1]Develop a Java program to demonstrate Banking System, demonstrate different features such deposit, withdrawal of the amount can be done with ease. Demonstrate for atleast 15 customers.

```
import java.util.*;

class BankAccount {
   long money = 0;
   int account_number;

   BankAccount(int acc) {
       this.account_number = acc;
   }

   BankAccount(int account_number, long money) {
       this.account_number = account_number;
       this.money = money;
   }

   void deposit(int depo) {
       this.money += depo;
       System.out.println(depo + " deposited, balance: " + money);
   }

   void withdraw(int depo) {
       if (money - depo < 0) {
           System.out.println("Insufficient balance");
           return;
       }
       this.money -= depo;
       System.out.println(depo + " withdrawn, balance: " + money);
   }
}

public class Bankmain {
   public static void main(String args[]) {
       Scanner input = new Scanner(System.in);
       int n;
       BankAccount[] accounts = new BankAccount[100];

       System.out.println("Welcome");
```

```
        System.out.println("Enter number of customers:");
        n = input.nextInt();

        for (int i = 0; i < n; i++) {
            System.out.println("Enter account number:");
            int acc = input.nextInt();
            System.out.println("Deposit money?(1/0):");
            int choice = input.nextInt();
            if (choice == 1) {
                System.out.println("Enter deposit amount:");
                long money = input.nextLong();
                accounts[i] = new BankAccount(acc, money);
            } else {
                accounts[i] = new BankAccount(acc);
            }
        }

        while (true) {
            System.out.println("Enter ur choice: 1)Deposit 2)Withdraw 3)Exit");
            int choice = input.nextInt();
            if (choice == 3) break;
            System.out.println("Enter account number:");
            int acc = input.nextInt();
            boolean found = false;
            for (int i = 0; i < n; i++) {
                if (accounts[i].account_number == acc) {
                    found = true;
                    if (choice == 1) {
                        System.out.println("Enter deposit amount:");
                        int money = input.nextInt();
                        accounts[i].deposit(money);
                    } else {
                        System.out.println("Enter withdrawal amount:");
                        int money = input.nextInt();
                        accounts[i].withdraw(money);
                    }
                    break;
                }
            }
            if (!found) System.out.println("Account not found");
        }
    }
```

```
}
op:
Welcome
Enter number of customers:
2
Enter account number:
101
Deposit money?(1/0):
1
Enter deposit amount:
5000
Enter account number:
102
Deposit money?(1/0):
0
Enter your choice: 1)Deposit 2)Withdraw 3)Exit
1
Enter account number:
101
Enter deposit amount:
2000 deposited, balance: 7000
Enter your choice: 1)Deposit 2)Withdraw 3)Exit
2
Enter account number:
102
Enter withdrawal amount:
1000
Insufficient balance
Enter your choice: 1)Deposit 2)Withdraw 3)Exit
2
Enter account number:
101
Enter withdrawal amount:
3000 withdrawn, balance: 4000
Enter your choice: 1)Deposit 2)Withdraw 3)Exit
3
```

2]Demonstrate Static variables and methods example

```
class Counter {
   static int ct= 0;

   static void increment() {
       ct++;
   }

   static void displayCount() {
       System.out.println("Count:" + ct);
   }
}

public class StaticExample {
   public static void main(String[] args) {
       Counter.increment();
       Counter.increment();
       Counter.displayCount();
   }
}
op:
Count:2



3]Demonstrate Inner classes.

public class Outer{
   public String of = "offff";

   public class Inner{
       public void display() {
           System.out.println(of);
       }
   }
}

public class InnerClass{
   public static void main(String[] args) {
       Outer outer = new Outer();
       Outer.Inner inner = outer.new Inner();
       inner.display();
   }
```

```
}
op:offff


4]Write a Java program to Method and constructor Overloading
class Overload {
   Overload() {
       System.out.println("No parameters");
   }
   Overload(int a) {
       System.out.println("One parameter: " + a);
   }
   Overload(int a, int b) {
       System.out.println("Two parameters: " + a + ", " + b);
   }
   void display() {
       System.out.println("No parameters");
   }
   void display(int a) {
       System.out.println("One parameter: " + a);
   }
   void display(int a, int b) {
       System.out.println("Two parameters: " + a + ", " + b);
   }
}
public class Overloading{
   public static void main(String[] args) {
       Overload obj1 = new Overload();
       Overload obj2 = new Overload(10);
       Overload obj3 = new Overload(20, 30);

       obj1.display();
       obj2.display(100);
       obj3.display(200, 300);
   }
}
op:
No parameters
One parameter: 10
Two parameters: 20, 30
No parameters in display
One parameter in display: 100
```

```
Two parameters in display: 200, 300
```

5]Write a Java program to Implement single inheritance using Box Example

```
class Box {
   int width;
   int height;

   Box(int w, int h) {
       width = w;
       height = h;
   }
   void display() {
       System.out.println("Width: " + width + ", Height: " + height);
   }
}
class Boxweight extends Box {
   int weight;

   Boxweight(int w, int h, int wt) {
       super(w, h);
       weight = wt;
   }
   void displayWeight() {
       System.out.println("Weight: " + weight);
   }
}
public class Inheritance{
   public static void main(String[] args) {
       BoxWeight box = new Boxweight(10, 20, 30);
       box.display();
       box.displayWeight();
   }
}
op:
Width: 10, Height: 20
Weight: 30
```