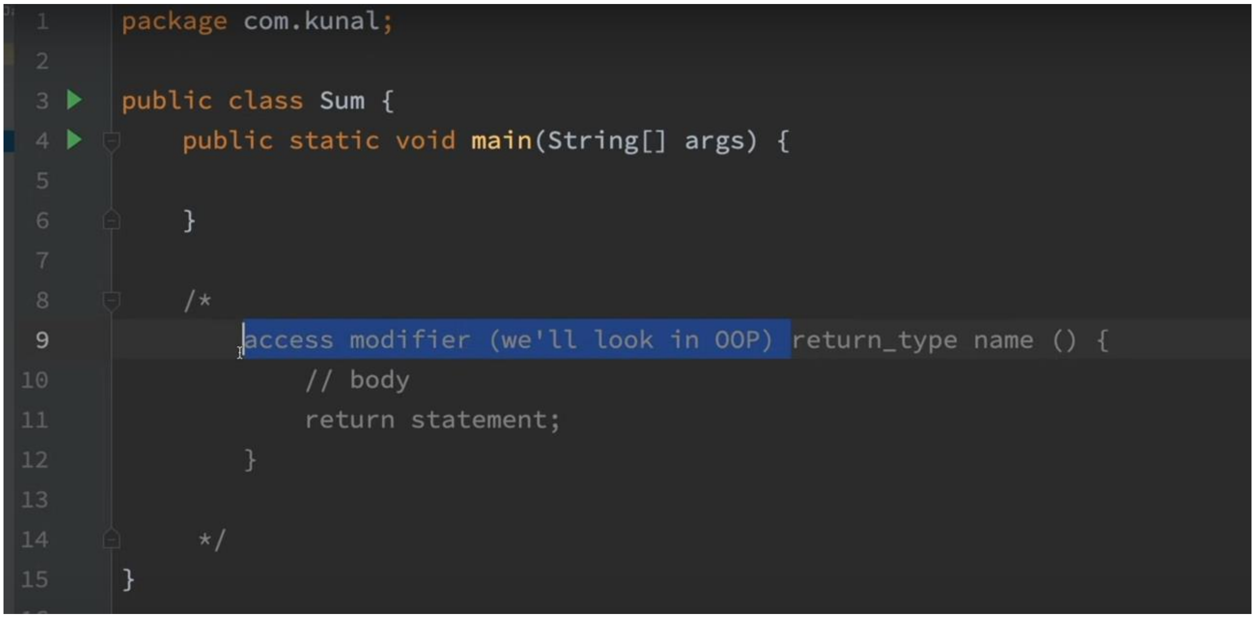
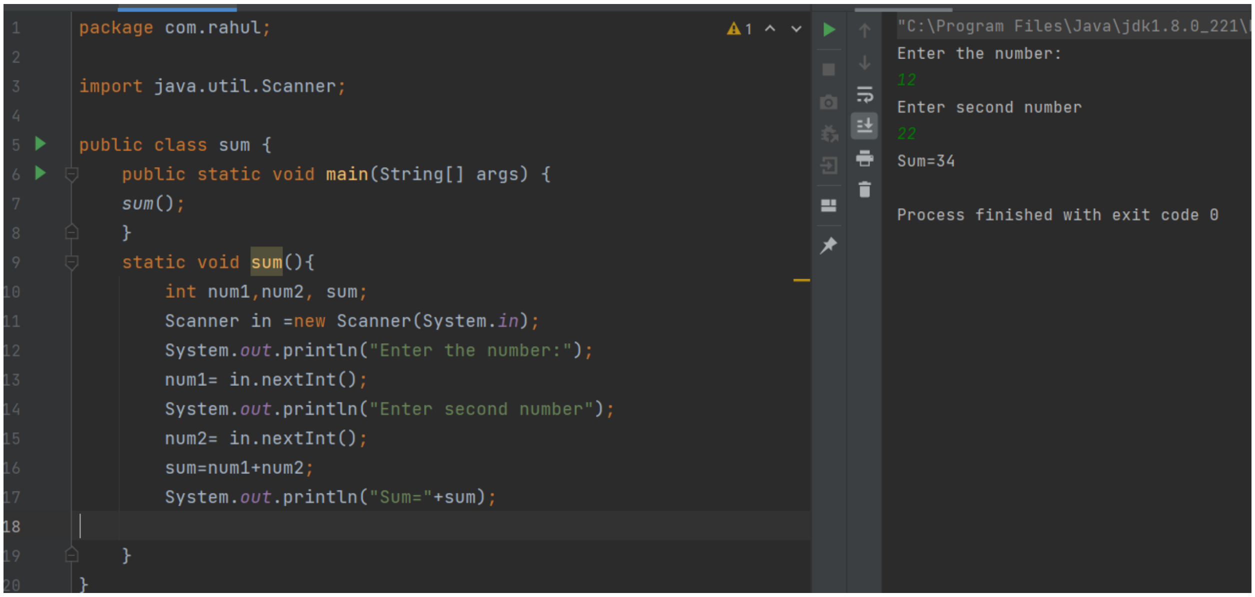
Function/Methods in Java

