Java Assignment -1

1. Write a Java menu driven program using switch case to perform all Arithmetic operations on two Integers.

```
import java.util.*;


class q1 {
   public static void main(String args[]) {
      Scanner sc = new Scanner(System.in);
      int a;
      int b;
      int c;
      int ch;
      System.out.println("Enter Value for a : ");
      a = sc.nextInt();
      System.out.println("Enter Value for b : ");
      b = sc.nextInt();
      do {
         System.out.println(
               "Enter your choice : \n1.Addition \n2.Subtraction \n3.Multiplication \n4.Division
\n5.Modulation \n6.Exit");
         ch = sc.nextInt();


         switch (ch) {
            case 1:
               c = a + b;
```

```
                System.out.println("Addition is :" + c);
                break;
            case 2:
                c = a - b;
                System.out.println("Subtraction is :" + c);
                break;
            case 3:
                c = a * b;
                System.out.println("Multiplication is :" + c);
                break;
            case 4:
                c = a / b;
                System.out.println("Division is :" + c);
                break;
            case 5:
                c = a % b;
                System.out.println("Modulation is :" + c);
                break;
            default:
                System.out.println("Invalid choice ...");
                break;
          }
      } while (ch != 6);

   }

}
```

Output :

Enter Value for a :

10

Enter Value for b :

20

Enter your choice :

1.Addition

2.Subtraction

3.Multiplication

4.Division

5.Modulation

6.Exit

1

Addition is :30

Enter your choice :

1.Addition

2.Subtraction

3.Multiplication

4.Division

5.Modulation

6.Exit

2

Subtraction is :-10

Enter your choice :

1.Addition

2.Subtraction

3.Multiplication

4.Division

5.Modulation

6.Exit

```
3
Multiplication is :200
Enter your choice :
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulation
6.Exit
6
Program Terminated ..
```

2.Write a Java program to demonstrate local, instance and static variables.

```
import java.util.*;

public class q2 {
   int iv = 10;
   static int sv = 20;

   public void showVars() {
      int lv = 30;
      System.out.println("Local: " + lv);
      System.out.println("Instance: " + iv);
      System.out.println("Static: " + sv);
   }

   public static void main(String[] args) {
      q2 demo = new q2();
      demo.showVars();
```

```
        System.out.println("Static via Class: " + q2.sv);
    }
}
```

Output :

```
Local: 30
Instance: 10
Static: 20
Static via Class: 20
```

3. Write a program to print odd numbers between 50 to 100 in reverse order.

```
import java.util.*;

class q3 {
    public static void main(String args[]) {
        int i = 99;
        System.out.println("Odd numbers between 50 to 100 in reverse order are : \n");
        while (i >= 50) {

            System.out.println(i);
            i = i - 2;

        }
    }
}
```

Output :

Odd numbers between 50 to 100 in reverse order are :

99 97 95 93 91 89 87 85 83 81 79 77 75 73 71 69 67 65 63 61 59 57 55 53 51

4. Write a Java program which prints given number is prime or not.

```
import java.util.Scanner;

class q4 {
   public static void main(String args[]) {
      Scanner sc = new Scanner(System.in);

      System.out.print("Enter a number: ");
      int num = sc.nextInt();

      boolean a = true;

      if (num <= 1) {
         a = false;
      } else {
         for (int i = 2; i < num; i++) {
            if (num % i == 0) {
               a = false;
               break;
            }
         }
      }

      if (a) {
         System.out.println(num + " is a prime number.");
```

```
        } else {
            System.out.println(num + " is not a prime number.");
        }
    }
}
```

Output :

```
Enter a number: 10
10 is not a prime number.
```

5. Write a Java program which accepts consumed units of electricity And Calculate the bill according to following table.

Unit range	Rate in Rs.
1 to 100	10 per unit
101 to 200	15 per unit
201 to 300	20 per unit
Above 300	25 per unit

```
import java.util.Scanner;

class q5 {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the consumed units of electricity: ");
        int units = sc.nextInt();

        double billAmount = 0;
```

```
        if (units <= 100) {
            billAmount = units * 10;
        } else if (units <= 200) {
            billAmount = 100 * 10 + (units - 100) * 15;
        } else if (units <= 300) {
            billAmount = 100 * 10 + 100 * 15 + (units - 200) * 20;
        } else {
            billAmount = 100 * 10 + 100 * 15 + 100 * 20 + (units - 300) * 25;
        }

        System.out.println("The total electricity bill is: Rs. " + billAmount);

        sc.close();
    }
}
```

