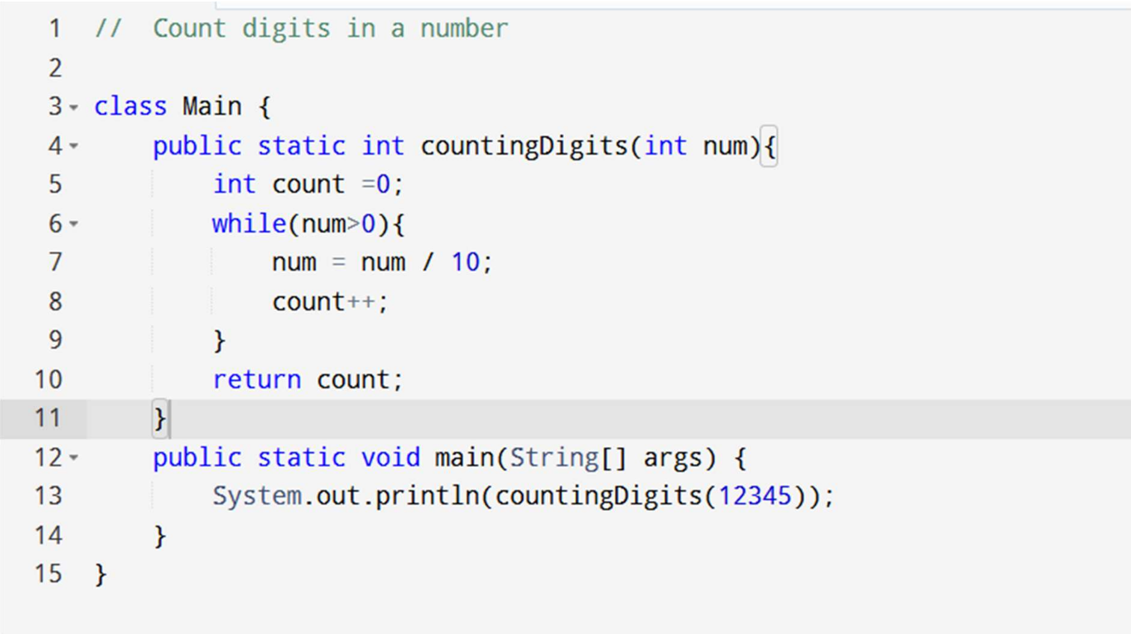
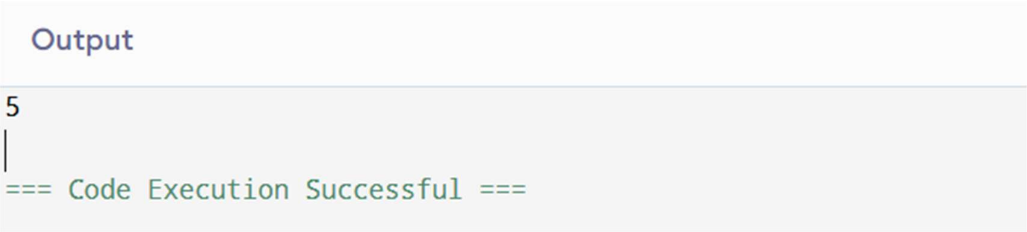
Assignment No 1
BASIC MATHS