General syntax of methods



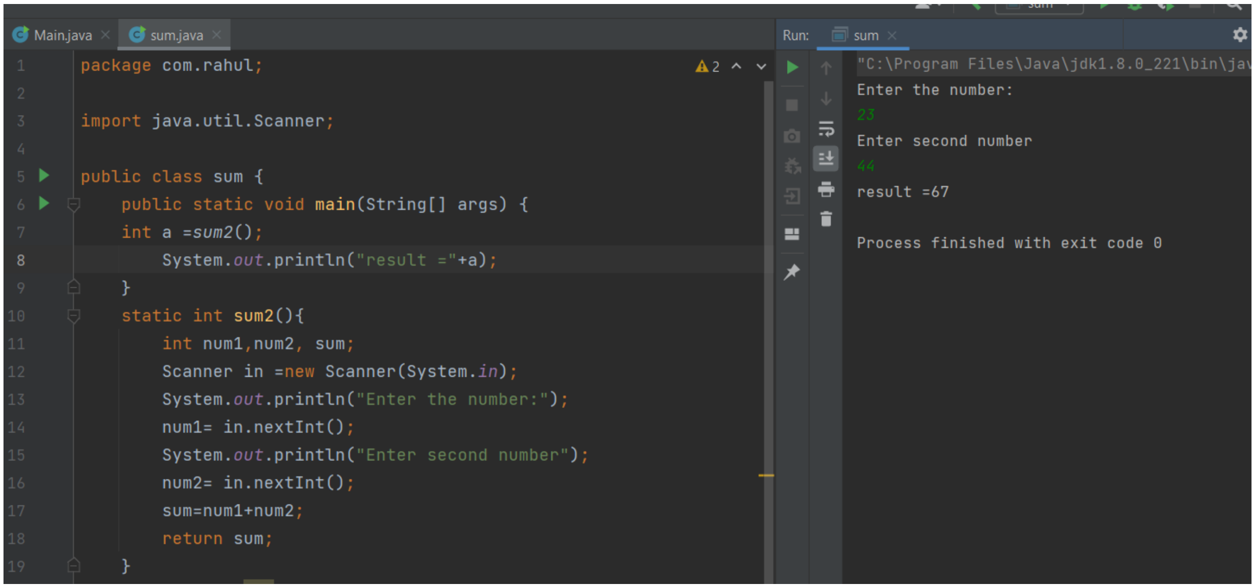
```java
package com.kunal;

public class Sum {
    public static void main(String[] args) {

    }

    /*
        access modifier (we'll look in OOP) return_type name () {
            // body
            return statement;
        }

     */
}
```



```java
package com.rahul;

import java.util.Scanner;

public class sum {
    public static void main(String[] args) {
    sum();
    }
    static void sum(){
        int num1,num2, sum;
        Scanner in =new Scanner(System.in);
        System.out.println("Enter the number:");
        num1= in.nextInt();
        System.out.println("Enter second number");
        num2= in.nextInt();
        sum=num1+num2;
        System.out.println("Sum="+sum);

    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_221\
Enter the number:
12
Enter second number
22
Sum=34

Process finished with exit code 0
```

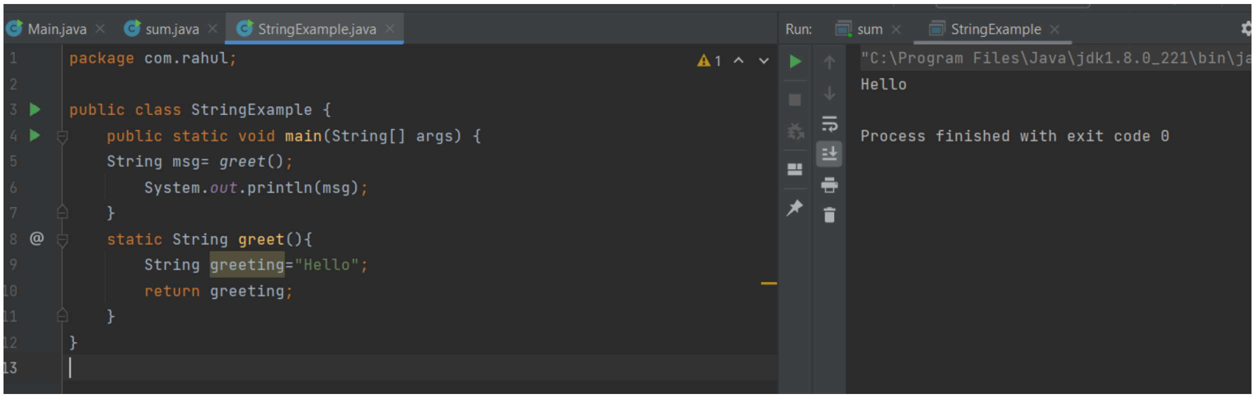
Return value



```java
package com.rahul;

import java.util.Scanner;

public class sum {
    public static void main(String[] args) {
    int a =sum2();
        System.out.println("result ="+a);
    }
    static int sum2(){
        int num1,num2, sum;
        Scanner in =new Scanner(System.in);
        System.out.println("Enter the number:");
        num1= in.nextInt();
        System.out.println("Enter second number");
        num2= in.nextInt();
        sum=num1+num2;
        return sum;
    }
```

```
"C:\Program Files\Java\jdk1.8.0_221\bin\ja
Enter the number:
23
Enter second number
44
result =67

Process finished with exit code 0
```

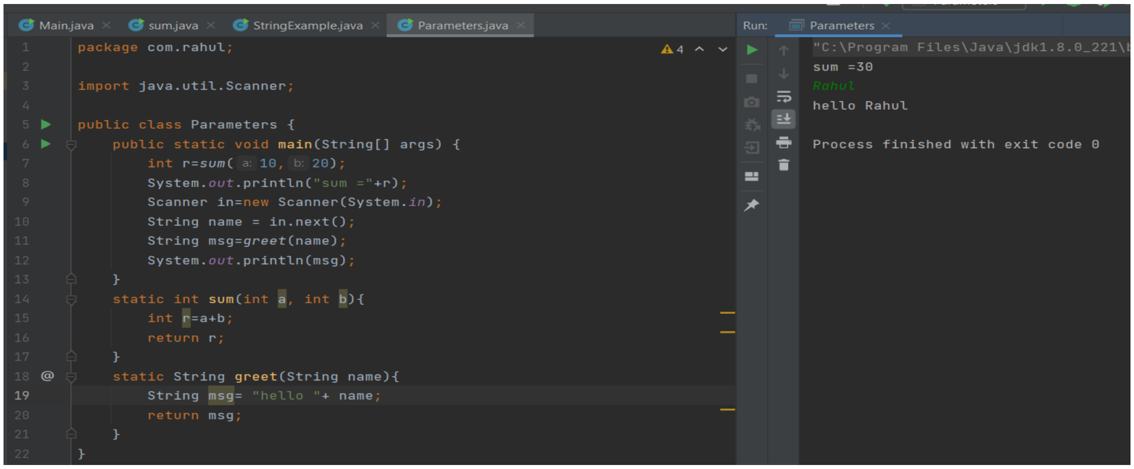
Return string



```java
package com.rahul;

public class StringExample {
    public static void main(String[] args) {
    String msg= greet();
        System.out.println(msg);
    }
    static String greet(){
        String greeting="Hello";
        return greeting;
    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_221\bin\ja
Hello

Process finished with exit code 0
```

