Lab 5:

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:

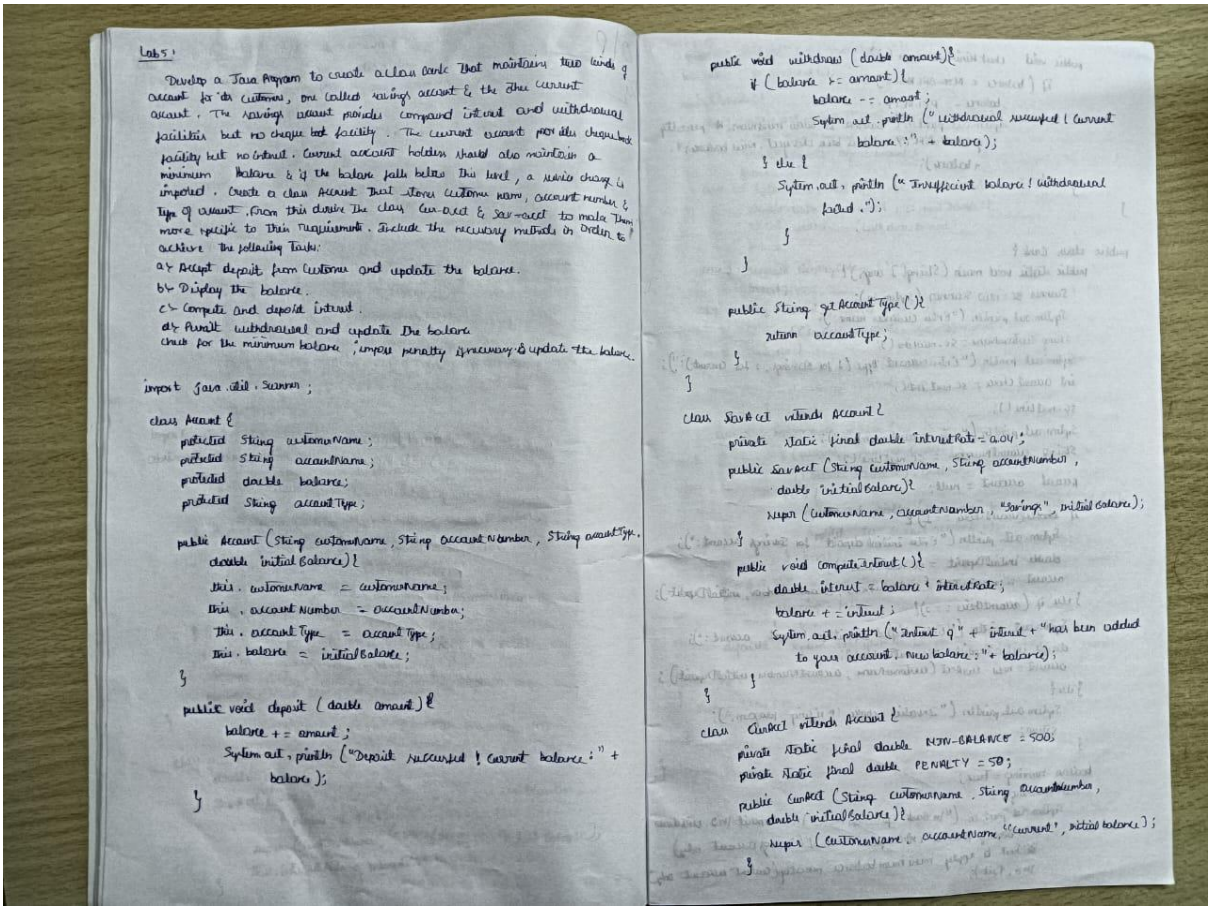
a) Accept deposit from customer and update the balance.

b) Display the balance.

