AIM: Programs on Encapsulation. Write a program to demonstrate classes and objects

Program 1

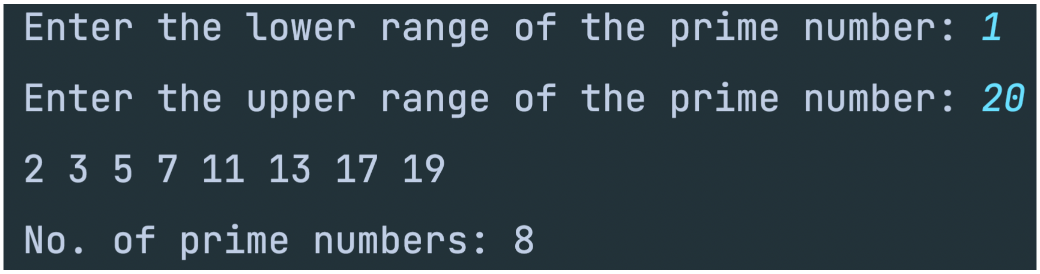
PROBLEM STATEMENT : Programs on Encapsulation. Write a program to demonstrate classes and objects

PROGRAM:

```java
import java.util.*;
public class prime {
    public static int FindPrime(int n) {
        if (n == 0 || n == 1) {
            return 0;
        }
        for (int i = 2; i < n; i++) {
            if (n % i == 0) {
                return 0;
            }
        }
        return 1;
    }
    public static void main(String[] args) {
        prime obj = new prime();
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the lower range of the prime number: ");
        int lower = scanner.nextInt();
        System.out.print("Enter the upper range of the prime number: ");
        int upper = scanner.nextInt();
        int count = 0;
        for (int i = lower; i <= upper; i++) {
            if (obj.FindPrime(i) == 1) {
                System.out.print(i + " ");
                count++;
            }
        }
        System.out.println("\nNo. of prime numbers: " + count);

    }
}
```

RESULT:



```
Enter the lower range of the prime number: 1

Enter the upper range of the prime number: 20

2 3 5 7 11 13 17 19

No. of prime numbers: 8
```

Program 2

PROBLEM STATEMENT:

A Mersenne prime is a prime number that has the form 2p−1 where p is a positive number greater than 1. Write a program that calculates candidate Mersenne primes 2p − 1 for 2≤p≤31. Then test the number to see if it is prime. If you detect that the number is prime, print out the number and the value of p.

PROGRAM:

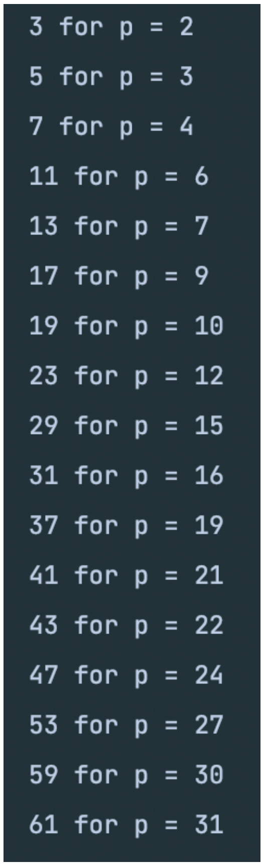
```
class merPrime{
    public static void main(String[] args){
        int  k;
        prime obj = new prime();

        for (k = 2; k <= 31; k++){

            if(obj.check_prime( 2*k-1)==1){
                System.out.print(2*k-1);
                System.out.println(" for p = " + k);
            }
        }
    }
}

class prime {
    int flag;
    int check_prime(int a) {
        for (int i = 1; i <= a; i++) {
            if (i == 1 || i == 0)
                continue;
            flag = 1;
            for (int j = 2; j <= i / 2; ++j) {
                if (i % j == 0) {
                    flag = 0;
                    break;
                }
            }
        }
        return flag;
    }
}
```

RESULT:



```
3 for p = 2
5 for p = 3
7 for p = 4
11 for p = 6
13 for p = 7
17 for p = 9
19 for p = 10
23 for p = 12
29 for p = 15
31 for p = 16
37 for p = 19
41 for p = 21
43 for p = 22
47 for p = 24
53 for p = 27
59 for p = 30
61 for p = 31
```

Program 3

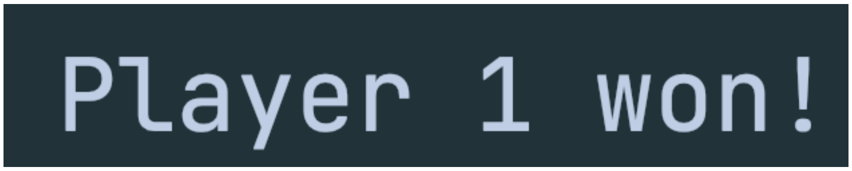
PROBLEM STATEMENT:

A random number is generated between 0-1000 which is the cash inflow to the person playing the game. Two people play the game 5 times and the winner is the one whose total cash is maximum and the program displays the winner. Set the initial amount as zero.
import java.lang.Math; and use Math.random()

PROGRAM:

```
import java.lang.Math;
class game{
    int number;
    int getnumber(){
        number=(int)(Math.random()*1000);
        return number;
    }
}
class file{
    public static void main(String[] args){
        game player1 = new game();
        game player2 = new game();
        int sum1=0,sum2=0,a,b;
        for(int i=0;i<5;i++){
            a=player1.getnumber();
            sum1+=a;
            b=player2.getnumber();
            sum2+=b;
        }
        if(sum1>sum2){
            System.out.println("Player 1 won!");
        }
        else{
            System.out.println("Player 2 won!");
        }
    }
}
```

RESULT:



```
Player 1 won!
```

Program 4

PROBLEM STATEMENT:

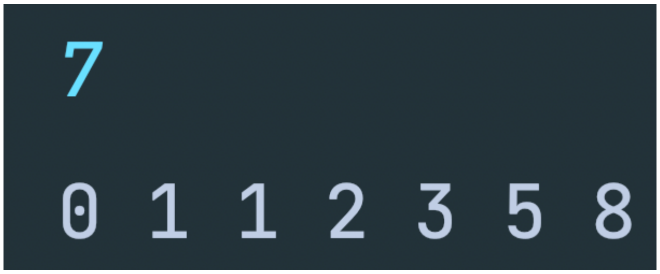
To write a java program to find fibonacci series up to entered integer number. Use set(int) to set the value of the number, use fibbo(int) method to calculate fibonacci next number, use display() to display fibonacci. Use display() calling fibbo method. Take the input from the user

PROGRAM:

```
import java.util.Scanner;
class Fibonacci_series{
    int number;
    void set(int n){
        number=n;
    }
    int fibbo(int nu){
        if (nu==1){
            return 0;
        }
        else if (nu==2){
            return 1;
        }
        else{
            return fibbo(nu-1)+fibbo(nu-2);
        }

    }
    void display(){
        for(int i=1;i<=number;i++){
            System.out.print(fibbo(i)+" ");
        }
    }
}
class file{
    public static void main(String[] args){
        Scanner sc =new Scanner(System.in);
        int num=sc.nextInt();
        Fibonacci_series f1 =new Fibonacci_series();
        f1.set(num);
        f1.display();
    }
}
```

RESULT:



```
7

0 1 1 2 3 5 8
```

CONCLUSION: In this experiment, we learned about the basic programs in java by using control flow statements and loop