Output :

```
Enter the consumed units of electricity: 102
The total electricity bill is: Rs. 1030.0
```

6 . Write a Java program to which input elements from user for 1D array and display it.

```
import java.util.*;

class one {
    Scanner sc = new Scanner(System.in);
```

```
    int[] x = new int[3];
    int i;

    void show() {
        System.out.println("Enter 3 Elements:");
        for (i = 0; i < 3; i++) {
            x[i] = sc.nextInt();
        }

        System.out.println("Array Elements are:");
        for (i = 0; i < 3; i++) {
            System.out.println("\t" + x[i]);
        }
    }
}

class q6 {
    public static void main(String args[]) {
        one p = new one();
        p.show();
    }
}
```

Output :

```
Enter 3 Elements:
10
20
30
Array Elements are:
```

```
10
20
30
```

7. Write a Java program to implement jagged array

```
import java.util.Scanner;

class q7 {
   public static void main(String[] args) {
      int[][] jArr = { { 1, 2 }, { 3, 4, 5 }, { 6 } };

      for (int[] r : jArr) {
         for (int n : r) {
            System.out.print(n + " ");
         }
         System.out.println();
      }
   }
}
```

Output :

```
1 2
3 4 5
6
```

8. Write a Java program to implement multidimensional array.

```
import java.util.*;

class one {
   Scanner sc = new Scanner(System.in);
   int[][][] x = new int[2][2][2];
   int i, j, k;

   void show() {
      System.out.println("Enter Elements:");
      for (i = 0; i < 2; i++) {
         for (j = 0; j < 2; j++) {
            for (k = 0; k < 2; k++) {
               x[i][j][k] = sc.nextInt();
            }
         }

      }

      System.out.println("Array :");
      for (i = 0; i < 2; i++) {
         for (j = 0; j < 2; j++) {
            for (k = 0; k < 2; k++) {
               System.out.print(" " + x[i][j][k]);
            }
         }
```

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

        }

    }
}

class q8 {
    public static void main(String args[]) {
        one p = new one();
        p.show();
    }
}
```

Output :

```
Enter Elements:
1
2
3
4
5
6
7
8
Array :
 1 2 3 4
 5 6 7 8
```

9. Write a Java program to find length of array.

```
import java.util.*;

class q9 {
   public static void main(String args[]) {
      Scanner sc = new Scanner(System.in);
      int x[] = new int[5];
      int i;
      System.out.println("Enter Elements :");
      for (i = 0; i < 5; i++) {
         x[i] = sc.nextInt();
      }
      System.out.println("Size of array is : " + x.length);
   }

}
```

Output :

```
Enter Elements :
1
2
3
4
5
```

Size of array is : 5

10. Write a java class to accept student details like roll no., name and address and display it.

```
import java.util.*;

class one {
   Scanner sc = new Scanner(System.in);
   int roll, age;
   String name, address;

   void get() {
      System.out.println("Enter Student Name : ");
      name = sc.next();
      System.out.println("Enter Student Roll no : ");
      roll = sc.nextInt();
      System.out.println("Enter Student Age : ");
      age = sc.nextInt();
      System.out.println("Enter Student Address : ");
      address = sc.next();

   }

   void show() {
      System.out.println("Student Name : " + name);
      System.out.println("Student Roll no : " + roll);
      System.out.println("Student Age : " + age);
      System.out.println("Student Address : " + address);
```

```
    }
}

class q10 {
    public static void main(String args[]) {
        one obj = new one();
        obj.get();
        obj.show();
    }
}
```

Output :

```
Enter Student Name :
abc
Enter Student Roll no :
101
Enter Student Age :
20
Enter Student Address :
sangola
Student Name : abc
Student Roll no : 101
Student Age : 20
Student Address : sangola
```

11. Write a Java program which accepts emp id and emp name, use them as actual parameter and send them to the constructor, after that display the record from constructor.

