
        }
    }
}

class SavAccount extends Account{
double interestRate;
public savaccount { String customer
                                Name,
                    long accountNumb,
                    double balance,
                    double interestRa

    super (customerName, accountNumber,
        "Savings", balance);
    this.interestRate = interestRate;
}
public void computeInterest() {
    double interest = balance
                        (intRate/100);
    balance += interest;
```

```
System.out.println("interested computed and
                    deposited = $" +
                    interest);

displayBalance();
}
public void withdraw(double
                     amount){

if (amount > balance)
{
    System.out.println("Insufficient
                        funds. withdrawal
                        failed");
}
else{

balance -= amount;
System.out.println("Withdrawal
    of $" + amount + "Successful.
    update balance: $" + balance);
}
}
}
public class Bank {
    public static void main(String
                            args[])
    {
        Scanner scanner = new
                          Scanner
                          (System.in);
```

```
CurAccount currentAccount =
        new CurAccount("J",
                       12346789
                       1000, 10);

SavAccount SavingsAccount = new
                SavAccount(
                "D", 9845234,
                2000, 5);

    int choice;
    do {

    System.out.println("Select an option");
    System.out.println("1. Deposite");
    System.out.println("2. Display Balance");
    System.out.println("3. Compute Interest");
    System.out.println("4. Withdraw");
    System.out.println("5. Exit");
    System.out.println("Enter your
                        choice");
    choice = Scannee.nextInt();

switch choice() {

 Case 1:
     System.out.println("Enter amount
                         to deposit");

     double depositAmount =
                Scannes.nextDouble();

     System.out.println("Select account
                (1. Current,
                 2. Savings)");
```

```
int acctype = scannee.nextInt();
if (acctype == 1)
  {
    currentAccount.displayBalance();
  }

  else if (acctype == 2)
  {
    SavingsAccount displayBalance();
  }
  else
  {
    System.out.println("Invalid acc
                        type");
  }
  break;
case 2:
    System.out.println("Select acc
                        (1. Current,
                          2. Savings):
                        ");

      int acctype == 1)
  { }
    currentAccount.displayBalance();
  }
  else if (acctype == 2)
    {
      SavingsAccount.DisplayBalance().
  }
  else {
      System.out.println("Invalid");
    }
    break;
  case 3:
    if(SavingsAccount instanceof
                        SavAccount)
```

```
{
    ((SavAccount) SavingsAccount).
                    computeInterest();
}
else{
    System.out.println("Invalid");
}
break;
case 4:
    System.out.println("Enter
                    account to be
                    withdrawed");
double withdrawAmount =
                    Scanner.next
                    Double();
System.out.println("Select Acc
                    (1. Current,
                    2. Savings): ");
int accType = Scannee.nextInt();
if (accType == 1)
{
    currentAccount.withdraw(with
                    drawAmt);
}
else if (AccType == 2)
{
    SavingsAccount.withdraw(withdra
                    drawAcc);
}
else {
    System.out.println("Invalide");
}
break;
Case 5:
```

```
            System.out.println("Exiting");
            break;
        default:
            System.out.println("Invalid");
    }
}
while (choice != 5)
    scanner.close();
}
}
```

output

```
Select an option
1. Deposit
2. Display Balance
3. Compute Interest
4. Withdraw
5. Exit
Enter your choice: 1
Enter amount to deposit: 1000
Select account (1. Current 2. Savings): 2
Deposit of $10000.0 Successful.
```

AB

Fri 12/1/24