Day-3 Bank System Assignment

Bank.java

```
package Assignment;

public interface Bank {
    void deposit(double amount); void
    withdraw(double amount); double
    getBalance();
}
```

AbstractAccount.java

```
package Assignment;

public abstract class AbstractAccount implements Bank { protected String
    accountNumber;
    protected String accountHolderName; protected
    double balance;

    public AbstractAccount(String accountNumber, String accountHolderName) { this.accountNumber = accountNumber;
        this.accountHolderName = accountHolderName; this.balance = 0.0;
    }

    @Override
    public void deposit(double amount) { if (amount > 0) {
            balance += amount;
        }
    }

    @Override
    public double getBalance() { return balance;
    }

    public abstract void withdraw(double amount);
}
```

SavingsAccount.java

```
package Assignment;

public class SavingsAccount extends AbstractAccount { private double interestRate;

        public SavingsAccount(String accountNumber, String accountHolderName, double interestRate) {
        super(accountNumber, accountHolderName); this.interestRate =
        interestRate;
    }
    @Override
    public void withdraw(double amount) {
        if (amount > 0 && amount <= balance) { balance -= amount;
        }
    }
}
```

Transaction.java

```
package Assignment;

public class Transaction { private String type;
    private double amount;

    public Transaction(String type, double amount) { this.type = type;
        this.amount = amount;
    }

    public String getType() { return type;
    }

    public double getAmount() { return amount;
    }

    public void display() { System.out.println(type + ": " + amount);
    }
}
```

Main.java

```
package Assignment;

importjava.util.ArrayList; import
java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        SavingsAccount acc = new SavingsAccount("SB1001", "Riya", 4.5); ArrayList<Transaction>
        transactions = new ArrayList<>(); Scanner sc = new Scanner(System.in);
        intchoice; do {
            System.out.println("\n1. Deposit"); System.out.println("2.
            Withdraw"); System.out.println("3. Check Balance");
```

```
                System.out.println("4. View Transactions"); System.out.println("5.
                Exit"); System.out.print("Enter your choice: "); choice = sc.nextInt();

                switch (choice) { case 1:
                        System.out.print("Enter amount to deposit: "); double dAmount =
                        sc.nextDouble(); acc.deposit(dAmount);
                        transactions.add(new Transaction("Deposit", dAmount)); break;
                    case 2:
                        System.out.print("Enter amount to withdraw: "); double wAmount =
                        sc.nextDouble(); acc.withdraw(wAmount);
                        transactions.add(new Transaction("Withdraw", wAmount)); break;
                    case 3:
                        System.out.println("Balance: " + acc.getBalance()); break;
                    case 4:
                        for (Transaction t : transactions) { t.display();
                        }
                        break;
                    case 5:
                        System.out.println("Thank you for using our service."); break;
                    default:
                        System.out.println("Invalid choice.");
                }
            } while (choice != 5); sc.close();
        }
}
```