

NAME: ZISHNENDU SARKER

ROLL: 2K19/CO/450

JAVA PROGRAMMING

LAB ASSIGNMENT 03

Group: A, G3

- **Java Program to Display Prime Numbers Between Intervals Using Function**

Code:

```
public class Prime {

    public static void main(String[] args) {

        int low = 20, high = 50;

        while (low < high) {
            if(checkPrimeNumber(low))
                System.out.print(low + " ");

            ++low;
        }
    }

    public static boolean checkPrimeNumber(int num) {
        boolean flag = true;

        for(int i = 2; i <= num/2; ++i) {

            if(num % i == 0) {
                flag = false;
                break;
            }
        }

        return flag;
    }
}
```

```
}
```

Output:

```
23 29 31 37 41 43 47
```

- **Java Program to Find Factorial of a Number Using Recursion**

Code:

```
public class Factorial {  
  
    public static void main(String[] args) {  
        int num = 6;  
        long factorial = multiplyNumbers(num);  
        System.out.println("Factorial of " + num + " = " + factorial);  
    }  
    public static long multiplyNumbers(int num)  
    {  
        if (num >= 1)  
            return num * multiplyNumbers(num - 1);  
        else  
            return 1;  
    }  
}
```

Output:

```
Factorial of 6 = 720
```

- **Java Program to Convert Binary Number to Decimal and vice-versa using functions**

Code:

```
class Main {
```

```

public static void main(String[] args) {

    long num = 110110111;

    int decimal = convertBinaryToDecimal(num);

    System.out.println("Binary to Decimal");
    System.out.println(num + " = " + decimal);
}

public static int convertBinaryToDecimal(long num) {
    int decimalNumber = 0, i = 0;
    long remainder;

    while (num != 0) {
        remainder = num % 10;
        num /= 10;
        decimalNumber += remainder * Math.pow(2, i);
        ++i;
    }

    return decimalNumber;
}

```

Output:

```
110110111 in binary = 439 in decimal
```

- **Java Program to Call One Constructor from another**

Code :

```

class Main {

    int sum;

```

```

Main() {
    this(5, 2);
}

Main(int arg1, int arg2) {
    this.sum = arg1 + arg2;
}

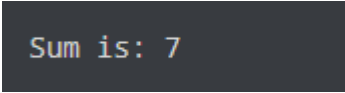
void display() {
    System.out.println("Sum is: " + sum);
}

public static void main(String[] args) {

    Main obj = new Main();
    obj.display();
}
}

```

Output:



```
Sum is: 7
```

- **Java Program to implement private constructors**

Code:

```

class Test {
    private Test () {
        System.out.println("This is a private constructor.");
    }
    public static void instanceMethod() {
        Test obj = new Test();
    }
}

class Main {

```

```
public static void main(String[] args) {  
    Test.instanceMethod();  
}  
}
```

Output:

```
This is a private constructor.
```

- **Java Program to pass method call as arguments to another method**

Code:

```
class Main {  
    public int add(int a, int b) {  
  
        int sum = a + b;  
        return sum;  
    }  
  
    public void square(int num) {  
        int result = num * num;  
        System.out.println(result); // prints 576  
    }  
    public static void main(String[] args) {  
  
        Main obj = new Main();  
        obj.square(obj.add(15, 9));  
  
    }  
}
```

Output:

```
576
```