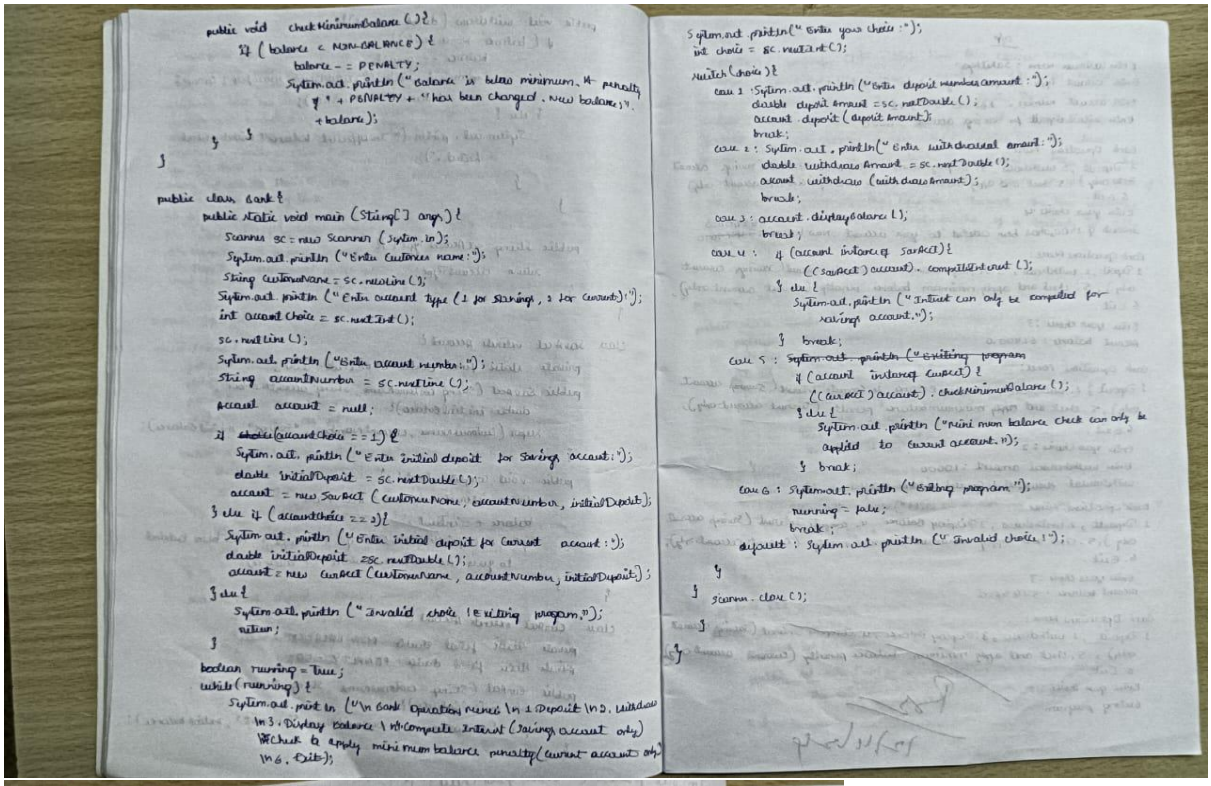
c) Compute and deposit interest

d) Permit withdrawal and update the balance

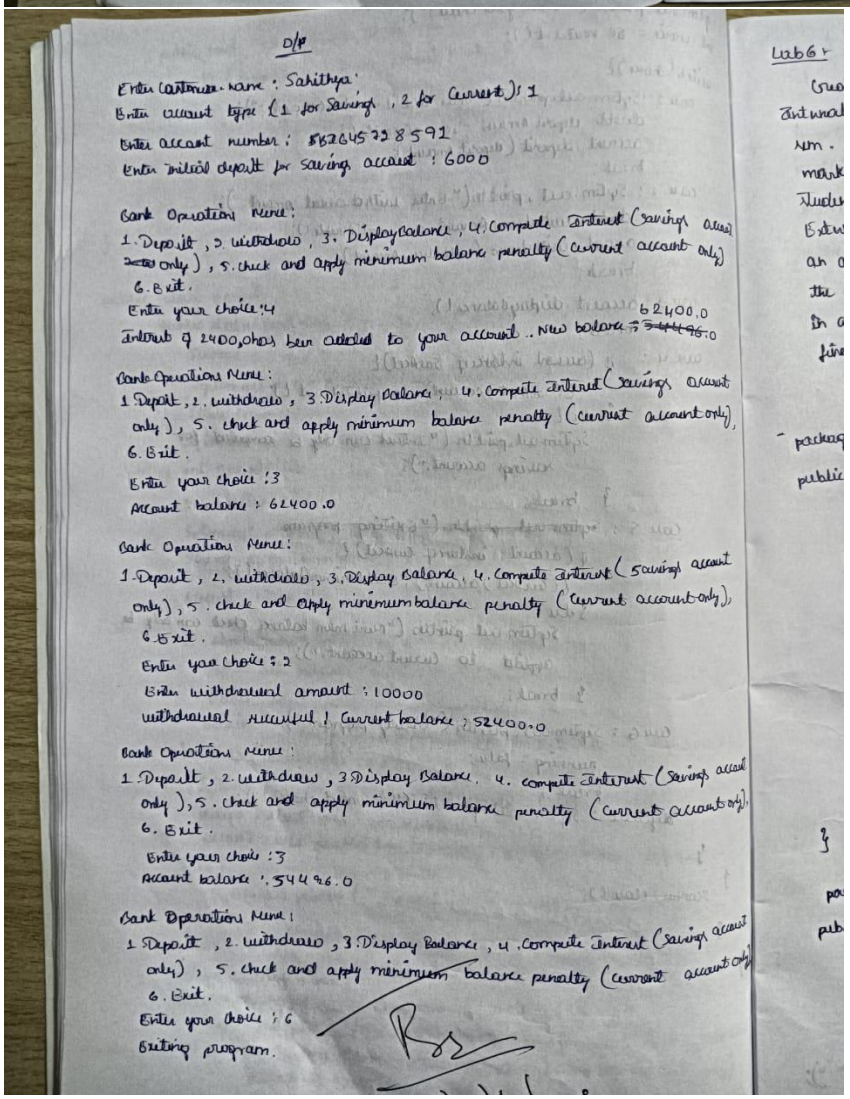
Check for the minimum balance, impose penalty if necessary and update the balance.



```
Lab 5:
Develop a Java Program to create a class Bank that maintains two kinds of
account for its customers, one called savings account & 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 & if the balance falls below this level, a service charge is
imposed. Create a class Account that stores customer name, account number &
type of account. From this derive the class Cur-acct & 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 & update the balance.

import java.util.Scanner;

class Account {
    protected String customerName;
    protected String accountName;
    protected double balance;
    protected String accountType;

    public Account (String customerName, String accountNumber, String accountType,
        double initialBalance) {
        this.customerName = customerName;
        this.accountNumber = accountNumber;
        this.accountType = accountType;
        this.balance = initialBalance;
    }

    public void deposit (double amount) {
        balance += amount;
        System.out.println("Deposit successful ! Current balance:" +
            balance);
    }

    public void withdraw (double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println("Withdrawal successful ! Current
                balance:" + balance);
        } else {
            System.out.println("Insufficient balance ! Withdrawal
                failed.");
        }
    }

    public String getAccountType() {
        return accountType;
    }
}

class SavAcct extends Account {
    private static final double interestRate = 0.04;
    public SavAcct (String customerName, String accountNumber,
        double initialBalance) {
        super (customerName, accountNumber, "Saving", initialBalance);
    }
    public void computeInterest() {
        double interest = balance * interestRate;
        balance += interest;
        System.out.println("Interest @" + interest + "has been added
            to your account. New balance:"+ balance);
    }
}

class CurAcct extends Account {
    private static final double MIN_BALANCE = 500;
    private static final double PENALTY = 50;
    public CurAcct (String customerName, String accountNumber,
        double initialBalance) {
        super (customerName, accountNumber, "Current", initialBalance);
    }
}
```



```
public void checkMinimumBalance(){
    if (balance < MIN_BALANCE) {
        balance -= PENALTY;
        System.out.println("Balance is below minimum. A penalty
            of " + PENALTY + " has been charged. New balance: "
            + balance);
    }
}