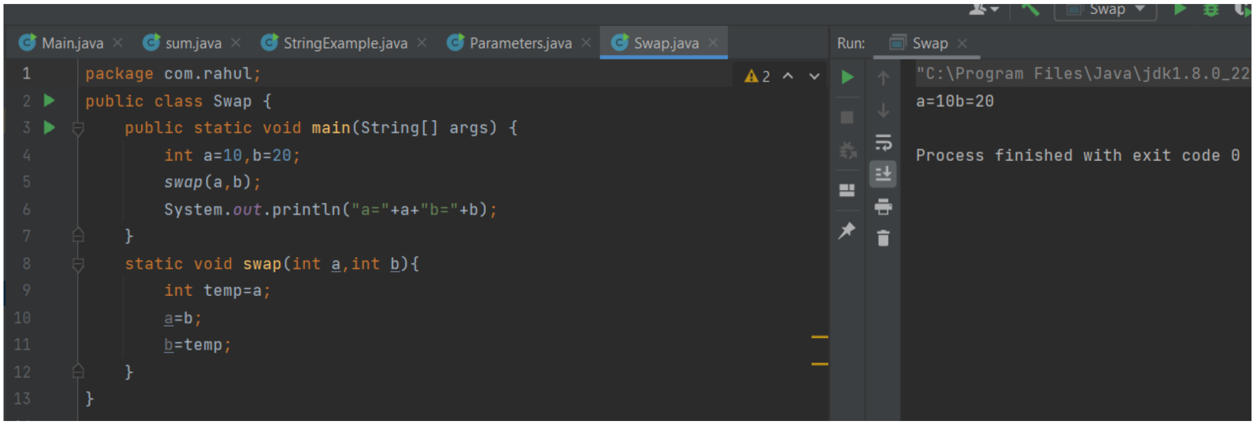
Parameters (integer function) & Parameters (String function)



```java
package com.rahul;

import java.util.Scanner;

public class Parameters {
    public static void main(String[] args) {
        int r=sum( a: 10, b: 20);
        System.out.println("sum ="+r);
        Scanner in=new Scanner(System.in);
        String name = in.next();
        String msg=greet(name);
        System.out.println(msg);
    }
    static int sum(int a, int b){
        int r=a+b;
        return r;
    }
    static String greet(String name){
        String msg= "hello "+ name;
        return msg;
    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_221\b
sum =30
Rahul
hello Rahul

Process finished with exit code 0
```

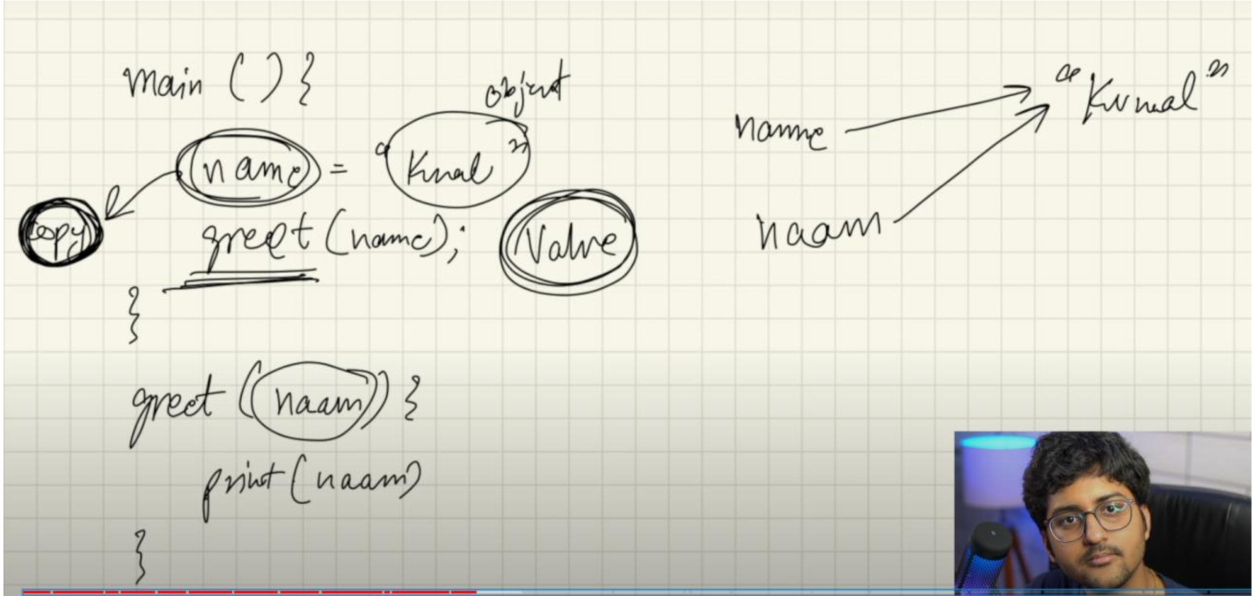
Swap Program



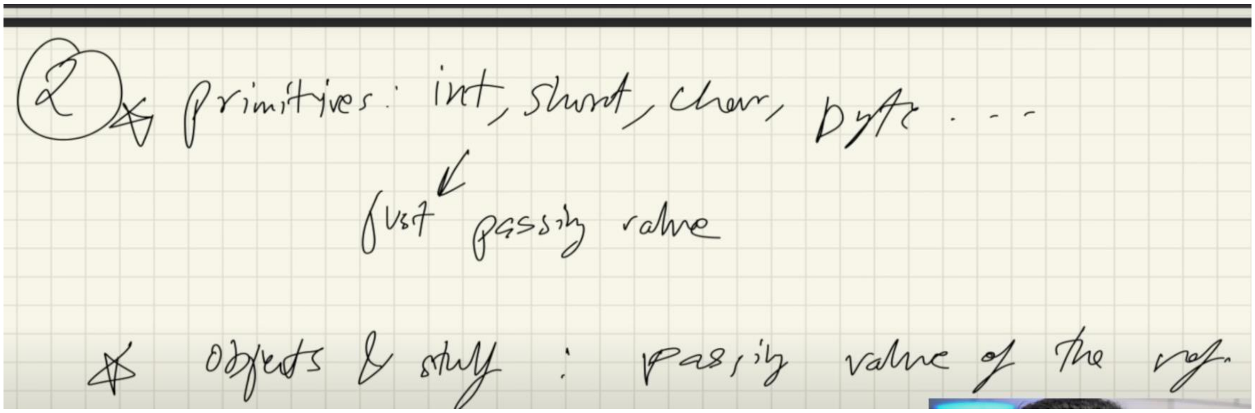
```
package com.rahul;
public class Swap {
    public static void main(String[] args) {
        int a=10,b=20;
        swap(a,b);
        System.out.println("a="+a+"b="+b);
    }
    static void swap(int a,int b){
        int temp=a;
        a=b;
        b=temp;
    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_22
a=10b=20

Process finished with exit code 0
```