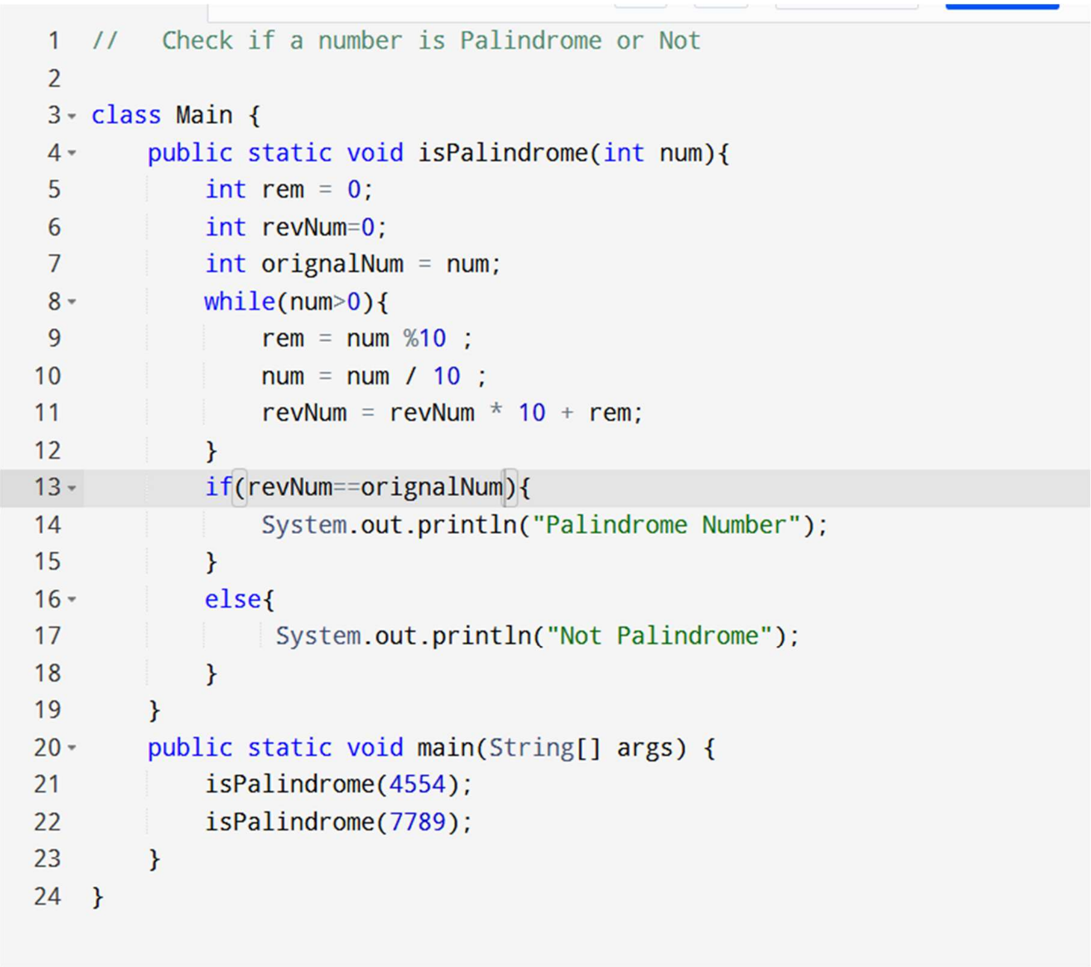


```java
//  Count digits in a number

class Main {
    public static int countingDigits(int num){
        int count =0;
        while(num>0){
            num = num / 10;
            count++;
        }
        return count;
    }
    public static void main(String[] args) {
        System.out.println(countingDigits(12345));
    }
}
```

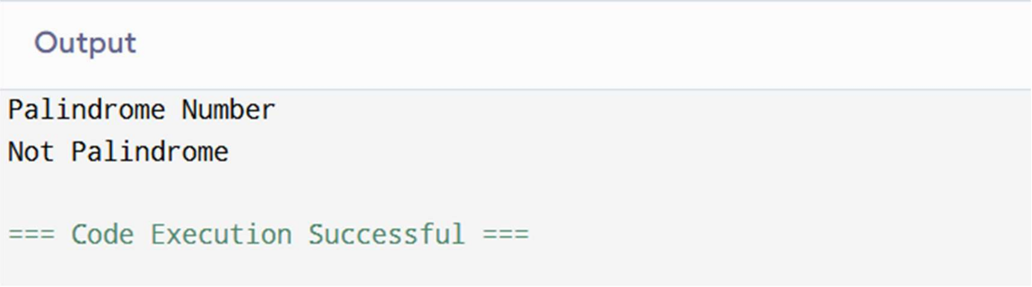


```
5

=== Code Execution Successful ===
```

