:- Objected oriented Programming :-

OOPs is approach of developing Software in todays worl
In this every component is an object and has two things

a) data : also called attributes / property / data mem

b) function : also called method / behaviour / actions e.t

Syntax of class :-

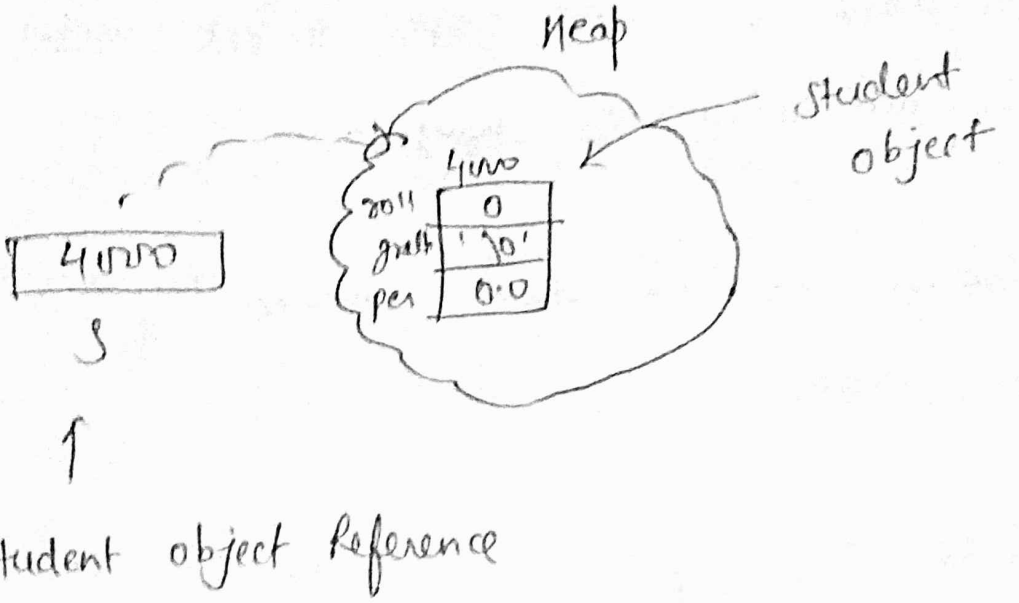
```
<access modifier> class <class_Name>
{
    <access modifier> <datatype> VarName =<value>;
    <access modifier> <datatype> VarName = <value>;
        :
    <access modifier> <datatype> VarName = value;
    <access modifier> return type <Method_Name> <args list>
    {
        // Method body
    }
}
```

```
class Student
{
    int roll;
    char grade;
    double per;
}

class UseStudent
{
    public static void main(String [] args)
    {
        Student S;
        S = new Student();

        S.roll = 10;
        S.grade = 'A';
        S.per = 76.5;

        Sout(+S.roll);
        Sout(+S.grade);
        Sout(+S.per);
    }
}
```



Student object Reference

1) Drawback of this Code :-

Although the previous code Compiled as well as executed successfully, but breaking the Rule of oop's Concept called "ENCAPSULATION".

* Encapsulation says that the data members of class should never be accessed directly from outside the class.

This is because of Security Reason data Can not be accessed outside the classes.

So to make our code proper object oriented code, we will make 3 changes.

1) Declare all the data memember of student class as private.

2) Define method / member function in the student class to initialize and display the values of these data member.

3) From the driver class, instead of directly accessing Roll, grade, per, call the Method defined by the student class.

```
Class Student
{
    private int Roll;
    private char grade;
    private double per;

  public void setStudent()
    {
        Roll = 10;
        grade = 'A';
        per = 78.5;
    }
  Public void ShowStudent().
    {
        System.out.println("Roll = "+Roll+"\n Grade = "+grade+
                                "\n per = "+per);
    }
}
Class UseStudent
  {
    public static void main(String[] args)
    {
        Student S1, S2;
        S1 = new Student();
        S2 = new Student();
```

```
S1.SetStudent();
S2. Set Student();
S1. ShowStudent();
S2. ShowStudent();
}
}
```

```
roll = 10
grade = A
per = 78.5
roll = 10
grade = A
per = 78.5
```