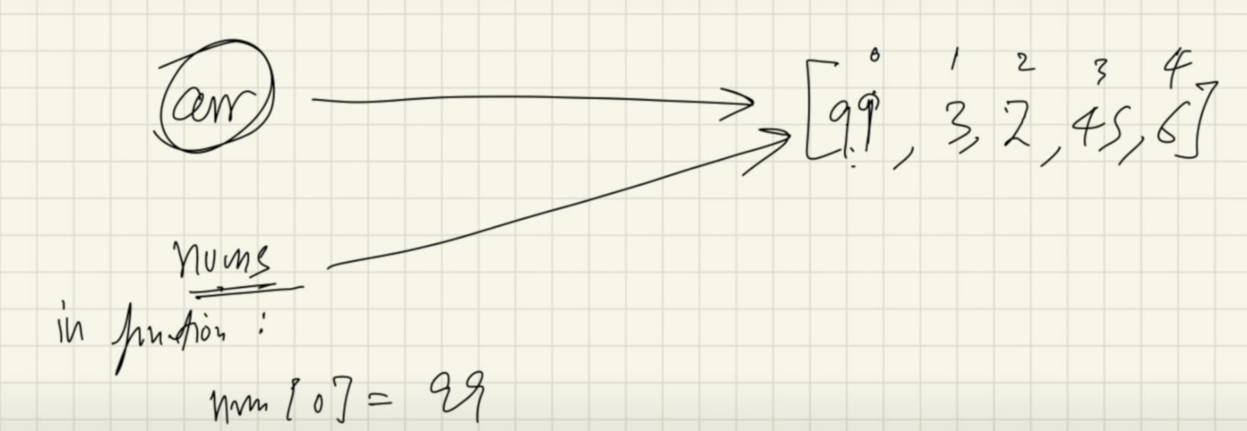
Here a and b value is not swaped.

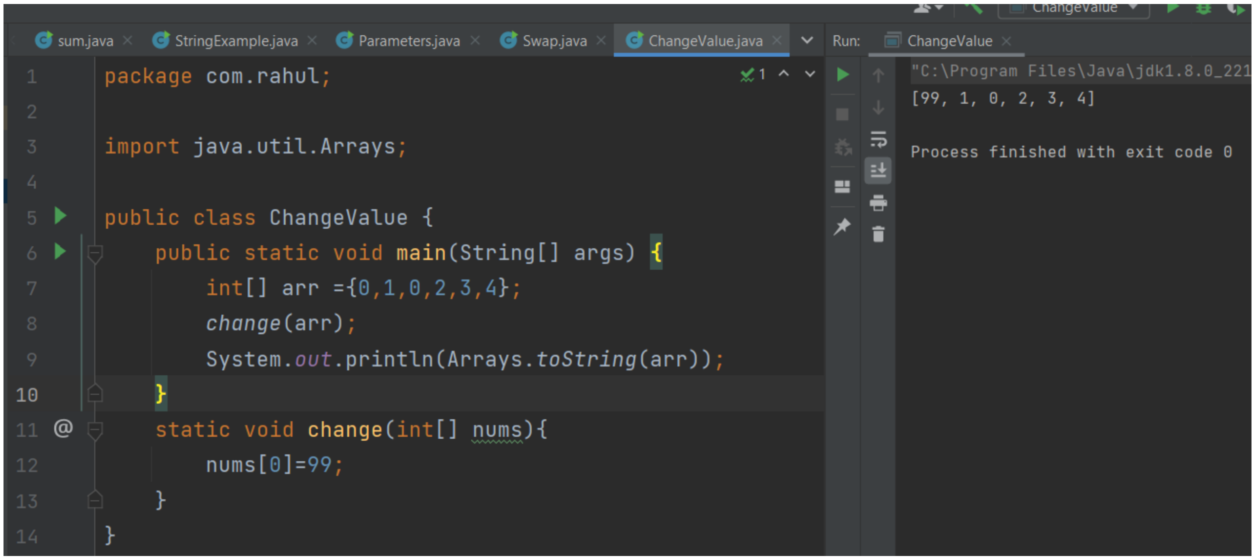


Java has return by value it does not has return by reference.


2
* Primitives : int, short, char, byte . . .
just passing value
* objects & stuff : passing value of the ref.

As array is an object so pass by value of reference happens here so during swap or change creted in function is reflected in the original array.