Assignment No 1
BASIC MATHS



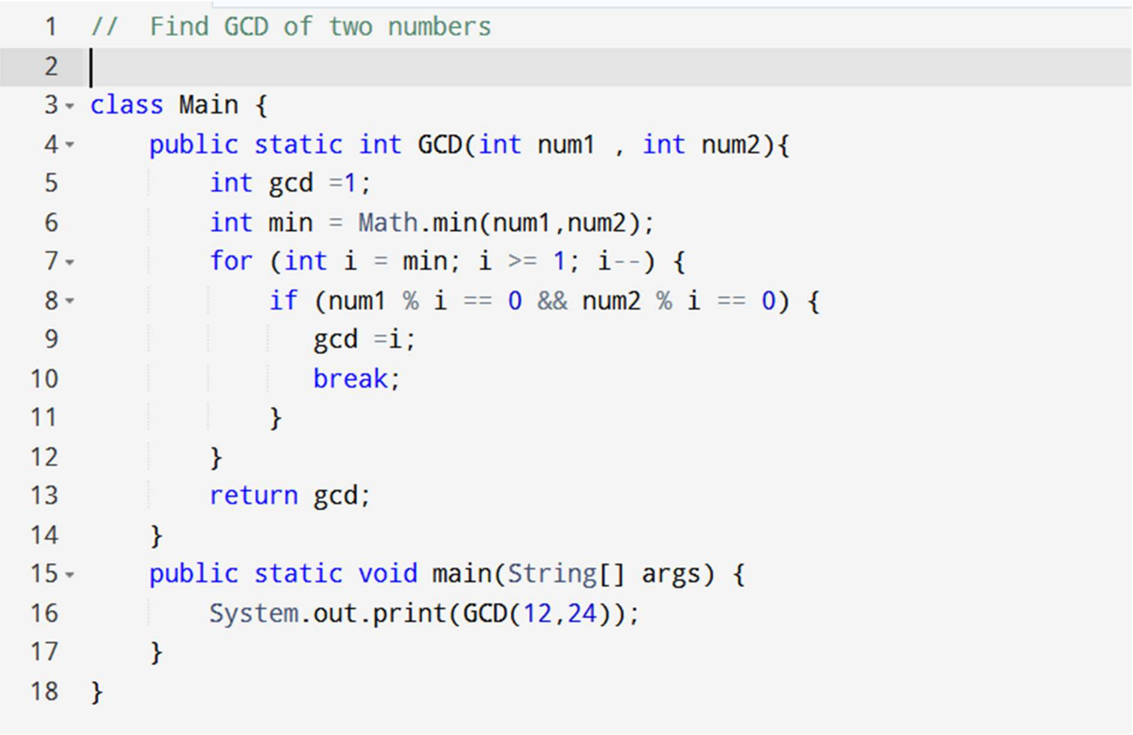
```java
//   Check if a number is Palindrome or Not

class Main {
    public static void isPalindrome(int num){
        int rem = 0;
        int revNum=0;
        int orignalNum = num;
        while(num>0){
            rem = num %10 ;
            num = num / 10 ;
            revNum = revNum * 10 + rem;
        }
        if(revNum==orignalNum){
            System.out.println("Palindrome Number");
        }
        else{
             System.out.println("Not Palindrome");
        }
    }
    public static void main(String[] args) {
        isPalindrome(4554);
        isPalindrome(7789);
    }
}
```



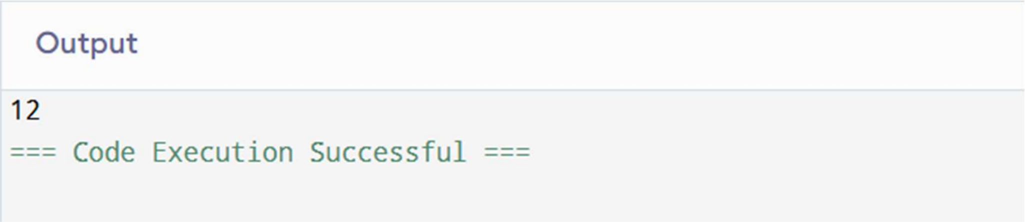
Output

```
Palindrome Number
Not Palindrome

=== Code Execution Successful ===
```

