Book database

1) Create a class Books that contains four members: name, author, price and num_pages. Include a Constructor to set the values for the members. Include ~~mem~~ methods to set and get the details of the objects. Include a toString() method that could display the complete details of the book Develop a java program to create n books objects.

```
import java.util.Scanner;
public class books {
    String name;
    String author;
    int price, numpages;

    books (String name, String author, int price,
    int numpages) {
    this.name = name;
    this.author = author;
    this.price = price;
    this.numpages = numpages; }

    public String toString() {
      String name, author, price, numpages;
      name = "Book name:" + this.name + "\n";
      author = "author name:" + this.author + "\n";
      price = "price:" + this.price + "\n";
      numpages = "number of pages:" + this.numpages + "\n";
      return name + author + price + numpages; }}
```

```
Class main {
public static void main (String args[]) {
 Scanner s= new Scanner (System.in);
 int n;
 String name, author;
 int price, numpages;
 System.out.println ("Enter the number of books:");
 n= s.nextInt();
 books b[];
 b= new books[n];
 for (int i=0; i<n ; i++) {
 System.out.println ("book"+ (i+1)+ ":");
 System.out.println ("enter name of the books:");
 name = s.next();
 System.out.println ("enter name of the author:");
 author = s.next();
 System.out.println ("enter price:");
 price = s.nextInt();
 System.out.println ("enter name of pages:");
 numpages = s.nextInt();
 b[i] = new books(name, author, price, numpages);
 for(i=0; i<n; i++) {
   System.out.println ("Book" + (i+1) + ": /n" +b[i]);
 } } }
}
```

Output: enter the no of book: 1
Book 1:
enter the number of the book: Jungle book
enter the author of the book: Rudyard
enter the price of the book: 1000
enter the number of pages of the book: 500

Book name: Jungle book
Author: Rudyard kipling
price: 1000
Number of pages: 500

BANK

```
import java.util.Scanner;

class Account {
    protected String name;
    protected int accno;
    protected double balance;

    public void get_info() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Name: ");
        name = sc.nextLine();
        System.out.print("Enter Account Number: ");
        accno = sc.nextInt();
    }

    public void deposit(double amount) {
        balance += amount;
        System.out.println("Amount deposited successfully.");
    }

    public void display() {
        System.out.println("Name: " + name);
        System.out.println("Account Number: " + accno);
        System.out.println("Balance: " + balance);
    }
}

class Cur_acct extends Account {
    private final double min_balance = 500;
    private final double penalty = 100;


    public void withdraw(double amount) {
        if (balance - amount >= min_balance) {
            balance -= amount;
            System.out.println("Amount withdrawn successfully.");
        } else {
            System.out.println("Insufficient balance for
withdrawal.");
        }
        check_min_balance();
    }

    private void check_min_balance() {
        if (balance < min_balance) {
```

```
            balance -= penalty;
            System.out.println("Penalty imposed for falling below
minimum balance.");
        }
    }
}

class Sav_acct extends Account {
    private final double interest_rate = 0.04;

    public void compute_interest() {
        double interest = balance * interest_rate;
        balance += interest;
        System.out.println("Interest credited successfully.");
    }
}

class Bank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter 1 for Current Account or 2 for
Savings Account: ");
        int choice = sc.nextInt();

        Account acc;
        if (choice == 1) {
            acc = new Cur_acct();
        } else {
            acc = new Sav_acct();
        }

        acc.get_info();

        while (true) {
            System.out.println("\nMenu:");
            System.out.println("1. Deposit");
            System.out.println("2. Withdraw");
            System.out.println("3. Display Balance");
            System.out.println("4. Compute Interest (Savings Account
only)");
            System.out.println("5. Exit");
            System.out.print("Enter your choice: ");
            int choice2 = sc.nextInt();

            switch (choice2) {
```

```
                case 1:
                    System.out.print("Enter amount to deposit: ");
                    double amount = sc.nextDouble();
                    acc.deposit(amount);
                    break;
                case 2:
                    if (acc instanceof Sav_acct) {
                        System.out.println("Withdrawal not allowed
for Savings Account.");
                    } else {
                        System.out.print("Enter amount to withdraw:
");
                        amount = sc.nextDouble();
                        ((Cur_acct) acc).withdraw(amount);
                    }
                    break;
                case 3:
                    acc.display();
                    break;
                case 4:
                    if (acc instanceof Sav_acct) {
                        ((Sav_acct) acc).compute_interest();
                    } else {
                        System.out.println("Interest computation not
applicable for Current Account.");
                    }
                    break;
                case 5:
                    System.exit(0);
                default:
                    System.out.println("Invalid choice.");
            }
        }
    }
}
```

OUTPUT :

```
Enter 1 for Current Account or 2 for Savings Account:
1
Enter Name: Clara
Enter Account Number: 1122334455

Menu:
```

BANK

```
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 1
Enter amount to deposit: 1000
Amount deposited successfully.

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 2
Enter amount to withdraw: 500
Amount withdrawn successfully.

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Name: Clara
Account Number: 1122334455
Balance: 500.0

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 5
PS C:\Users\ADMIN\Documents\CSE III\java prgms> cd
"c:\Users\ADMIN\Documents\CSE III\java prgms\" ; if ($?) { javac
Bank.java } ; if ($?) { java Bank }
Enter 1 for Current Account or 2 for Savings Account:
2
Enter Name: Rosy
Enter Account Number: 101202303

Menu:
```

BANK

```
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 1
Enter amount to deposit: 5000
Amount deposited successfully.

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 2
Withdrawal not allowed for Savings Account.

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 500
Invalid choice.

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 4
Interest credited successfully.

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 3
Name: Rosy
Account Number: 101202303
```

BANK

```
Balance: 5200.0

Menu:
1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings Account only)
5. Exit
Enter your choice: 5
```