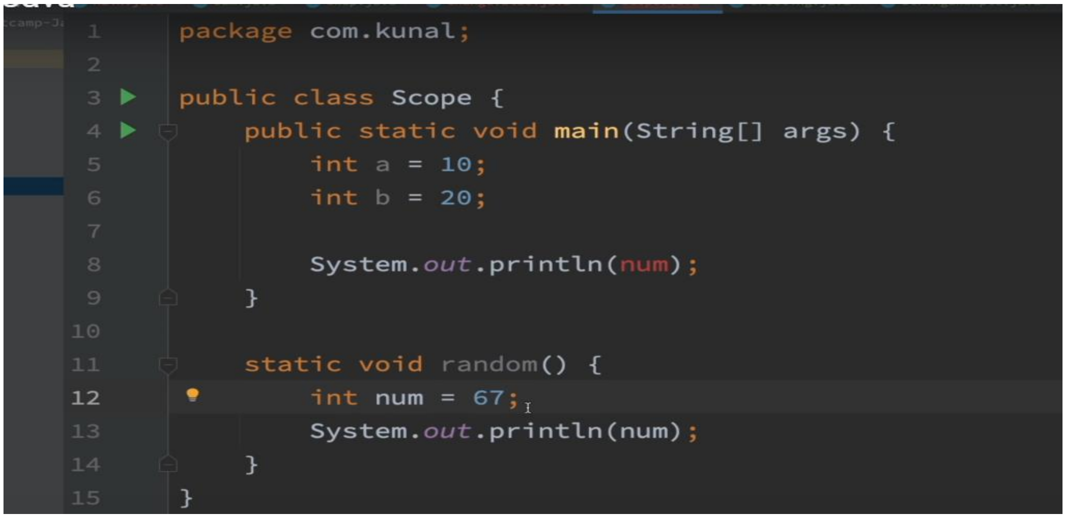




```java
package com.rahul;

import java.util.Arrays;

public class ChangeValue {
    public static void main(String[] args) {
        int[] arr ={0,1,0,2,3,4};
        change(arr);
        System.out.println(Arrays.toString(arr));
    }
    static void change(int[] nums){
        nums[0]=99;
    }
}
```

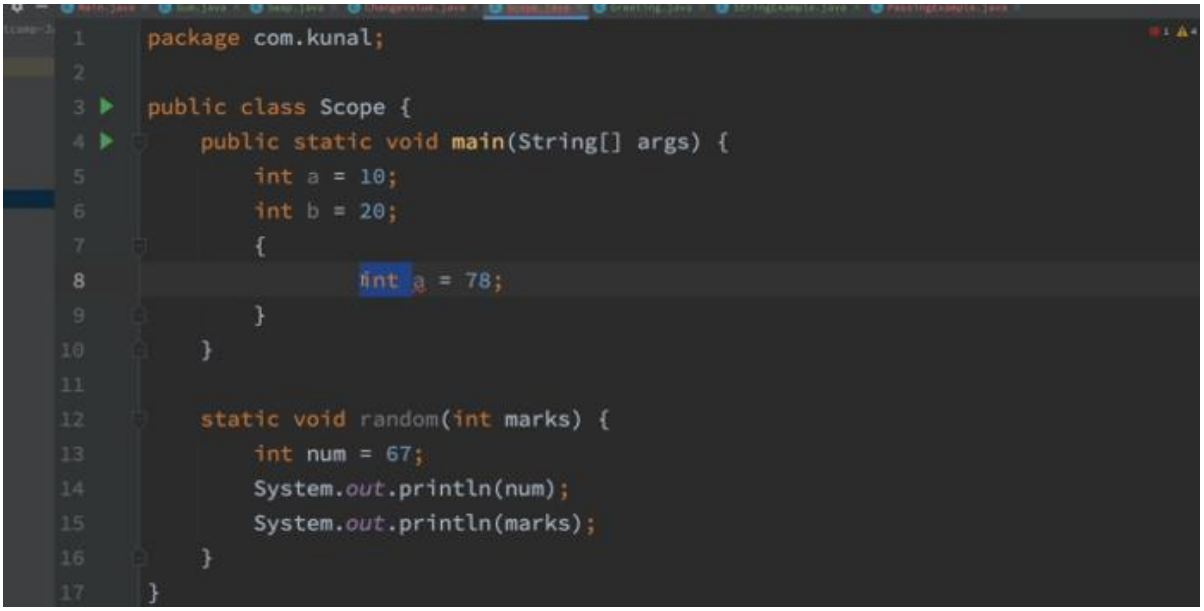
```
"C:\Program Files\Java\jdk1.8.0_221
[99, 1, 0, 2, 3, 4]

Process finished with exit code 0
```

SCOPE: Method Scope/Function Scope

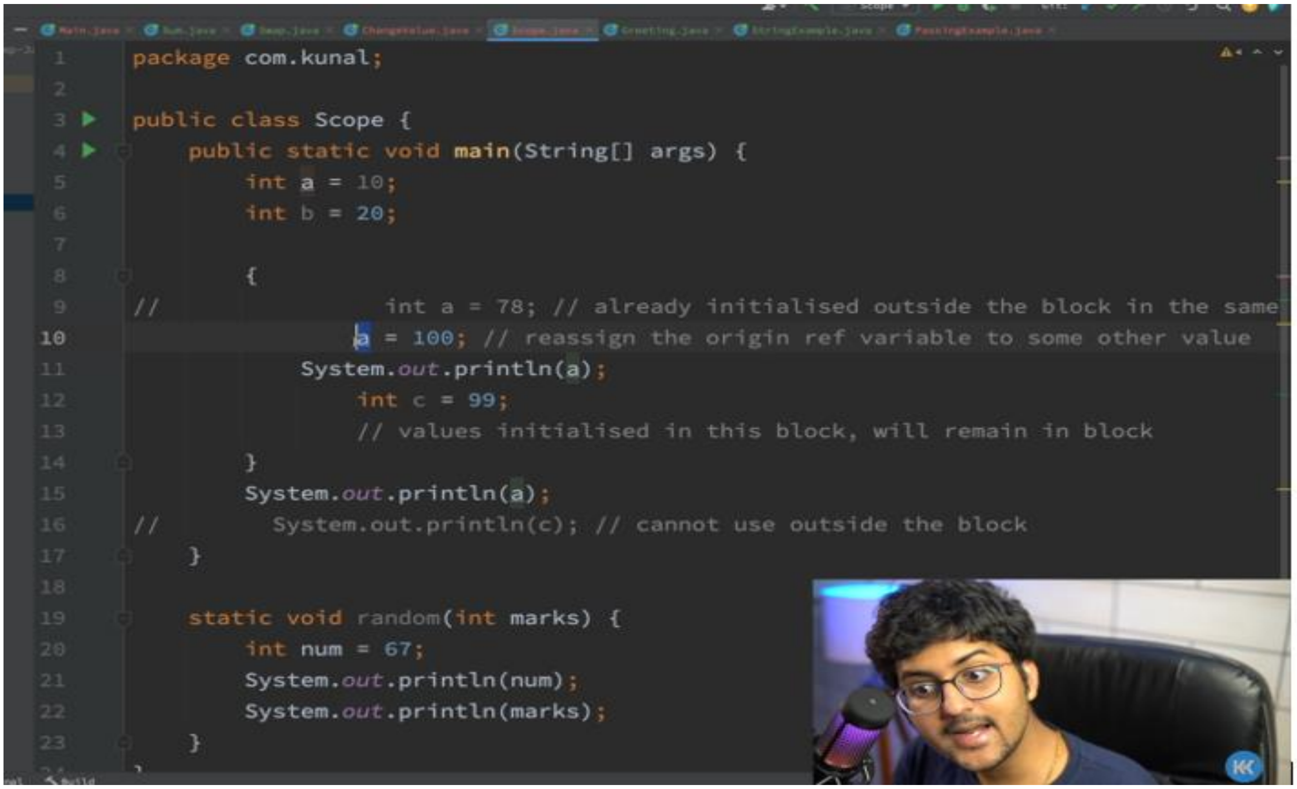


```java
package com.kunal;

public class Scope {
    public static void main(String[] args) {
        int a = 10;
        int b = 20;

        System.out.println(num);
    }

    static void random() {
        int num = 67;
        System.out.println(num);
    }
}
```