```
import java.util.*;

class one {
   int eid;
   String ename;

   one(int id, String name) {
      eid = id;
      ename = name;
   }

   void show() {
      System.out.println("Enp id : " + eid);
      System.out.println("Enp name : " + ename);
   }
}

class q11 {
   public static void main(String args[]) {
      int id;
      String name;
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter Emp id : ");
      id = sc.nextInt();
      System.out.println("Enter Emp name : ");
      name = sc.next();
      one obj = new one(id, name);
      obj.show();
```

```
    }
}
```

Output :

```
Enter Emp id :
100
Enter Emp name :
abc
Enp id : 100
Enp name : abc
```

12. Write a Java program of constructor overloading.

```
import java.util.*;

class q12 {
    int id;
    String name;

    q12() {
        id = 101;
        name = "student";
    }

    q12(int eid, String ename) {
        id = eid;
        name = ename;
```

```
    }

    void show() {
        System.out.println("Emp id : " + id);
        System.out.println("Emp name : " + name);

    }

    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        int id;
        String name;
        System.out.println("Enter Emp id : ");
        id = sc.nextInt();
        System.out.println("Enter Emp Name : ");
        name = sc.next();
        q12 p = new q12();
        q12 q = new q12(id, name);
        p.show();
        q.show();
    }
}
```

Output :

```
Enter Emp id :
100
Enter Emp Name :
abc
```

Emp id : 101

Emp name : student

Emp id : 100

Emp name : abc

13. Write a Java program which refers current executing object.

a)inside the constructor. b) inside the member function.

```
class q13 {
   String name;

   q13(String n) {
      name = n;
      System.out.println("Constructor called");
   }

   void display() {
      System.out.println("Name: " + name);
   }

   public static void main(String[] args) {
      q13 p = new q13("sangola");
      p.display();
   }
}
```

Output :

Constructor called

Name: sangola

14.Write a Java program to implement static method and static block

```
class q14 {
    static int n;

    static {
        n = 10;
        System.out.println("Static block executed.");
    }

    static void show() {
        System.out.println("Value of n: " + n);
    }

    public static void main(String[] args) {
        System.out.println("Main method started.");
        q14.show();
    }
}
```

Output :

Static block executed.

Main method started.

Value of n: 10

15.Write a java program to demonstrate user defined package

(Will be Updated in future …)

16.Write a java program to implements all types of inheritance.

```
//single inheritance

import java.util.*;

class A {
   Scanner sc = new Scanner(System.in);
   int a, b;

   void get() {
      System.out.println("Enter First Number : ");
      a = sc.nextInt();
      System.out.println("Enter Second Number : ");
      b = sc.nextInt();
   }
}

class B extends A {
   int c;

   void add() {
      c = a + b;
      System.out.println("Addition is : " + c);
```

```
    }
}

class single {
    public static void main(String args[]) {
        B a = new B();
        a.get();
        a.add();
    }
}
```

Output :

```
Enter First Number :
10
Enter Second Number :
20
Addition is : 30
```

// multilevel inheritance

```
import java.util.Scanner;

class A {
    Scanner sc = new Scanner(System.in);
    int a;

    void first() {
```

```
        System.out.println("Enter First Number : ");
        a = sc.nextInt();
    }
}

class B extends A {
    Scanner sc = new Scanner(System.in);
    int b;

    void second() {
        System.out.println("Enter First Number : ");
        b = sc.nextInt();
    }
}

class C extends B {
    int c;

    void add() {
        c = a + b;
        System.out.println("Addition is : " + c);
    }
}

class multilevel {
    public static void main(String args[]) {
        C a = new C();
        a.first();
        a.second();
```

```
        a.add();
    }
}
```

Output :

```
Enter First Number :
43
Enter First Number :
53
Addition is : 96
```

// hierarchical inheritance

```
import java.util.*;

class A {
    Scanner sc = new Scanner(System.in);
    int a, b;

    void get() {
        System.out.println("Enter First Number : ");
        a = sc.nextInt();

