

Java for beginners

Classes and Objects



Saurabh Shukla (MySirG)

Agenda

- ① Object Oriented Programming
- ② Key concepts of OOP
- ③ Encapsulation
- ④ Class
- ⑤ Example
- ⑥ Object
- ⑦ Instantiation in C++
- ⑧ Instantiation in Java
- ⑨ Reference variable
- ⑩ Code in Java file
- ⑪ Garbage Collector

Object Oriented Programming

Java is an object oriented programming language.

Real world is full of objects.

Common Noun (~~clone~~) Proper Noun {objects}

Tree

chhotu, lambu,
motu

Car

xumpyasi, khatarci

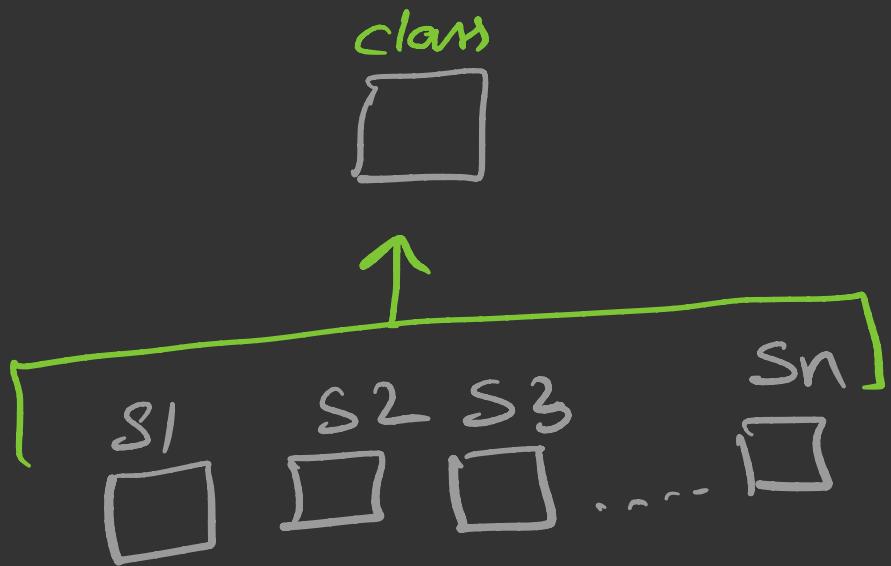
Person

vivek, Atanksha, Rohit

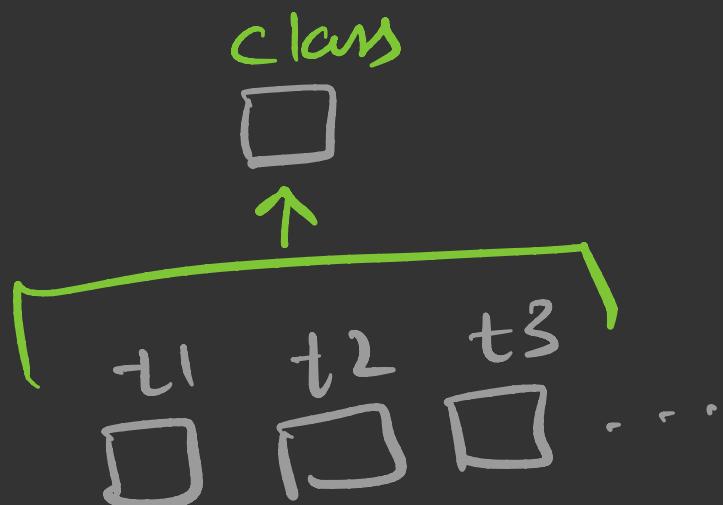


Object is used to represent a real world entity.

Class is a description of object



Object is an instance
of a class.