Block Scope



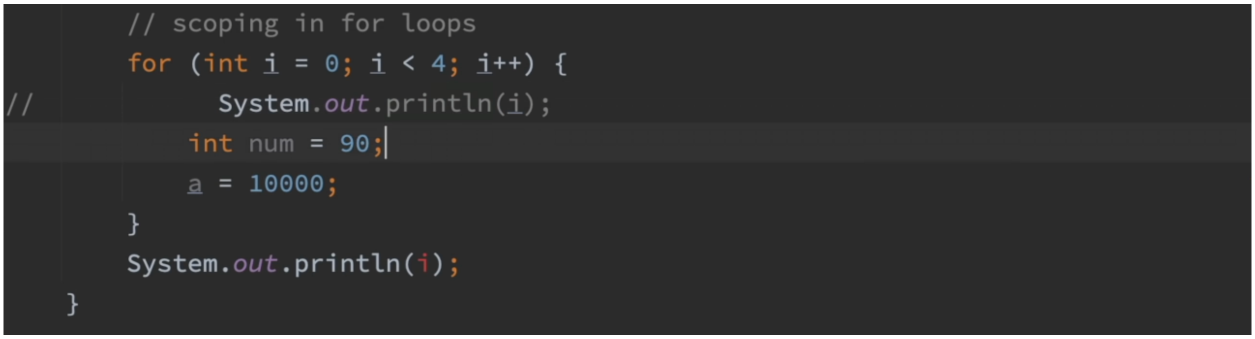
```java
package com.kunal;

public class Scope {
    public static void main(String[] args) {
        int a = 10;
        int b = 20;
        {
                int a = 78;
        }
    }

    static void random(int marks) {
        int num = 67;
        System.out.println(num);
        System.out.println(marks);
    }
}
```



```java
package com.kunal;

public class Scope {
    public static void main(String[] args) {
        int a = 10;
        int b = 20;

        {
//                int a = 78; // already initialised outside the block in the same
                a = 100; // reassign the origin ref variable to some other value
            System.out.println(a);
                int c = 99;
                // values initialised in this block, will remain in block
        }
        System.out.println(a);
//        System.out.println(c); // cannot use outside the block
    }

    static void random(int marks) {
        int num = 67;
        System.out.println(num);
        System.out.println(marks);
    }
}
```

Anything that is initialized outside the block cannot be initialized inside the block on value can be changed. But anything that is initialized inside the block can be initialed outside the block.

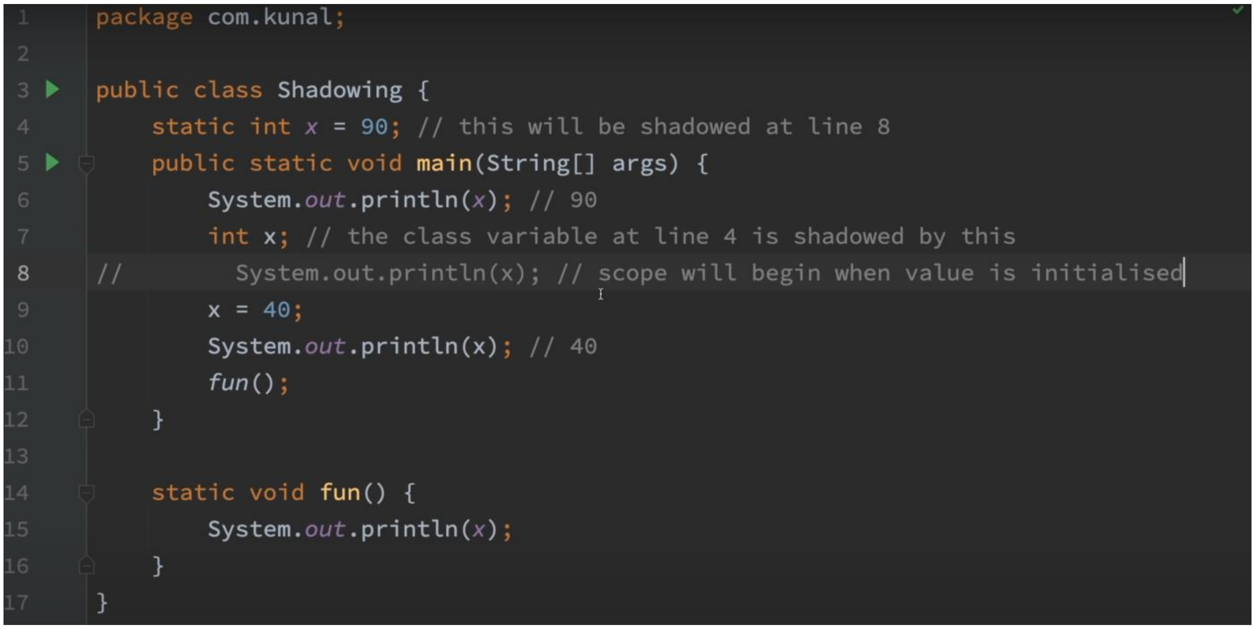
Scoping in loop



```
    // scoping in for loops
    for (int i = 0; i < 4; i++) {
//          System.out.println(i);
        int num = 90;
        a = 10000;
    }
    System.out.println(i);
}
```

It is same as block.

