

# Instance Variable in Java

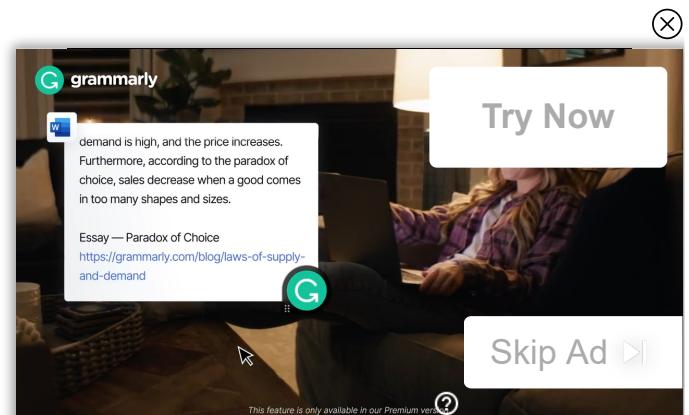
In any programming language, the program needs identifiers for storing different values that can be used throughout the program. These identifiers are variables.

## Variable in Java

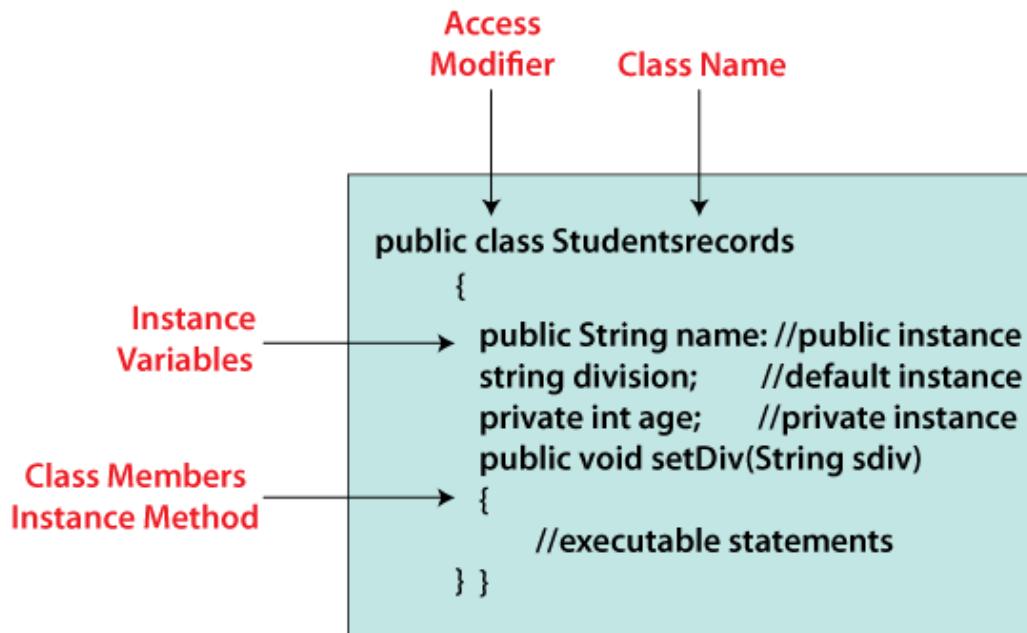
- A variable is a name assigned to a value that is stored inside the system memory. The value can be updated during the program execution.
- In Java programming, the variables used for the program need to declare them first.
- The variable is declared using a data type followed by the identifier name. The variable can be initialized at the time of declaration or it can be assigned a value taken from the user during the program execution.
- There are basically three types of variables in Java,
  1. Java Local variable
  2. Java Instance variable
  3. Java Static variable / Java class variable

## Java Instance Variable

- The variables that are declared inside the class but outside the scope of any method are called instance variables in Java.
- The instance variable is initialized at the time of the class loading or when an object of the class is created.
- An instance variable can be declared using different access modifiers available in Java like default, private, public, and protected.
- Instance variables of different types have default values that are specified in the next point.



↑ SCROLL TO TOP



## Features

1. To use an instance variable an object of the class must be created.
2. An instance variable is destroyed when the object it is associated with is destroyed.
3. An instance variable does not compulsory need to be initialized.
4. Instance variables are accessible inside the same class that declares them.

## Limitations of Instance Variable

1. It cannot be declared static, abstract, striftp, synchronized, and native.
2. It can be declared final and transient.
3. It can be of any of the four access modifiers available in Java (private, public, protected, and default).

## Default Values of Instance Variables in Java

The instance variables in Java are of different data types as follows;

Instance variable type
boolean
byte
short

↑ SCROLL TO TOP
Try Now

Skip Ad ▶

A video player window showing a Grammarly advertisement. The ad features a woman sitting on a couch, looking at a laptop. The text in the ad reads: "demand is high, and the price increases. Furthermore, according to the paradox of choice, sales decrease when a good comes in too many shapes and sizes." Below this, it says "Essay — Paradox of Choice" and provides a link: "https://grammarly.com/blog/laws-of-supply-and-demand". There are "Try Now" and "Skip Ad" buttons in the corners of the video player.

int	0
double	0.0d
float	0.0
long	0L
Object	null
char	\u0000

Let's use instance variable in Java program.