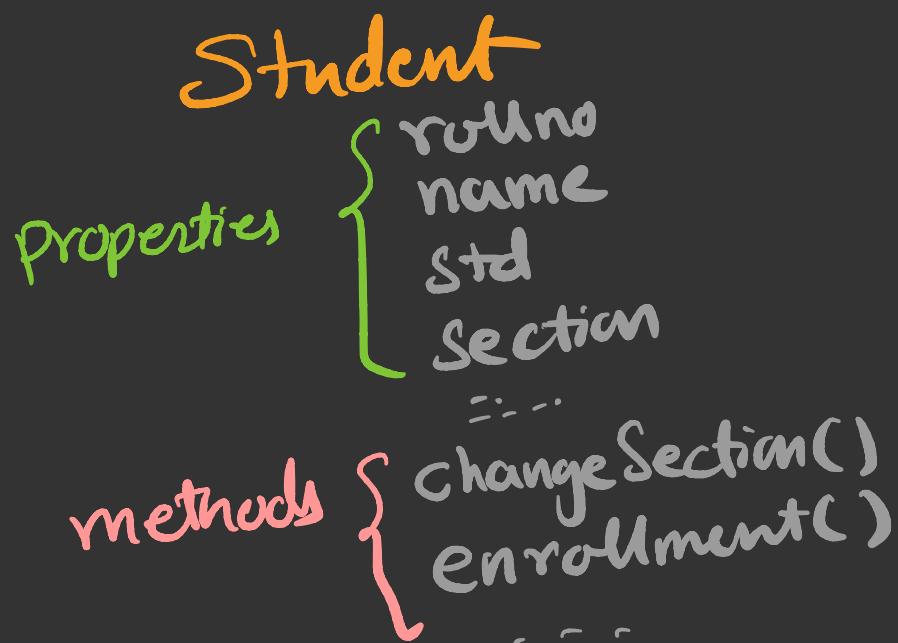
Key concepts of OOP

1. Encapsulation
2. Data Hiding
3. Abstraction
4. Polymorphism
5. Inheritance

Encapsulation

An act of combining properties and methods related to the same entity is called Encapsulation.

Encapsulation can be implemented by defining class.



Classes

- Class is a description of an object
- class is a blueprint of an object
- class == Common noun
- class is a way to achieve encapsulation
- class is defined with the class keyword

```
class className {
```

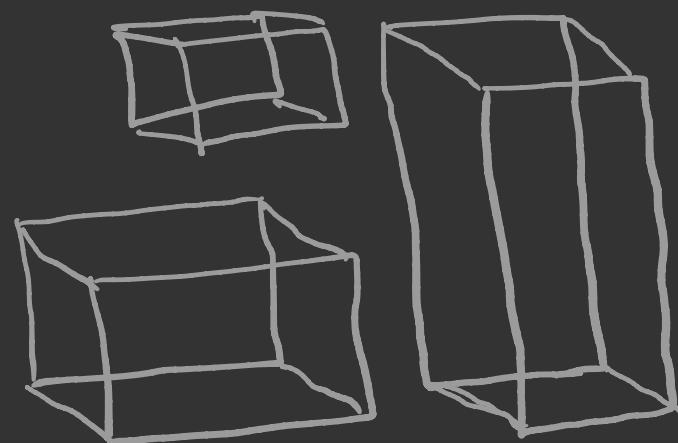
```
    // variables
```

```
    // methods
```

```
}
```

Example

Define a class Box with length, breadth and height as member variables. Also define setDimension() and showDimension() as member functions.



```
class Box {  
    int length, breadth, height; // Instance member variables  
    void setDimension (int x,int y,int z) {  
        length = x; // Local variable  
        breadth = y;  
        height = z;  
    }  
    void showDimension() {  
        System.out.println ("Length=" + length);  
        System.out.println ("Breadth=" + breadth);  
        System.out.println ("Height=" + height);  
    }  
}
```

Object

Object is an instance of a class

Object == Proper noun

Object is a real world entity

Object can be created using class

Instantiation (Creating Object)

In C++, Objects can be created in two ways

① SMA

Box b1;

b1 is an object in C++

b1



② DMA

Box *p = new Box;

No name

or

Box *p = new Box();



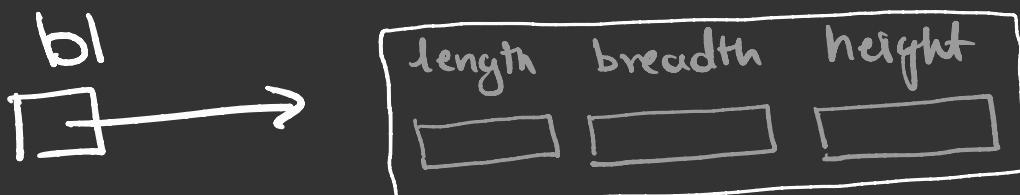
p
→



Instantiation in Java

In Java, object can be created only by DMA.

- { Box b1;
 - b1 = new Box();
 - or
 - Box b1 = new Box();
- b1 is not an object
 - b1 is a reference variable
 - b1 is of nullable type
 - if b1 is a local variable then it is blank when not initialized.



Reference Variable

C++

- Any type
- use & to declare
- Mandatory to initialize during declaration
- Cannot modify

Java

- only class type
- no symbol required
- optional
- can modify

Java file

Option 1

Class Box {

```

int length, breadth, height;
void setDimension (int x, int y, int z) {
    length = x;
    breadth = y;
    height = z;
}
void showDimension () {
    System.out.println ("Length = " + length);
    System.out.println ("Breadth = " + breadth);
    System.out.println ("Height = " + height);
}

public static void main (String [] args) {
    Box b1 = new Box();
}

```

Option 2

Class Box {

```

int length, breadth, height;
void setDimension (int x, int y, int z) {
    length = x;
    breadth = y;
    height = z;
}
void showDimension () {
    System.out.println ("Length = " + length);
    System.out.println ("Breadth = " + breadth);
    System.out.println ("Height = " + height);
}

```

Class Example {

```

public static void main (String [] args) {
    Box b1 = new Box();
}

```

}

Main()

```
public static void main( String []args) {
```

```
    Box b1 = new Box();  
    b1.setDimension( 20, 10, 5);  
    b1.showDimension();  
  
    Box b2 = new Box();  
    b2.setDimension( 15, 12, 10);  
    b2.showDimension();
```

```
}
```