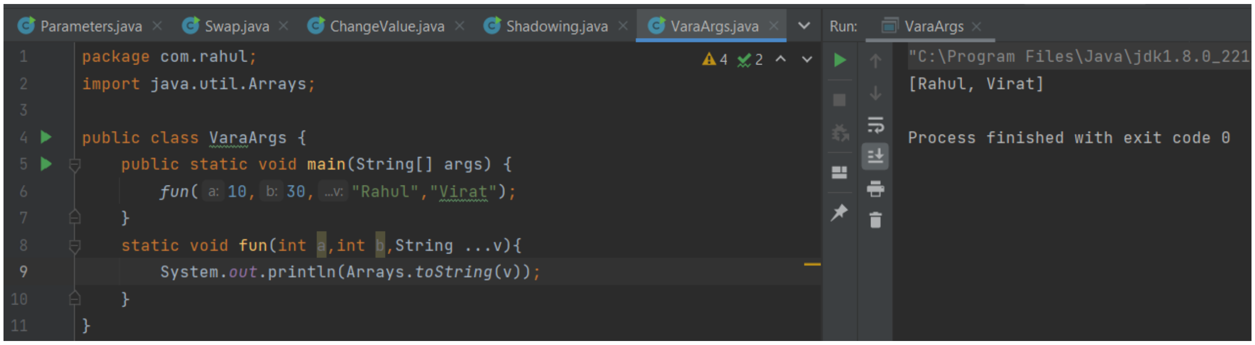
Shadowing



```
1   package com.kunal;
2
3   public class Shadowing {
4       static int x = 90; // this will be shadowed at line 8
5       public static void main(String[] args) {
6           System.out.println(x); // 90
7           int x; // the class variable at line 4 is shadowed by this
8   //          System.out.println(x); // scope will begin when value is initialised
9           x = 40;
10          System.out.println(x); // 40
11          fun();
12      }
13
14      static void fun() {
15          System.out.println(x);
16      }
17  }
```

Shadowing does not take place in methods.

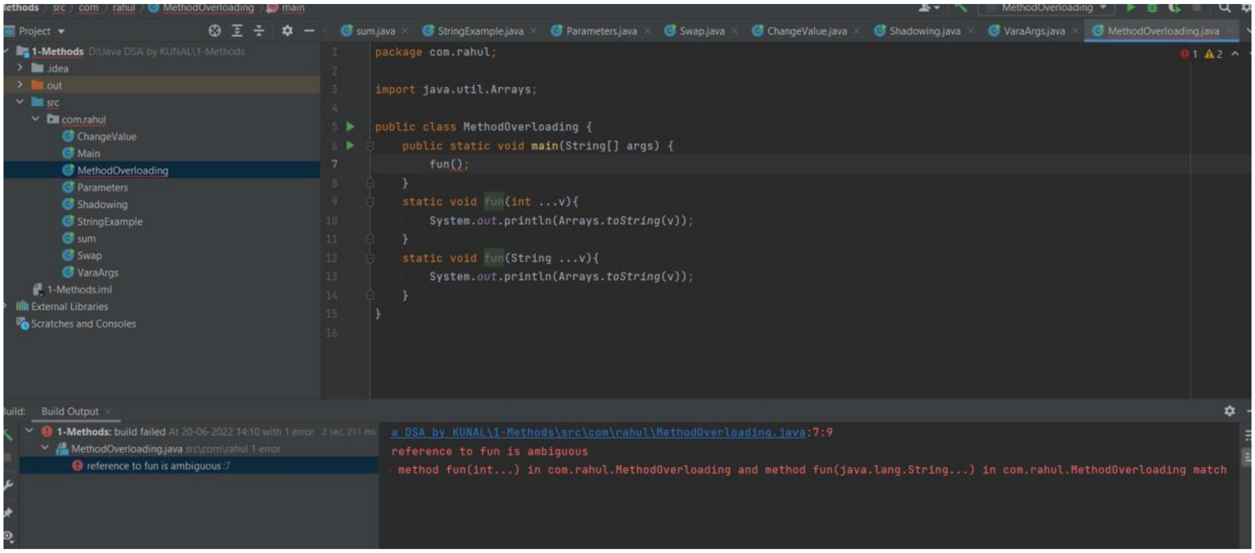
Variable length Argument



```
package com.rahul;
import java.util.Arrays;

public class VaraArgs {
    public static void main(String[] args) {
        fun( 10, 30, "Rahul","Virat");
    }
    static void fun(int a,int b,String ...v){
        System.out.println(Arrays.toString(v));
    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_221
[Rahul, Virat]

Process finished with exit code 0
```

