

Announcement

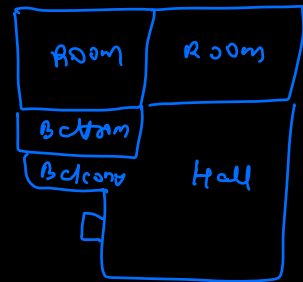
PSS (16th Dec, Friday 9 PM)

Holidays 23rd Dec - 2nd Jan

Class & Object ✓

OOPs ✗

Class → It is a blueprint

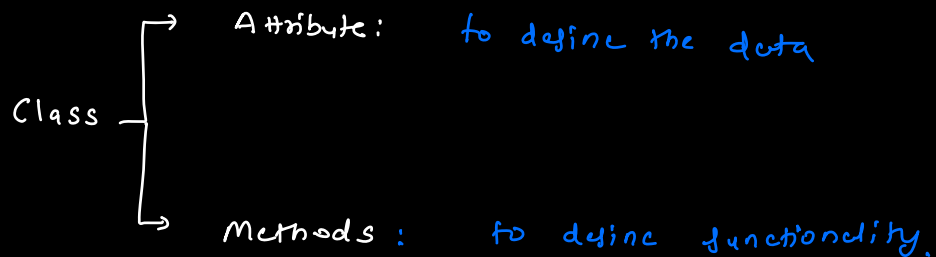


Ex Floor plan of a apartment

Object → It is an instance a class

Ex Actual floor of an apartment

One class can be used to create multiple objects



```

Class car
{
    name
    color
    tyre

    drive() { ... }
    Acc() { ... }
    horn() { ... }
}

```

Car: Yash

Lambo
Red
5

Car: Shreyash

Porsche
black
5

```

drive() { ... }   drive() { ... }
Acc() { ... }     Acc() { ... }
horn() { ... }    horn() { ... }

```

Same functionality across
all objects.

It contains reference to object.

Student s1 = new Student();

reference object.

name: "Rahul"
roll_no: 45

s1.name = "Rahul"

s1.roll_no = 45

print(s1.name)

→ Rahul

```

Class Student
{
    String name;
    int roll_no;

    study() { ... }
    bunk() { ... }
}

```

Attributes →

Methods →

1. Shallow copy

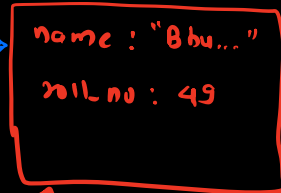
2. Deep copy.

SHALLOW COPY

Student S₂ = new Student();

S₂.name = "Subhas"

S₂.roll_no = 149



Student S₃; NULL

Print(S₃.name)

NULL POINTER
EXCEPTION

Student S₄ = S₂

print(S₂.name) → "Subhas"

S₄.name = "Bhupendar"

print(S₂.name)

→ "Bhupendar"

DEEP COPY

Student S₅ = new Student();

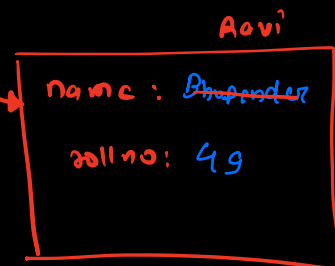
S₅.name = S₂.name

S₅.roll_no = S₂.roll_no

print(S₂.name) → Bhupendar

S₅.name = "Ravi"

print(S₂.name) → Bhupendar



Q. Create a class Rectangle, that supports

- 1) Find the area of the rectangle
- 2) Check is it is a square or not.

$l = 10$
 $b = 20$

Class Rectangle

```
{  
    int l;  
    int b;
```

```
    Rectangle (int lll, int bbb)  
    {  
        this.l = lll;  
        this.b = bbb;  
    }
```

```
    int Area() {  
        return l * b;  
    }
```

```
    bool Square() {  
        return l == b;  
    }
```

```
}
```

Rectangle r₁ = new Rectangle();

r₁.l = 10
r₁.b = 20

Rectangle r₂ = new Rectangle(10, 20)

print(r₂.Area()) \rightarrow 200

print(r₂.Square()) \rightarrow False

Constructor: method used for
initialisation attribute

→ return type None

→ name: ClassName

Q. Add a method to check if area is:

(a) greater than a int k

(b) greater than other Rectangle

Class Rectangle {

|||

function overloading

bool areaGreaterThan (int k)

{
return this.area() > k;
}

bool areaGreaterThan (Rectangle r1)

{
return this.area() > r1.area()
}

Object Reference inside class

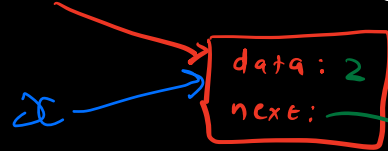
```
class Node
{
    int data
    Node next

    Node(int d)
    {
        this.data = d
        this.next = null
    }
}
```

Node a = new Node(1)

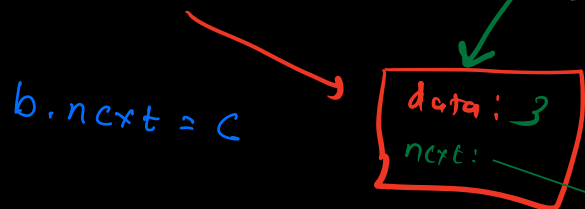


Node b = new Node(2)



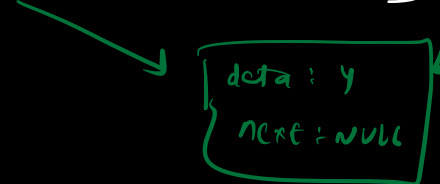
a.next = b

Node c = new Node(3)



b.next = c

Node d = new Node(4)



c.next = d

Print(a.data)

Print(a.next.data)

Print(a.next.next.data)

Print(c.next.data)