public class Bank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Customer name:");
        String customerName = sc.nextLine();
        System.out.println("Enter account type (1 for Savings, 2 for Current):");
        int accountChoice = sc.nextInt();
        sc.nextLine();
        System.out.println("Enter account number:");
        String accountNumber = sc.nextLine();
        Account account = null;
        if (accountChoice == 1) {
            System.out.println("Enter initial deposit for Savings account:");
            double initialDeposit = sc.nextDouble();
            account = new SavAct(customerName, accountNumber, initialDeposit);
        } else if (accountChoice == 2) {
            System.out.println("Enter initial deposit for Current account:");
            double initialDeposit = sc.nextDouble();
            account = new CurAct(customerName, accountNumber, initialDeposit);
        } else {
            System.out.println("Invalid choice! Exiting program.");
            return;
        }
        boolean running = true;
        while (running) {
            System.out.println("\n Bank Operations Menu: \n 1.Deposit \n 2. Withdraw
                \n 3. Display Balance \n 4. Compute Interest (Savings account only)
                \n 5. Check & apply minimum balance penalty (Current account only)
                \n 6. Exit");
            System.out.println("Enter your choice:");
            int choice = sc.nextInt();
            switch (choice) {
                case 1: System.out.println("Enter deposit amount:");
                    double depositAmount = sc.nextDouble();
                    account.deposit(depositAmount);
                    break;
                case 2: System.out.println("Enter withdrawal amount:");
                    double withdrawAmount = sc.nextDouble();
                    account.withdraw(withdrawAmount);
                    break;
                case 3: account.displayBalance();
                    break;
                case 4: if (account instanceof SavAct) {
                        ((SavAct) account).computeInterest();
                    } else {
                        System.out.println("Interest can only be computed for
                            savings account.");
                    } break;
                case 5: if (account instanceof CurAct) {
                        ((CurAct) account).checkMinimumBalance();
                    } else {
                        System.out.println("Min balance check can only be
                            applied to current account.");
                    } break;
                case 6: System.out.println("Exiting program");
                    running = false;
                    break;
                default: System.out.println("Invalid choice!");
            }
        }
        scanner.close();
    }
}
```