Method Overloading

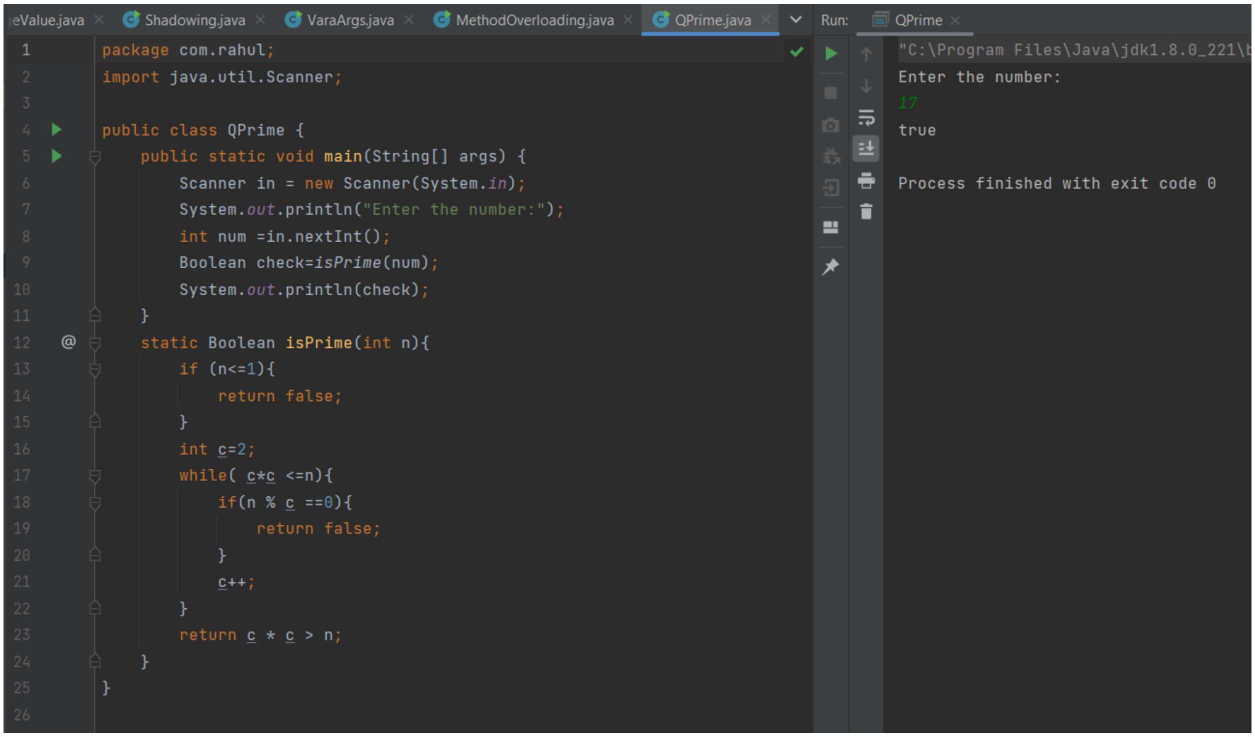


```java
package com.rahul;

import java.util.Arrays;

public class MethodOverloading {
    public static void main(String[] args) {
        fun();
    }
    static void fun(int ...v){
        System.out.println(Arrays.toString(v));
    }
    static void fun(String ...v){
        System.out.println(Arrays.toString(v));
    }
}
```

```
1-Methods: build failed At 20-06-2022 14:10 with 1 error  2 sec, 211 ms
  MethodOverloading.java src\com\rahul 1 error
    reference to fun is ambiguous :7

a DSA by KUNAL\1-Methods\src\com\rahul\MethodOverloading.java:7:9
reference to fun is ambiguous
 method fun(int...) in com.rahul.MethodOverloading and method fun(java.lang.String...) in com.rahul.MethodOverloading match
```

Questions

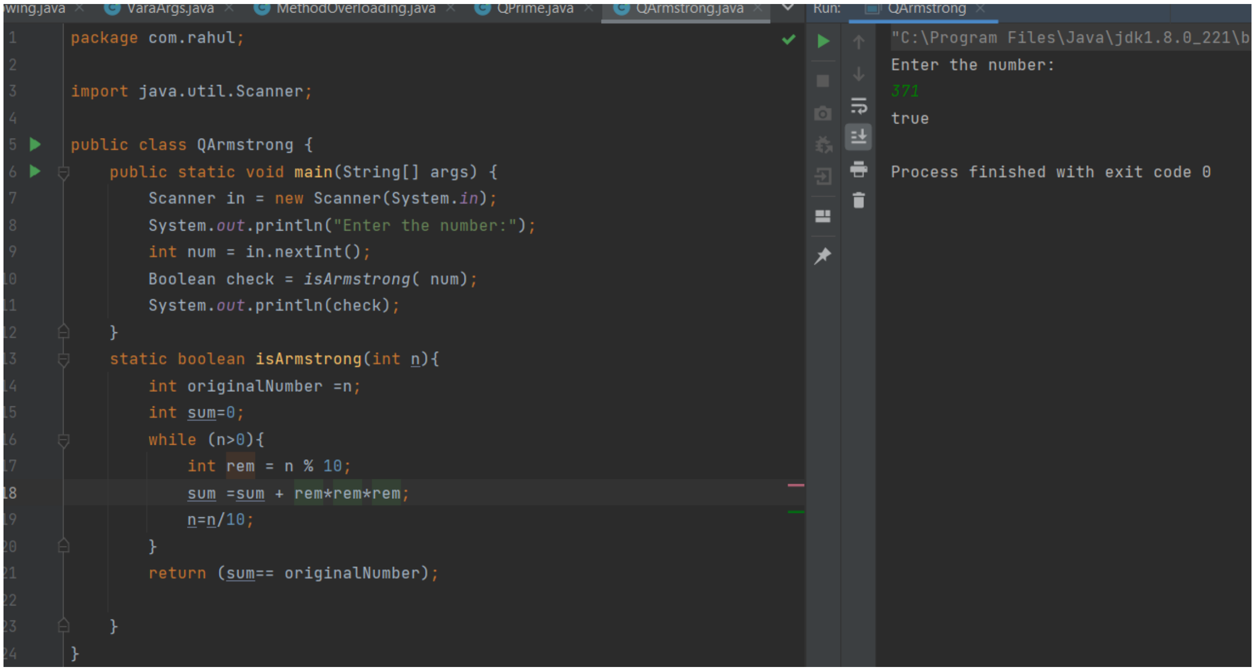
Prime number check



```java
package com.rahul;
import java.util.Scanner;

public class QPrime {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.println("Enter the number:");
        int num =in.nextInt();
        Boolean check=isPrime(num);
        System.out.println(check);
    }
    static Boolean isPrime(int n){
        if (n<=1){
            return false;
        }
        int c=2;
        while( c*c <=n){
            if(n % c ==0){
                return false;
            }
            c++;
        }
        return c * c > n;
    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_221\b
Enter the number:
17
true

Process finished with exit code 0
```

Armstrong number



```java
package com.rahul;

import java.util.Scanner;

public class QArmstrong {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.println("Enter the number:");
        int num = in.nextInt();
        Boolean check = isArmstrong( num);
        System.out.println(check);
    }
    static boolean isArmstrong(int n){
        int originalNumber =n;
        int sum=0;
        while (n>0){
            int rem = n % 10;
            sum =sum + rem*rem*rem;
            n=n/10;
        }
        return (sum== originalNumber);

    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_221\b
Enter the number:
371
true

Process finished with exit code 0
```