```
//Program on primitive variables

package com.ravi.variables_types;

class Test
{
    static int d = 100;  //Class Variable OR Static Field
    int b = 200;         //Instance Variable OR non static Field

    public void accept(int c)  //Parameter Variable
    {
        int d = 400;           //Local variable

        System.out.println("Class Variable :"+d);
        System.out.println("Instance Variable :"+b);
        System.out.println("Parameter Variable :"+c);
        System.out.println("Local Variable :"+d);
    }
}

public class PrimitiveVariables
{
    public static void main(String[] args)
    {
        Test t1 = new Test();
        t1.accept(300);
    }
}
```

```
//Program on reference variable :

package com.ravi.variables_types;

import java.util.Scanner;

class Customer
{
    static Scanner sc;   //static reference field
    Customer cj          //Non static reference field

    public void accept(Customer cust) //parameter variable
    {
        Customer cj = new Customer(); //local variable

    }
}

public class ReferenceVariable
{
    public static void main(String[] args) {
        // TODO Auto-generated method stub

    }

}
```

Object Oriented Programming [OOPs]

What is an Object ?

* An object is a **physical entity which exist in the real world.**
  Example : Pen, Pub, Car, Laptop and so on

* An object has 3 characteristics :
  a) Identification of the object (Name of the object)
  b) State of an object (Non static field OR properties OR Attributes)
  c) Behavior of an object (Functionality of the object)

* If we want to create an object in java by using new keyword then class name is required.
  Example :

```
public class Student
{
}

Student ravi = new Student();
```

class name — reference variable — keyword for memory allocation — Constructor

What is Object oriented Programming ;

* It is a technique to **design and develop** the programs by using **class and object.**

* Writing program on **real life objects known as Object Oriented Programming and the person who writes the program on those objects are known Object Oriented Programmer.**

* In OOP, We concentrate on objects rather than function/method.

Advantages of OOP :

1) Modularity :(Dividing the bigger task into smaller task)
2) Reusability : (Once we create the components (classes 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/blueprint/template OR prototype** for creating the object.

* A class is **userdefined data type which contains data member(non static field) and member function (non static method).**
  Example :
  public class Car
  {
  Car properties (Non static field)

  Car Behaviour (Non static method)
  }

* A CLASS IS A COMPONENT WHICH IS USED TO DEFINE OBJECT PROPERTIES AND OBJECT BEHAVIORS.

One story on class topic :

Hammer

Blacksmith	Engineer
1 HR : To gather the iron raw material.	1 HR : To gather the iron raw material.
2 HRS : To boil the iron raw.	2 HRS : To boil the iron raw.
4 HRS : To beat and shape	1 HR : To create a dye-Diagram. **class**
7 HRS	3 HRS : To create an original dye **Object**
100 X 7 = 700 HRS	1 HR : To cast the liquid.
	8 HRS
	100 X 4 = 400 HRS

Diagram for Student.java program :


```
package com.ravi.oop;

//Obj1
public class Student
{
    int rollNumber;  //Non static field
    String name;     //Non static field
    double height;   //Non static field

    public void talk()
    {
        System.out.println("Hello learner, I am "+name);
        System.out.println("My rollNumber is "+rollNumber);
        System.out.println("My height  is "+height);
    }

    public void writeExam()
    {
        System.out.println("I am "+name+" every saturday I am writing exam!!");
    }

}
```

```
package com.ravi.oop;

//Obj2
public class StudentDemo
{
    public static void main(String[] args)
    {
        Student raj = new Student();
        //Initialize the object properties (NSF) through object reference
        raj.rollNumber = 101;
        raj.name = "Raj Bourse";
        raj.height = 5.9;

        //Calling the behavior
        raj.talk();
        raj.writeExam();

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

        Student priya = new Student();
        //Initialize the object properties (NSF) through object reference
        priya.rollNumber = 201;
        priya.name = "Priya";
        priya.height = 5.7;

        //Calling the behavior
        priya.talk();
        priya.writeExam();
    }

}
```

Steps for creating Object Oriented Programming [Take a single package]

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

Step 2 :- Define all the object properties and behavior inside
      the BLC class based on your imagination. [Thinking]

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

step 4 :- call the behavior (calling the methods)

Only class level variables (static and non static field) are having default values.