        System.out.println("Enter Second Number : ");
        b = sc.nextInt();
    }
}

class B extends A {
```

```
    int c;

    void add() {
        c = a + b;
        System.out.println("Addition is : " + c);
    }
}

class C extends A {
    int c;

    void sub() {
        c = a - b;
        System.out.println("Subtraction is : " + c);
    }
}

class hierarchical {
    public static void main(String args[]) {
        B obj1 = new B();
        obj1.get();
        obj1.add();

        C obj = new C();
        obj.get();
        obj.sub();

    }
}
```

Output :

```
Enter First Number :
34
Enter Second Number :
23
Addition is : 57
Enter First Number :
10
Enter Second Number :
2
Subtraction is : 8
```

```
// hybrid inheritance

class A {
   int a = 10;
   int b = 5;
}

class B extends A {
   void sub() {
      int c = a - b;
      System.out.println("subtraction is=" + c);
   }
}

class C extends A {
   int z = 5;
```

```
}

class D extends C {
   void add() {
      int q = a + b + z;
      System.out.println("addition is=" + q);
   }
}

class hybrid {
   public static void main(String args[]) {
      B m = new B();
      m.sub();
      D n = new D();
      n.add();
   }
}
```

Output :

```
subtraction is=5
addition is=20
```

// multiple inheritance

```
import java.util.*;

interface A {
   Scanner sc = new Scanner(System.in);
   int a = 10;
```

```
}

interface B {
   Scanner sc = new Scanner(System.in);
   int b = 20;
}

interface C {
   Scanner sc = new Scanner(System.in);
   int c = 30;
}

class D implements A, B, C {
   int d;

   public void add() {
      d = a + b + c;
      System.out.println("Addition is :" + d);
   }
}

class multiple {
   public static void main(String args[]) {
      D obj = new D();
      obj.add();
   }
}
```

Output :

Addition is :60

17.Write a java program to stop inheritance.

```
class A {
   void first() {
      System.out.println("First class");
   }
}

final class B {
   void second() {
      System.out.println("Second class");
   }
}

class q17 {
   public static void main(String args[]) {
      A a = new A();
      B b = new B();
      b.second();
      a.first();

   }
}
```

Output :

Second class

First class

18.Write a java program to override method.

```
class A {
   void show() {
      System.out.println("First class");
   }
}

class B extends A {
   void show() {
      System.out.println("Second class");
   }
}

class q18 {
   public static void main(String args[]) {
      A a = new A();
      B b = new B();
      a.show();
      b.show();

   }
}
```

Output :

First class

Second class

19.Write a java program to implement abstract class and abstract method

```
import java.util.*;

abstract class A {
   abstract void show(int num);
}

class B extends A {
   void show(int num) {
      System.out.println("Square is: " + (num * num));
   }
}

class C extends A {
   void show(int num) {
      System.out.println("Cube is: " + (num * num * num));
   }
}

public class q19 {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);

      System.out.print("Enter a number: ");
      int num = sc.nextInt();
```

```
        A obj = new B();
        A obj1 = new C();

        obj.show(num);
        obj1.show(num);
    }
}
```

Output :

```
Enter a number: 23
Square is: 529
Cube is: 12167
```

20.Write a java program to implement any interface.

```
import java.util.*;

interface A {
    void add();

    int a = 10;
}

interface B {
    void add();

    int a = 20;
```

```
}

interface C {
   void add();

   int a = 30;
}

class D implements A, B, C {
   public void add() {
      int c = A.a + B.a + C.a;
      System.out.println("Addition is : " + c);
   }
}

class q20 {
   public static void main(String args[]) {
      D obj = new D();
      obj.add();
   }
}
```

Output :

```
Addition is : 60
```

21.Write a java program to implement wrapper class.

```
import java.util.*;
```

```
class q21 {
   public static void main(String args[]) {
      Scanner sc = new Scanner(System.in);
      Integer roll;
      Character div = 'A';
      System.out.println("Enter Roll No  :");
      roll = sc.nextInt();
      System.out.println("Roll No is : " + roll);
      System.out.println("div is : " + div);
   }
}
```

Output :

```
Enter Roll No  :
101
Roll No is : 101
div is : A
```