LAB 5 – BANK ACCOUNT

SOURCE CODE:

```
import java.util.Scanner;

class Account {
    private String customerName;
    private long accountNumber;
    private String accountType;
    protected 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("Deposited: Rs " + amount);
    }

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

    public String getCustomerName() {
        return customerName;
    }

    public long getAccountNumber() {
        return accountNumber;
    }

    public String getAccountType() {
        return accountType;
    }
}

class SavingsAccount extends Account {
    private final double interestRate;

    public SavingsAccount(String customerName, long accountNumber, double balance) {
        super(customerName, accountNumber, "Savings", balance);
        this.interestRate = 8; // 8% interest rate
    }
```

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

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            System.out.println("Withdrawn: Rs " + amount);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

class CurrentAccount extends Account {
    private final double minimumBalance;
    private final double serviceCharge;

    public CurrentAccount(String customerName, long accountNumber, double balance) {
        super(customerName, accountNumber, "Current", balance);
        this.minimumBalance = 10000; // Rs 10000 minimum balance
        this.serviceCharge = 100; // Rs 100 service charge
    }

    public void withdraw(double amount) {
        if (balance - amount >= minimumBalance) {
            balance -= amount;
            System.out.println("Withdrawn: Rs " + amount);
        } else {
            System.out.println("Insufficient funds (minimum balance not maintained)");
        }
    }

    public void deductServiceCharge() {
        if (balance < minimumBalance) {
            balance -= serviceCharge;
            System.out.println("Service charge deducted: Rs " + serviceCharge);
        }
    }
}

public class Bank {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
            System.out.println("1BM22CS179");
```

```
        // Create savings account
        System.out.println("Creating a Savings Account");
        System.out.print("Enter customer name: ");
        String customerName = scanner.nextLine();
        System.out.print("Enter account number: ");
        long accountNumber = scanner.nextLong();
        System.out.print("Enter initial balance: Rs ");
        double initialBalance = scanner.nextDouble();
        SavingsAccount savingsAccount = new SavingsAccount(customerName, accountNumber,
initialBalance);

        // Create current account
        System.out.println("\nCreating a Current Account");
        scanner.nextLine(); // Consume newline
        System.out.print("Enter customer name: ");
        customerName = scanner.nextLine();
        System.out.print("Enter account number: ");
        accountNumber = scanner.nextLong();
        System.out.print("Enter initial balance: Rs ");
        initialBalance = scanner.nextDouble();
        CurrentAccount currentAccount = new CurrentAccount(customerName, accountNumber,
initialBalance);

        // Perform operations on accounts
        while (true) {
            System.out.println("\nSelect an account type:");
            System.out.println("1. Savings Account");
            System.out.println("2. Current Account");
            System.out.println("3. Exit");
            System.out.print("Choice: ");
            int choice = scanner.nextInt();

            switch (choice) {
                case 1:
                    System.out.println("\nSavings Account Menu");
                    System.out.println("1. Deposit");
                    System.out.println("2. Withdraw");
                    System.out.println("3. Display Balance");
                    System.out.println("4. Compute and Deposit Interest");
                    System.out.println("5. Exit");
                    System.out.print("Choice: ");
                    int savingsChoice = scanner.nextInt();
                    switch (savingsChoice) {
                        case 1:
                            System.out.print("Enter amount to deposit: Rs ");
                            double depositAmount = scanner.nextDouble();
                            savingsAccount.deposit(depositAmount);
                            break;
```

```
            case 2:
                System.out.print("Enter amount to withdraw: Rs ");
                double withdrawAmount = scanner.nextDouble();
                savingsAccount.withdraw(withdrawAmount);
                break;
            case 3:
                savingsAccount.displayBalance();
                break;
            case 4:
                savingsAccount.compoundInterest();
                break;
            case 5:
                System.out.println("Exiting Savings Account Menu");
                break;
            default:
                System.out.println("Invalid choice");
        }
        break;
    case 2:
        System.out.println("\nCurrent Account Menu");
        System.out.println("1. Deposit");
        System.out.println("2. Withdraw");
        System.out.println("3. Display Balance");
        System.out.println("4. Deduct Service Charge");
        System.out.println("5. Exit");
        System.out.print("Choice: ");
        int currentChoice = scanner.nextInt();
        switch (currentChoice) {
            case 1:
                System.out.print("Enter amount to deposit: Rs ");
                double depositAmount = scanner.nextDouble();
                currentAccount.deposit(depositAmount);
                break;
            case 2:
                System.out.print("Enter amount to withdraw: Rs ");
                double withdrawAmount = scanner.nextDouble();
                currentAccount.withdraw(withdrawAmount);
                break;
            case 3:
                currentAccount.displayBalance();
                break;
            case 4:
                currentAccount.deductServiceCharge();
                break;
            case 5:
                System.out.println("Exiting Current Account Menu");
                break;
            default:
```

```
                System.out.println("Invalid choice");
            }
            break;
        case 3:
            System.out.println("Exiting Bank Application");
            System.exit(0);
            break;
        default:
            System.out.println("Invalid choice");
      }
    }
  }
}
```

OUTPUT: