Types of Variables based on the declaration position :

* Based on declaration position, variables are further divided into following types .

a) Class Variables OR Static Field
b) Instance Variables OR Non Static Field — **Class Level Variables**

c) Local Variable
d) Parameter Variable — **Method Level Variables**

Example :

```
public class Test
{
    static int x = 100;  //Static Field
           int y = 200;  //Non Static Field

    public void accept(int z)
    {
        int d = 400;
    }
}
```

d → Local Variable; z → Parameter Variable

What is Object Oriented Programming (OOPs):

What is an Object in Java ?

* An Object is a **physical entity** which exist in the real world.
Example : Car, Bike, Laptop, Person e.t.c

Characteristics of an Object :

* An Object has 3 characteristics :

1) Name of the Object (Identification of the Object)
2) State of the Object (Attributes OR Data OR Attributes OR **Non static field**)
3) Behavior of the Object (Functionality of the Object **[Non static method]**)

How we can create an object in java

* In order to create an object in java **new keyword in java. We should use compulsory use the class name. [Without class name we cannot create an Object using new keyword]**

Example :

```
public class Employee
{
}

Employee e1 = new Employee();
```

e1 → **Object name**; e1 → **reference variable**; new → **keyword to allocate memory allocation**; Employee() → **Constructor**

* An object oriented programming is a technique OR methodology to design and develop the programs using **class and Object.**

* In order to develop the programs on **real life objects, Object Oriented Programming is introduced. If a person is able to write the program on those real life object then we can say the person is an expert in object oriented programming.**

* In OOP we always concentrate on the **objects rather than function.**

Advantages of OOP :

1) Modularity (Dividing the bigger task into smaller task)
2) Reusability (Reuse the existing code properties and methods) so many times)
3) Flexibility (Easy to maintain [By using interface])

Features of OOP :

1) Class
2) Object
3) Abstraction
4) Encapsulation
5) Inheritance
6) Polymorphism

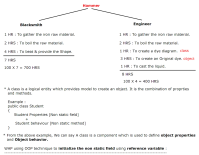
What is a class ?

* A Class is a **model OR blueprint OR template OR prototype** for creating the Object.
Example :

We need to talk to CEVE, engineer for a rough diagram before we build the Restaurant :

Diagram ---> class

Physical Restaurant ---> Objects



* A class is a logical entity which provides model to create an object. It is the combination of properties and methods.

Example :

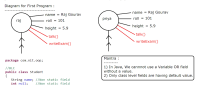
```
public class Student
{
    Student Properties [Non static field]

    Student behaviour [Non static method]
}
```

* From the above example, We can say A class is a component which is used to define **object properties** and **Object behavior.**

WKT using OOP technique to **initialize the non static field** using **reference variable** :

(As of now, Our BLC and ELC class should be in the same package)

Diagram for First Program :




```
package com.ist.raw;

import java.io.*;
public class Student
{
    String name;  //Non static field
    int roll;  //Non static field
    double height;  //Non static field

    public void sleep()
    {
        Stream.out.println("Hello Batch 14 & 15, I am I'name");
        Stream.out.println("My roll number is I'roll" and my height is I'height);
    }

    public void writeCode()
    {
        Stream.out.println("Every Saturday "name" is writing the Code");
    }

}
```

Note :
1) In Java, We cannot use a Variable OR field without it value.
2) Only Class level fields are having default value.

```
package com.ist.exec;

//BLC
public class TestStudent
{
    public static void main(String[] args)
    {
        Student r1 = new Student();

        //Initializing the non static fields (name) using reference variable
        r1.name = "Raj Gaurav";
        r1.roll = 201;
        r1.height = 5.9;

        //Calling the behavior
        r1.sleep();
        r1.writeCode();

        Stream.out.println("-----");

        Student r2 = new Student();
        //Initializing the non static fields (name) using reference variable
        r2.name = "Ankit Mittal";
        r2.roll = 202;
        r2.height = 5.8;

        //Calling the behavior
        r2.sleep();
        r2.writeCode();
    }
}
```

Steps for creating Object Oriented Programming [BLC & ELC must be in the same package]

Step 1 : Create the Object based on the BLC class [Student] inside ELC class [main method]

Step 2 : Define all the aligned properties and behaviour inside BLC class [Non static field and Non static method]

Step 3 : Initialize all the aligned properties with user friendly value by using reference variable. [r1.rollNumber = 111]

Step 4 : Call the behaviour [calling the methods]

==========