DAY-6

OOPS

Class → Logical Construct
Named Grp of properties & functions.

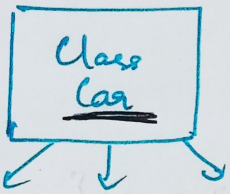
Create class

Syn: class Student {
// code
}

Class is a collection of obj
doesn't take any space on memory.

Class is help to define data type.
Instance of class

Obj → Physical Reality
Occupies space in Memory



Obj
Audi

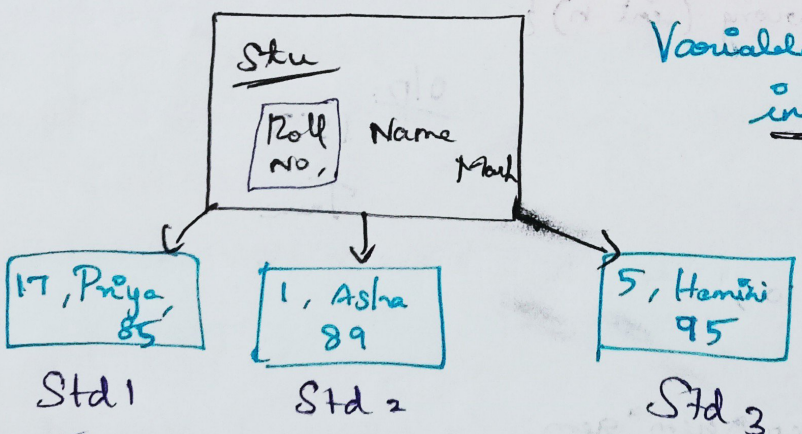
Maruthi
Obj

obj
Ferrari

obj → 3 property
- State of the obj
- Identity " " "
- Behaviour of the obj

(Differ from other)

Eg:



Variables inside obj are instance variables

Access instance variable use . operator

Sout (Std1.rollno) → . called separator

Can access any instance variables.

```
class Std {
    int rollno;
    String name;
    float marks;
}
```

Std std1; → runtime
compile time
Std std1 = new std();

Dynamically allocate memory at run time
Return ref to it.

Syn: Obj class name . obj name = new classname(); → Constructor

Eg: Student Priya = new Student();
System.out.println(Priya);

```
class Student{
    int roll no;      → 0
    String name;      → null
    float marks;      → 0
}
```

As a default value

We need to give or assign values.

Class

Student
rno = 0
Name = null → As a default value
Mark = 0

→ We can create only accessed values

New → Create an obj

Priya
rno = 17
Name = Priya Dhanchi
Mark = 85

→ If value is stored or accessed Here

Using class we can create diff object

Priya.rno = 17 → Mention in obj

↓ Allocate in constructor

Constructor: Basically defines what obj happens when obj is created

Special type of Method whose name is same as class name

Syn:
```
class class_name {
    class_name()
    {
    }
}
```

Automatically call when obj created.
Never contain any return type

By default Constructor

Eg: Student Priya = new Student (17, "Priya Dharshini", 85);

```
Class A {
    ...
}
```

Compiler →

```
Class A {
    A() {
    }
}
```

this Keyword

```
Student Priya = new Student ();
class Student {
    int sno;
    String name;
    Student () {
        this.sno = 17;
        this.name = "Priya Dharshini";
    }
}
```

Priya doesn't use

```
class Teacher {
    String myname = null;
    Teacher () {
        System.out.println ("Hello"); —①
    }
    void setName ( String myname) {
        this.myname = myname; —②
    }
    public static void main (String [] args) {
        Teacher t1 = new Teacher();
        t1. setName ("Priya");
        System.out.println (t1.myname);
    }
}
```

Call

o/p

Hello
Priya

Without Assign value

Eg:

```
public class Teacher {
    String name;
    Teacher() {
        System.out.print ("Hello");
    }
    public static void main
        (String [] args) {
        Teacher t1 = new Teacher();
    }
}
```

o/p:

⇒ Hello

In this keyword is not used

// myname = myname;

Then it call default value

o/p:

Hello
null