





# Day 4

 Created	@Mar 11, 2021
 Created by	 VIGNESH J
 Tags	Java

[Static](#)

[Method](#)

[Variable](#)

[Static Block](#)

[Pass by Value](#)

[Pass by Reference](#)

[Array](#)

## Static

## Method

- static method automatically loaded into the memory
- JVM - environment
- by default main method is static
- static method can call another static method without creating the object of the class
- non static methods, since they are not in memory, they cannot be called directly
- to call non static method you have to create the object for the class
- static method can't use super and this keyword
- static method of another class always called using class name

## Variable

- Instance variable
  - Auto initialized
- Class Variable
  - Auto initialized
- Local Variable
  - Not initialized by default
  - called as method variable
  - can't be static

## Static Block

- static block also created
- static block equal to constructor
- used for initialization
- called only once like constructor

```
package ga.vjee.day4;

public class StaticDemo {
    int i;

    public static void main(String[] args) {
        System.out.println("main method executed...");
        // main(new int[] {1,2,3});
        // StaticDemo obj=new StaticDemo();
        // obj.main(new char[] {'c','d'});
        A.met();
        A.met2();
    }

    public static void main(int a[]) {
        //static methods cannot use super and this keyword
        //super.toString();
        //this.i=10;
    }

    public void main(char c[]) {
        super.toString();
        this.i = 10;
        //non static method can call static methods without creating a object
        main(new int[]{1, 2, 3});
        //A.met();//static methods of another class should be always called using the class name.
    }
}

//static block is used for initializing , and remember it gets called only once...like constructor
class A {
    static {
        System.out.println("static block called...");
    }

    public A() {
        System.out.println("cons called for A");
    }

    static int i;

    public static void met() {
        System.out.println("met method called....");
    }

    public static void met2() {
        System.out.println("met222222222222 method called....");
    }
}

//static methods are always loaded by the jvm by default
//static methods can call another static method without creating the object of the class
//non static methods, since they are not in memory, they cannot be called directly
//To call non static methods, u have to create a object of the class
//static methods cannot have or use keywords "this and super"
```

---

## Pass by Value

- Copy of the value passed
- Original will not change

## Pass by Reference

- reference will be passed
- made a change in one reflect all ref

```
package ga.veee.day4;

public class PBVandPBR {
    public static void main(String[] args) {
        Laddu laddu = new Laddu();
        System.out.println("Size of original laddu..." + laddu.size);
        PassByValue pbv = new PassByValue();
        pbv.getLaddu(laddu.size);

        System.out.println("size of laddu after PBV..." + laddu.size);
        //in pass by value, a copy of the variable is passed, so the original will not change...

        PassByRef pbr = new PassByRef();
        pbr.getLaddu(laddu);
        //in case of pass by reference (laddu), then a copy is not made, but rather the original is supplied/passed
        System.out.println("size of laddu after PBR..." + laddu.size);
    }
}

class Laddu {
    int size = 10;
}

class PassByValue {
    public void getLaddu(int size) {
        size = 0;
    }
}

class PassByRef {
    public void getLaddu(Laddu laddu) {
        laddu.size = 0;
    }
}
```

## Array

- Array is collection same data type
- support 2D,3D,MD arrays
- index always start with 0
- by default arrays are initialized even though declare local variable
- once declared array size can not be changed

- arrays always pass by reference
- `.length` will be used to get the length of the array
- `System.arraycopy(arr, 0, copy, 0, arr.length);` will be used to copy the array

```
package ga.veee.day4;

public class ArrayDemo {
    public static void main(String[] args) {
        int a[] = new int[5];
        a[0] = 100;
        a[1] = 200;
        a[2] = 300;
        a[3] = 400;
        a[4] = 500;

        System.out.println(a[1]);

        for (int i = 0; i < a.length; i++) {
            System.out.println(a[i]);
        }

        int arr[] = {1, 2, 3, 4, 5};
        System.out.println(arr[0]);

        int arr2[] = new int[5];
        //by default arrays are initialized, even though they are declared local
        for (int i : arr2) {
            System.out.println(i);
        }
        //once declared the array size cannot be changed...

        int twodarr[][] = {
            {1, 2, 3, 4},
            {2, 3, 4, 5},
            {4, 5, 6, 7}
        };
        int twodarr2[][] = new int[3][4]; //three rows four columns
        for (int i = 0; i < twodarr2.length; i++) { //this will give you length of rows...
            for (int j = 0; j < twodarr2[i].length; j++) { //this will give length of columns in that row
                System.out.println(twodarr2[i][j]);
            }
        }

        for (int i[] : twodarr) {
            for (int j : i) {
                System.out.print(j + "\t");
            }
            System.out.println();
        }
    }
}
```

```
package ga.veee.day4;

public class ArraysPBR {
    public static void main(String[] args) {
        int arr[] = {1, 2, 3, 4, 5};
        for (int i : arr) {
            System.out.print(i + "\t");
        }
    }
}
```

```

//you should be very carefull when passing a array because the original will get changed...
int copy[] = new int[5];
System.arraycopy(arr, 0, copy, 0, arr.length); //to create a copy of the original array
getArray(copy);

System.out.println();
for (int i : arr) {
    System.out.print(i + "\t");
}

System.out.println("\nprint the copy...\n");

for (int i : copy) {
    System.out.print(i + "\t");
}


}

public static void getArray(int a[]) { // arrays are always pass by reference....
    a[0] = 100;
    a[4] = 500;
}
}

```

#### Best Free Online Video Recording & Screen Capture Software | Fluvid

Fluvid is the best all-in-one online screen capture & video recording software that is available for free. Fluvid helps you record, edit, communicate & share your video messages in the most simplest way.

 <https://fluvid.com/videos/detail/aQadEUZV24HnqX61B#.YEmrygJ5F4.link>

10 AM - 11 AM