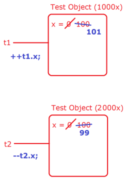
Role of non static field in Object creation :
----

* As we know, when we create the object then only the memory will be allocated for all the non static fields.

* Whenever we create an **object in Java**, A **separate copy** of all the **non static fields are created** with each and every object.

```
package com.nit.role_of_fields;

class Test
{
    int x = 100;
}
public class RoleOfNonStaticFields
{
    public static void main(String[] args)
    {
        Test t1 = new Test();
        Test t2 = new Test();

        ++t1.x;
        --t2.x;

         IO.println(t1.x); //101
         IO.println(t2.x); //99
    }

}
```



What is Class Variable OR Static field in Java ?
----

* A class Variable OR static field is a variable which we should declare **at class level**.

* If we declare a Variable inside the class with **static modifier then It is called Static Field.**

* Whenever We load a class in Java then automatically the memory will be allocated for static field and initialized with default value.

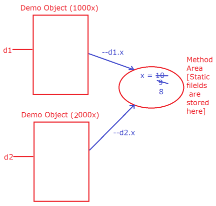
* As far as its **accessibility is concerned**, It is accessible from anywhere in the class and It will be automatically destroy when the class will be un-loaded.

* A static field, we can directly access with the class name here object is not required.

Role of static field in Object creation :
----

* Whenever we create an object in Java then a **single copy of static filed will be created and It will be sharable by all the objects.**

```
package com.nit.role_of_fields;

class Demo
{
    static int x = 10;
}

public class RoleOfStaticField
{
    public static void main(String[] args)
    {
        Demo d1 = new Demo();
        Demo d2 = new Demo();

        --d1.x;
        --d2.x;

        IO.println(d1.x);
        IO.println(d2.x);

    }

}
```



Note : static fields are mainly used to save the memory.

Non static Fields = Multiple Copies with each and every object

Static Field = Single copy and Sharable by all the Objects

* When we should declare a field as a non static field and when we should declare as a static field ?

Non static field :
----
* If the value of the variable **is <u>different</u> with respect to objects then we should declare non static field.**

static field :
----
* If the value of the variable **is <u>common</u> to all the objects then we should declare static field.**

Example 1 :

```
public class Student
{
   int rollNumber;
   String name;
   String address;
   static String collegeName = "NIT";
   static String courseName = " Java";
}
```

Example 2 :

```
public class Customer
{
  int custId;
  String name;
  String address;
  long mobileNumber;
  static String IFSCCode = "SBI001";
  static String branchName = "Ameerpet";
}
```

WAP to show the use of static and non static fields :

```
package com.nit.blc;

//BLC
public class Student
{
   int rollNumber;
   String studentName;
   String studentAddress;
   static String collegeName = "NIT";
   static String courseName = "Adv Java";

   public void setStudentData(int roll, String name, String address)
   {
       rollNumber = roll;
       studentName = name;
       studentAddress = address;
   }

   public void showStudentData()
   {
       IO.println("Roll Number is :"+rollNumber);
       IO.println("Name is :"+studentName);
       IO.println("Address is :"+studentAddress);
       IO.println("College Name is  :"+collegeName);
       IO.println("Course Name is :"+courseName);
   }
}


package com.nit.elc;

import com.nit.blc.Student;

//ELC
public class StudentDemo
{
   void main()
   {
       Student raj = new Student();
       raj.setStudentData(101, "Raj", "Ameerpet");
       raj.showStudentData();

       IO.println("..............");
       Student priya = new Student();
       priya.setStudentData(102, "Priya", "S R Nagar");
       priya.showStudentData();
   }
}
```

Assignement on Bank Customer : [Example 2]

Upto Here How many ways we have already learned to initialize the non static field :
----

1) By using Object referenec
   Example : raj.rollNumber = 101;

2) By using Method Parameter
   Example :
```
public void setStudentData(int roll, String name)
{
   rollNumber = roll;
   studentName = name;
}
```

3) At the time of declaration, We can also initialize
   int x = 100;

==