Difference between static and non-static variables in Java

Last Updated: 26-04-2019
There are three types of variables in Java:

- Local Variables
- Instance Variables
- Static Variables

The Local variables and Instance variables are together called Non-Static variables. Hence it can also be said that the Java variables can be divided into 2 categories:

- **Static Variables:** When a variable is declared as static, then a single copy of the variable is created and shared among all objects at a class level. Static variables are, essentially, global variables. All instances of the class share the same static variable.
  Important points for static variables :-
  - We can create static variables at class-level only. See here
  - static block and static variables are executed in order they are present in a program.

  Below is the Java program to demonstrate that static block and static variables are executed in order they are present in a program.

filter_none
edit
play_arrow
brightness_4

```
// Java program to demonstrate execution
// of static blocks and variables
class Test {
    // static variable
    static int a = m1();

    // static block
    static
    {
        System.out.println("Inside static block");
    }

    // static method
    static int m1()
    {
        System.out.println("from m1");
        return 20;
    }

    // static method(main !!)
    public static void main(String[] args)
    {
        System.out.println("Value of a : " + a);
        System.out.println("from main");
    }
}
```

Output:

```
from m1
Inside static block
Value of a : 20
from main
```

- **Non-Static Variable**
  - **Local Variables**: A variable defined within a block or method or constructor is called local variable.
    - These variable are created when the block in entered or the function is called and destroyed after exiting from the block or when the call returns from the function.
    - The scope of these variables exists only within the block in which the variable is declared. i.e. we can access these variable only within that block.
    - Initialisation of Local Variable is Mandatory.
  - **Instance Variables:** Instance variables are non-static variables and are declared in a class outside any method, constructor or block.
    - As instance variables are declared in a class, these variables are created when an object of the class is created and destroyed when the object is destroyed.
    - Unlike local variables, we may use access specifiers for instance variables. If we do not specify any access specifier then the default access specifier will be used.
    - Initialisation of Instance Variable is not Mandatory. Its default value is 0
    - Instance Variable can be accessed only by creating objects.

Example :

filter_none
edit
play_arrow
brightness_4

```
// Java program to demonstrate
// non-static variables

class GfG {

    // non-static variable
    int rk = 10;

    public static void main(String[] args)
    {
        // Instance created inorder to access
        // a non static variable.
        Gfg f = new Gfg();

        System.out.println("Non static variable"
```

```
                                + " accessed using instance"
                                + " of a class");
            System.out.println("Non Static variable "
                                + f.rk);
        }
    }
```

Output:

```
Non static variable accessed using instance of a class.

Non Static variable 10
```

The main differences between static and non static variables are:

STATIC VARIABLE	NON STATIC VARIABLE
Static variables can be accessed using class name	Non static variables can be accessed using instance of a class
Static variables can be accessed by static and non static methods	Non static variables cannot be accessed inside a static method.
Static variables reduce the amount of memory used by a program.	Non static variables do not reduce the amount of memory used by a program
Static variables are shared among all instances of a class.	Non static variables are specific to that instance of a class.
Static variable is like a global variable and is available to all methods.	Non static variable is like a local variable and they can be accessed through only instance of a class.

Recommended Posts:

- Difference between Static variables and Register variables in C
- Static and non static blank final variables in Java
- Understanding storage of static methods and static variables in Java
- Difference between static and non-static method in Java

- Difference between Static-1 and Static-0 hazard
- Understanding "static" in "public static void main" in Java
- Why non-static variable cannot be referenced from a static method in Java
- Are static local variables allowed in Java?
- Assigning values to static final variables in Java
- Static Variables in Java with Examples
- Internal static variable vs. External static variable with Examples in C
- Difference between Static and Dynamic SQL
- Difference between Static and Shared libraries
- Difference between Static and Dynamic Routing
- Difference between Static and Dynamic Web Pages
- Difference between Static and Dynamic IP address
- Difference between Static allocation and Stack allocation
- Difference between Static Allocation and Heap Allocation
- Difference between Static and Dynamic Hazard
- Difference between self::$bar and static::$bar in PHP

bestharadhakrishna
Check out this Author's contributed articles.

Please Improve this article if you find anything incorrect by clicking on the "Improve Article" button below.

Improved By : shubham_singh

Article Tags :
Difference Between

Java

Static Keyword

Practice Tags :
Java

thumb_up

6

To-do Done

3.5

Based on **2** vote(s)

Improve Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Post navigation
Previous
first_page How to Clone a List in Java?