

**Variable in Java – How to Define and Types Of Variables:** This is another extended post of core java tutorial post, but after discussion of the first java hello world program, now we are moving to the essential learning of Java programming language. Now Let's start with learning what a variable, different type of variables, and their behaviors is.

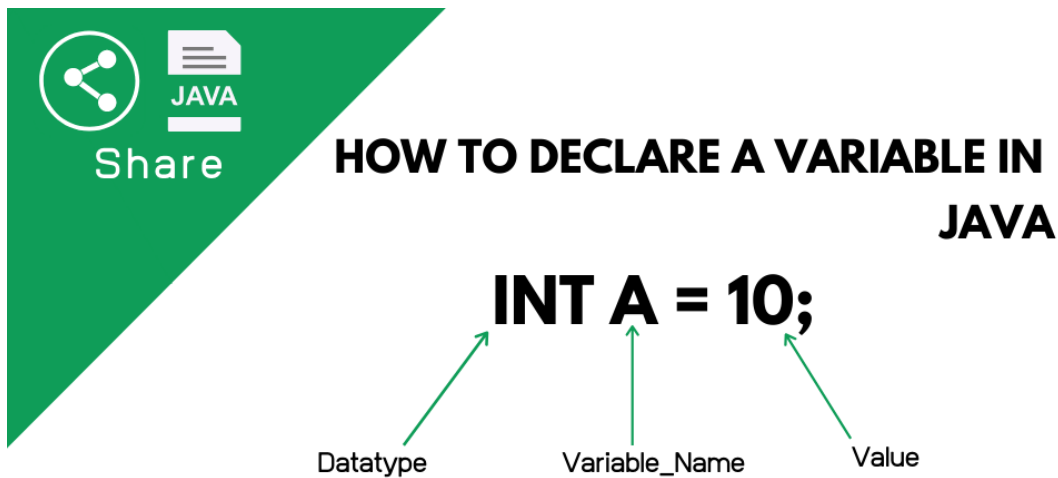
## Variable In Java & Types

A Variable in java is nothing but a name given to the memory location and is the central area where we store the program. the values stored in the variable that can change during the execution of the program. Those variables used in the program they all are must declare before using in the program, and whatever operation you can perform on the variable that will affect that memory location where the variable is stored.

## How to Declare a variable in Java

we can declare a variable in java by following the below syntax:

```
Data_type Name_of_Variable = Value;
```



www.softwaretestingo.com

### How to Declare a variable in Java

**Note:** The Data\_type and Name\_of\_Variable are compulsory during declare a variable, but the Value is optional because, in java, we can first declare the variable, and later on, as per the requirement, we can assign the value to that variable.

```
Int var_num;  
Int var_num=50;
```

In the first statement, if you see, we are declaring a variable without a value, and in the second statement, we are declaring a variable with the value. So in both statements, Int is a data type, and var\_num is the variable name. Don't worry, we are going to discuss more briefly about the different **data types in Java** in the future posts for the time being understand that Int data type allows the var\_num variable to hold only **integer** values in it.

# Different ways to Declare and Assign Values to a Variable

We Can assign the value to a variable during at the time of declaration like below

```
Int num=10;  
Char ch='S';  
String str="SoftwareTestingo";
```

Similarly, we can define the variable at the beginning of the program, but we can assign the value later, like below.