Assignment No 1
BASIC MATHS

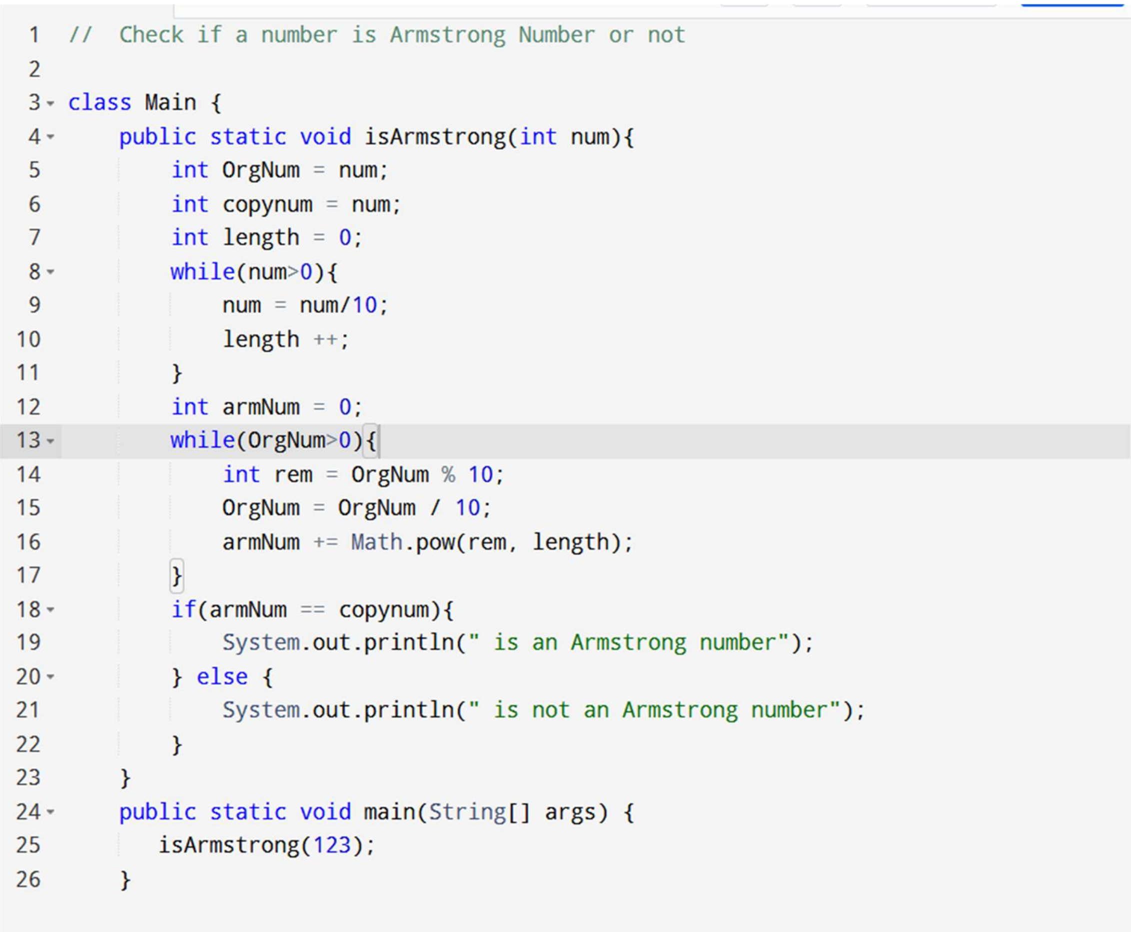


```java
//  Find GCD of two numbers

class Main {
    public static int GCD(int num1 , int num2){
        int gcd =1;
        int min = Math.min(num1,num2);
        for (int i = min; i >= 1; i--) {
            if (num1 % i == 0 && num2 % i == 0) {
                gcd =i;
                break;
            }
        }
        return gcd;
    }
    public static void main(String[] args) {
        System.out.print(GCD(12,24));
    }
}
```

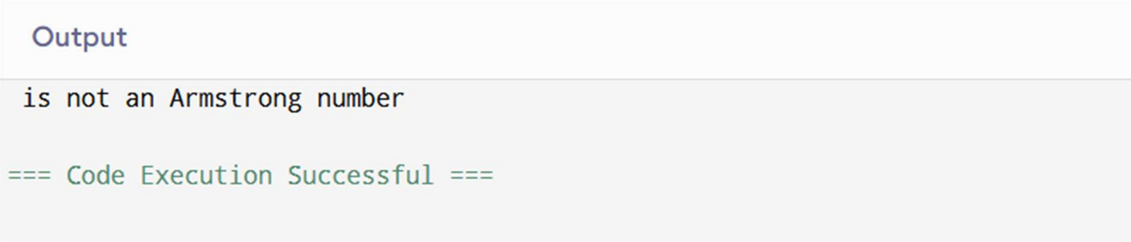


```
12
=== Code Execution Successful ===
```

