

```
//code for prime numbers
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        int num = 29;
```

```
        boolean flag = false;
```

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

```
            // condition for nonprime number
```

```
            if (num % i == 0) {
```

```
                flag = true;
```

```
                break;
```

```
            }
```

```
        }
```

```
        if (!flag)
```

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

```
        else
```

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

```
    }
```

```
}
```

```
//code for checking whether a number can be expressed as sum of two prime number
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        int number = 34;
```

```
        boolean flag = false;
```

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

```
            // condition for i to be a prime number
```

```
            if (checkPrime(i)) {
```

```
                // condition for n-i to be a prime number
```

```
                if (checkPrime(number - i)) {
```

```
                    // n = primeNumber1 + primeNumber2
```

```
                    System.out.printf("%d = %d + %d\n", number, i, number - i);
```

```
                    flag = true;
```

```
                }
```

```
            }
```

```
        }
```

```
        if (!flag)
```

```
            System.out.println(number + " cannot be expressed as the sum of two prime numbers.");
```

```
    }
```

```
// Function to check prime number
```

```
static boolean checkPrime(int num) {
```

```
    boolean isPrime = true;
```

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

```
        if (num % i == 0) {
```

```
            isPrime = false;
```

```

        break;
    }
}

return isPrime;
}
}
sum of natural numbers using recursion
public class AddNumbers {

    public static void main(String[] args) {
        int number = 20;
        int sum = addNumbers(number);
        System.out.println("Sum = " + sum);
    }

    public static int addNumbers(int num) {
        if (num != 0)
            return num + addNumbers(num - 1);
        else
            return num;
    }
}

```

Example: Factorial of a Number Using Recursion

```

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;
    }
}

```