## Using Instance Variable in a Java Program

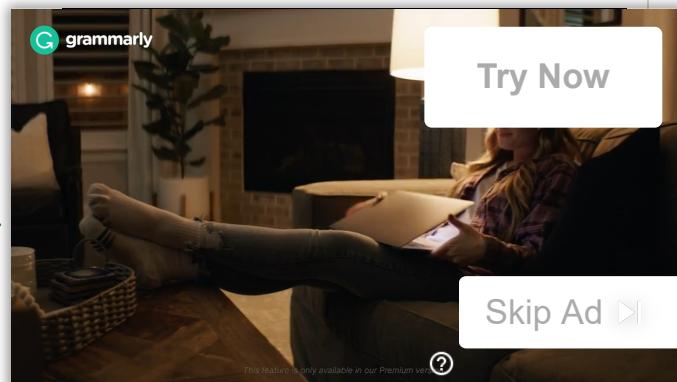
In the following Java program, a class **Studentsrecords** is declared and instance variables are created using different access modifiers.



### Studentsrecords.java

```
public class Studentsrecords
{
    /* declaration of instance variables.*/
    public String name; //public instance
    String division; //default instance
    private int age; //private instance
    /* Constructor that initialize an instance variable.
    public Studentsrecords(String sname)
```

↑ SCROLL TO TOP



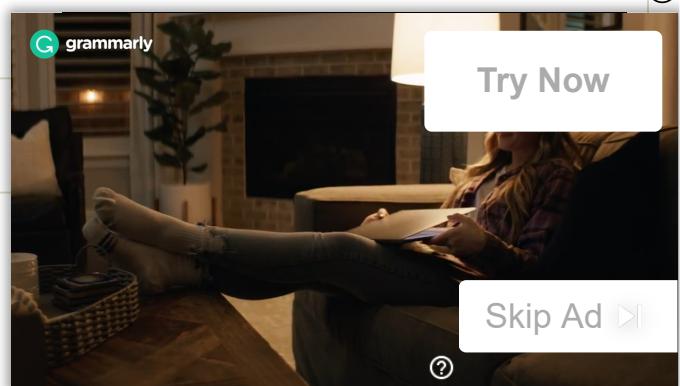
```
name = sname;  
}  
  
/* Method to intialize an instance variable. */  
public void setDiv(String sdiv)  
{  
    division = sdiv;  
}  
  
/* Method to intialize an instance variable. */  
public void setAge(int sage)  
{  
    age = sage;  
}  
  
/* Method to display the values of instance variables. */  
public void printstud()  
{  
    System.out.println("Student Name: " + name );  
    System.out.println("Student Division: " + division);  
    System.out.println("Student Age: " + age);  
}  
  
/* Driver Code */  
public static void main(String args[])  
{  
    Studentsrecords s = new Studentsrecords("Monica");  
    s.setAge(14);  
    s.setDiv("B");  
    s.printstud();  
}
```

### Output:

```
Student Name: Monica
```

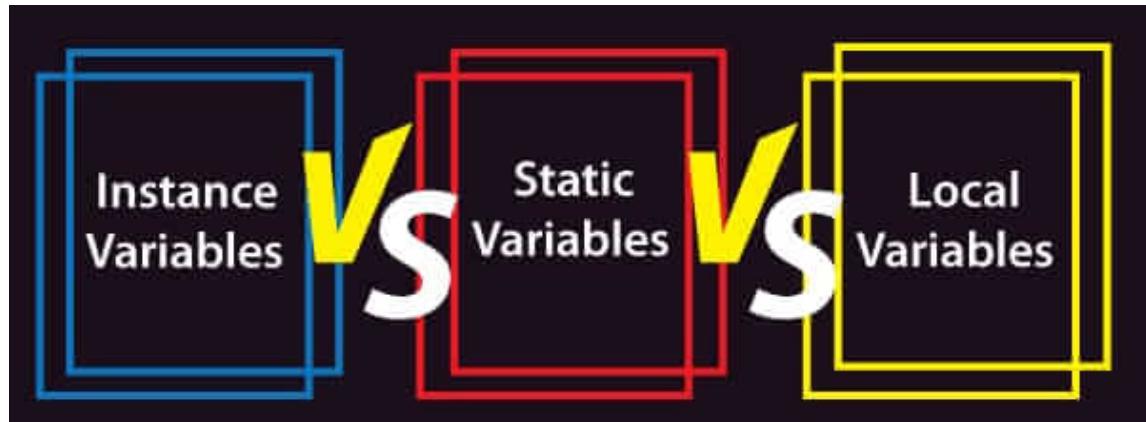
```
: B
```

```
↑ SCROLL TO TOP
```



The above Java program initializes the instance variables by declaring an object of the class ***Studentsrecords***. The values of instance variables ***name***, ***division***, and ***age*** are displayed using ***printstud()*** method.

## Difference between Local, Instance and Static variables in Java



Sr. No.	Local variables	Instance variables	Static variables
1.	Variables declared within a method are local variables.	An instance variable is declared inside a class but outside of any method or block.	Static variables are declared inside a class but outside of a method starting with a keyword static.
2.	The scope of the local variable is limited to the method it is declared inside.	An instance variable is accessible throughout the class.	The static variable is accessible throughout the class.
3.	A local variable starts its lifetime when the method is invoked.	The object associated with the instance variable decides its lifetime.	The static variable has the same lifetime as the program.
4.	Local variable is accessible to all the objects of the class.	Instance variable has different copies for different objects.	Static variables only have one single copy of the entire class.
5.	Used to store values that are required for a particular method.	Used to store values that are needed to be accessed by different methods of the class.	

↑ SCROLL TO TOP