Let's Check the below program to understand how we can declare variables and also assign values to those variables later.

```
package softwaretestingo.java.basics;

public class VariableProgram1
{
    public static void main(String[] args)
    {
        /*This example we are see how to declare a variable first and later
        * we can assign value to those variable */
        //Declaring Variables
        int num;
        char ch;
        String str;

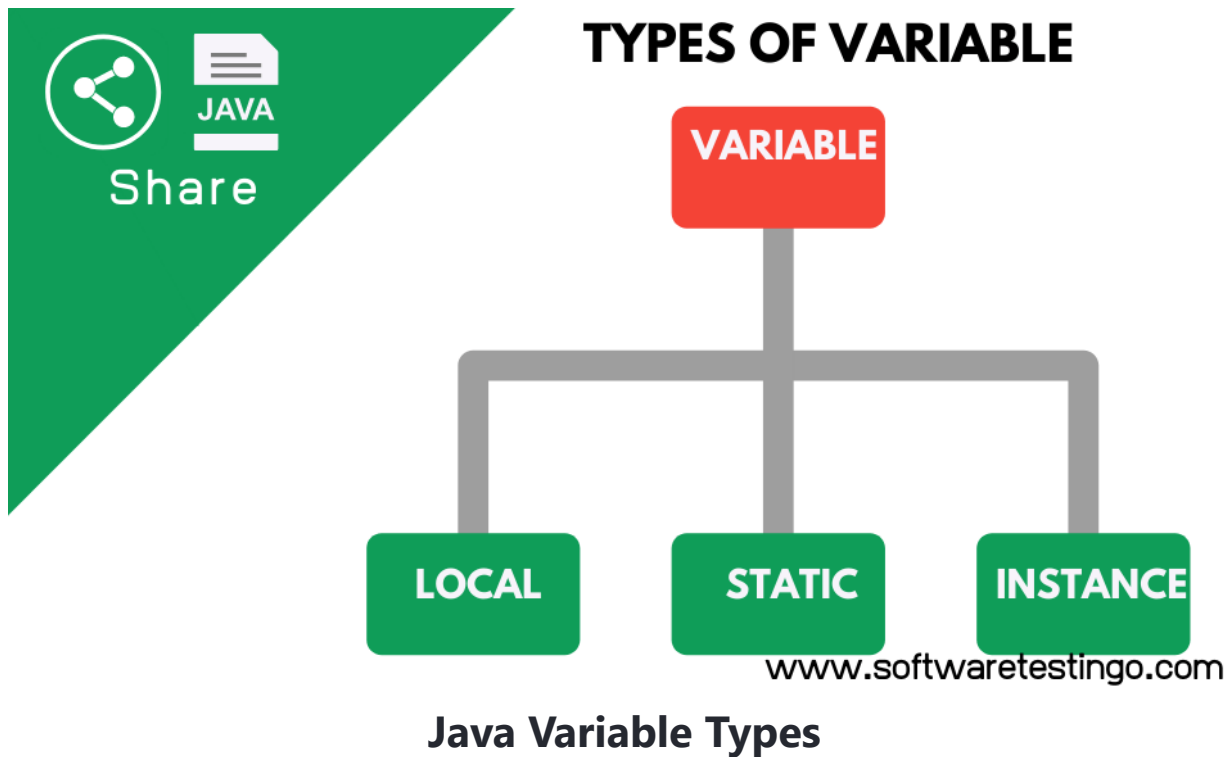
        //Assign the Value to the above variables
        num=2019;
        ch='A';
        str="SoftwareTestingo";

        //Let's Print The Values
        System.out.println("The Value Of num:-"+num);
        System.out.println("The Value Of ch:-"+ch);
        System.out.println("The Value Of str:-"+str);
    }
}
```

## Java Variable Types

In Java mainly we have three different types of variables are there, that are

- Local variable Java
- Static or Class Variable Java
- Instance Variable Java



## Local Variables in Java

Those variables we are declaring inside the method body those variables called as Local variable. The scope of the local variables limited that is within the method area only because the local variable created when the method, constructor, or block starts execution, and the variable destroyed when you exit from the method or block. So you can't change their values and access them outside of that method.

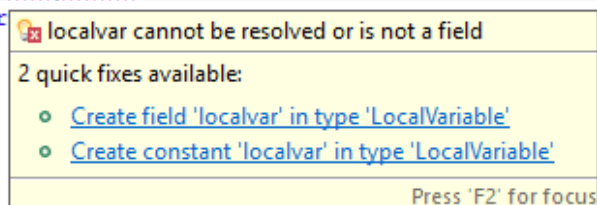
```
package softwaretestingo.java.basics;

public class LocalVariable
{
    String myVar="Outside Local Test Method";
    //Local Method
    public void localTestMethod()
    {
        String localvar="Inside Local Test Method";
        System.out.println("Local Variable str Value:-"+localvar);
    }

    public static void main(String[] args)
    {
        LocalVariable obj=new LocalVariable();
        obj.localTestMethod();
        //System.out.println(obj.localvar);
        System.out.println("Variable Value Outside Method:-"+obj.myVar);
    }
}
```

If you see the above Java program, we can see that the local Variable (localvar) can be accessible inside the local method. Still, when we try to access that variable outside of the localTestMethod, we are getting below error. This demonstrates the scope of a variable.

```
public static void main(String[] args)
{
    LocalVariable obj=new LocalVariable();
    obj.localTestMethod();
    System.out.println(obj.localvar);
    System.out.println("Var" ;
}
```



## Local Variable Error

# Static or Class Variables In Java

A static variable is also called a class variable because these variables are associated with the class and can be accessed by all the instances of that class. When we create a class or static variable, at the time of declaration, we declare with the static keyword. One copy of class or static variable created for each class and the same copy is used by all the objects.