O/P

Enter Customer name: Sahithya
Enter account type (1 for Savings, 2 for Current): 1
Enter account number: 582645228591
Enter initial deposit for saving account: 6000

Bank Operations Menu:
1. Deposit, 2. Withdraw, 3. Display Balance, 4. Compute Interest (savings account only), 5. Check and apply minimum balance penalty (current account only)
6. Exit.
Enter your choice: 4
Interest of 2400.0 has been added to your account. New balance: 62400.0

Bank Operations Menu:
1 Deposit, 2. withdraw, 3 Display Balance, 4. Compute Interest (Savings account only), 5. Check and apply minimum balance penalty (current account only),
6. Exit.
Enter your choice: 3
Account balance: 62400.0

Bank Operations Menu:
1. Deposit, 2. withdraw, 3. Display Balance, 4. Compute Interest (Savings account only), 5. Check and apply minimum balance penalty (current account only),
6. Exit.
Enter your choice: 2
Enter withdrawal amount: 10000
withdrawal successful! Current balance: 52400.0

Bank Operations Menu:
1. Deposit, 2. withdraw, 3. Display Balance, 4. Compute Interest (Savings account only), 5. Check and apply minimum balance penalty (current account only),
6. Exit.
Enter your choice: 3
Account balance: 54496.0

Bank Operations Menu:
1. Deposit, 2. withdraw, 3. Display Balance, 4. Compute Interest (Savings account only), 5. Check and apply minimum balance penalty (current account only),
6. Exit.
Enter your choice: 6
Exiting program.

Output:

```
D:\Java_Lab_Programs>javac Bank.java

D:\Java_Lab_Programs>java Bank
Enter customer name: Shasha
Enter account type (1 for Savings, 2 for Current): 1
Enter account number: SBI123456789
Enter initial deposit for Savings account: 6000

Bank Operations Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings account only)
5. Check and apply minimum balance penalty (Current account only)
6. Exit
Enter your choice: 4
Interest of 240.0 has been added to your account. New balance: 6240.0

Bank Operations Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings account only)
5. Check and apply minimum balance penalty (Current account only)
6. Exit
Enter your choice: 3
Account balance: 6240.0

Bank Operations Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings account only)
5. Check and apply minimum balance penalty (Current account only)
6. Exit
Enter your choice: 2
Enter withdrawal amount: 1000
Withdrawal successful! Current balance: 5240.0
```

```
Bank Operations Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings account only)
5. Check and apply minimum balance penalty (Current account only)
6. Exit
Enter your choice: 3
Account balance: 5240.0

Bank Operations Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings account only)
5. Check and apply minimum balance penalty (Current account only)
6. Exit
Enter your choice: 6
Exiting program.

D:\Java_Lab_Programs>
```