Assignment No 1
BASIC MATHS



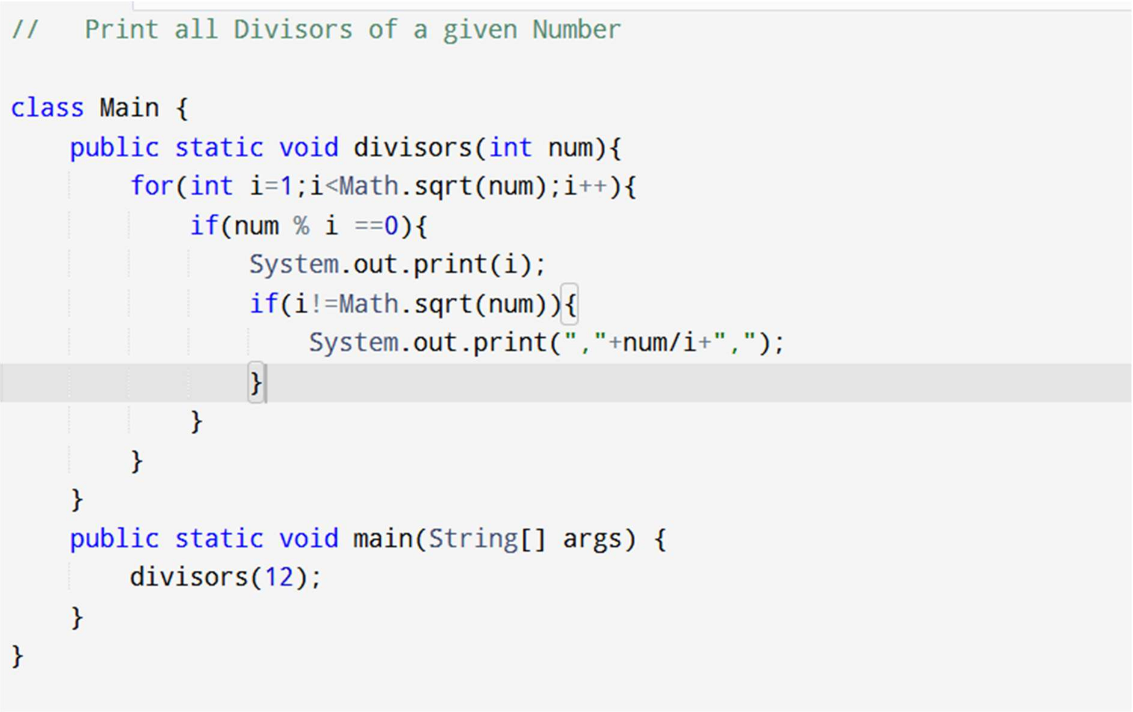
```
//  Check if a number is Armstrong Number or not

class Main {
    public static void isArmstrong(int num){
        int OrgNum = num;
        int copynum = num;
        int length = 0;
        while(num>0){
            num = num/10;
            length ++;
        }
        int armNum = 0;
        while(OrgNum>0){
            int rem = OrgNum % 10;
            OrgNum = OrgNum / 10;
            armNum += Math.pow(rem, length);
        }
        if(armNum == copynum){
            System.out.println(" is an Armstrong number");
        } else {
            System.out.println(" is not an Armstrong number");
        }
    }
    public static void main(String[] args) {
       isArmstrong(123);
    }
```



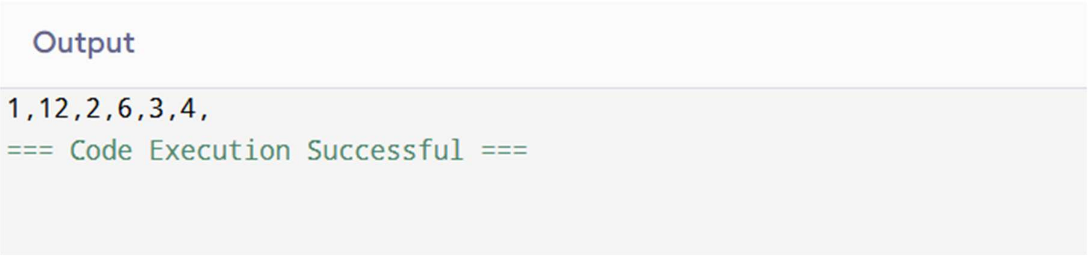
Output

```
 is not an Armstrong number

=== Code Execution Successful ===
```