## Difference: Instance Variable VS Static Variable

**Note:** Static Variable or Methods are executed when your creating the objects.

```
package softwaretestingo.java.variable;
public class ClassVariable
{
    public static int intVar=10;
    public static void main(String[] args)
    {
        ClassVariable obj1=new ClassVariable();
        ClassVariable obj2=new ClassVariable();
        System.out.println("Object1: "+obj1.intVar);
        System.out.println("Object2: "+obj2.intVar);

        //Initialize Class Variable Inside Main Method
        obj1.intVar=20;
        System.out.println("Updated Value: "+obj1.intVar);
        System.out.println("After Update the String Value: "+intVar);
        System.out.println("By Calling From a Newly Created Object: "+obj2.intVar);
    }
}
```

## Output:

```
Object1: 10  
Object2: 10  
Updated Value: 20  
After Update the String Value: 20  
By Calling From a Newly Created Object: 20
```

## Explanation:

From the above output, you can find out that when you are using a class variable, then the value is the same for all the different instances which you can create during the program. If you change the value by a single instance, also it will reflect all other instances.

```
System.out.println("Class Variable: "+intVar);
```

You can also access a class variable like above, but you can not access instance and local variables like this.

## Instance Variables In Java

When you are creating an instance (Object), then that instance will create an own copy of instance variables. If you change something on a specific instance variable, then that does not affect other instance variables. This shows that each instance has its copy of instance variables.



```
package softwaretestingo.java.variable;

public class InstanceVariable
{
    public String instVariable="Instance Variable";
    public static void main(String[] args)
    {
        InstanceVariable obj1=new InstanceVariable();
        InstanceVariable obj2=new InstanceVariable();

        System.out.println("Obj1 Value: "+obj1.instVariable);
        System.out.println("Obj2 Value: "+obj2.instVariable);

        //Instance Variable Value Changed
        obj1.instVariable="Variable Value Changed";
        System.out.println("Obj1 Value: "+obj1.instVariable);
        System.out.println("Obj2 Value: "+obj2.instVariable);
    }
}
```

## Output:

```
Obj1 Value: Instance Variable
Obj2 Value: Instance Variable
Obj1 Value: Variable Value Changed
Obj2 Value: Instance Variable
```

## Explanation:

If you see the above output, then we find that the instance variable value is not the same for all the different instances like a class variable.

## Final Words:

We hope this article gives a clear idea about the variables in Java, different types of java variable types, instance variable, java class variable, local variable java, & Also discuss How to define variables in java. Let me know if you have any doubts and confusion about the variables, and then I am happy to help you.

## Write A Program Of Refernce Java Example Program?

```
package com.java.Softwaretestingblog;

class a112
{
    int i=10;
}

class b112 extends a112
{
    int i=20;
    //int j=40;
}

public class ReferEnceExample {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        a112 obj=new b112();
        System.out.println(obj.i);
        //System.out.println(obj.j); Newly added variables in child class are not
        //accessible
    }

}
```

## Output:

10

**What is variables in Java?**

A Java variable is a piece of memory that can contain a data value. A variable thus has a data type.

**What are the different variable types?**

In Java we have 3 different types of variables that are local, static and instance variables.

**What are local variables in Java?**

A local variable in Java is a variable that's declared within the body of a method. Then you can use the variable only within that method.

**How do you declare a variable in Java?**

To declare a variable in java, you will specify the data type after that leave at least one space, then the name for the variable and end the line with a semicolon ( ; ).

**What is local variable in Java with example?**

A local variable is a variable which is declared inside a method body, block or constructor. It means that variable is only accessible inside the method, block or constructor where that is declared.

**What are static variables in Java?**

A static variable is common to all the objects of the class because it is a class level variable. In other words you can say that only a single copy of static variable is created and shared among all the instances of the class.

**What is scope of variable in Java?**

The scope of a variable specifies the region of the source program where that variable is known, accessible and can be used.

**What is instance variable in Java?**

Instance variable is also known as object level variable as for every instance the value of instance variable is newly initialized.