Assignment No 1
BASIC MATHS



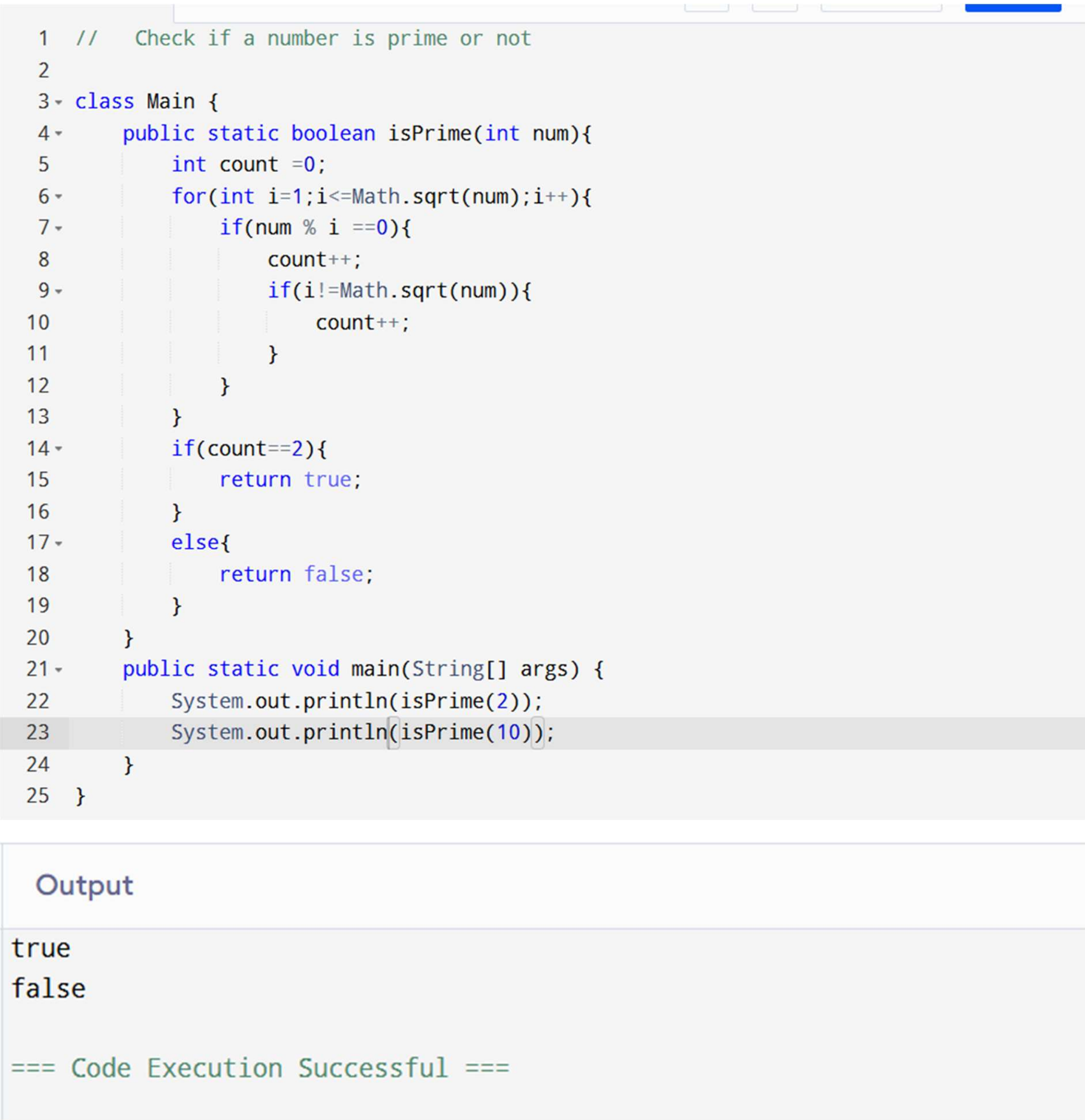
```java
//   Print all Divisors of a given Number

class Main {
    public static void divisors(int num){
        for(int i=1;i<Math.sqrt(num);i++){
            if(num % i ==0){
                System.out.print(i);
                if(i!=Math.sqrt(num)){
                    System.out.print(","+num/i+",");
                }
            }
        }
    }
    public static void main(String[] args) {
        divisors(12);
    }
}
```



Output

```
1,12,2,6,3,4,
=== Code Execution Successful ===
```

Assignment No 1
BASIC MATHS



```
//   Check if a number is prime or not

class Main {
    public static boolean isPrime(int num){
        int count =0;
        for(int i=1;i<=Math.sqrt(num);i++){
            if(num % i ==0){
                count++;
                if(i!=Math.sqrt(num)){
                    count++;
                }
            }
        }
        if(count==2){
            return true;
        }
        else{
            return false;
        }
    }
    public static void main(String[] args) {
        System.out.println(isPrime(2));
        System.out.println(isPrime(10));
    }
}
```

```
true
false

=== Code Execution Successful ===
```