Object Oriented Programming (OOP) In JAVA

① Principals of OOPS (this are basic for all OOL)

Ⓐ Abstraction Ⓑ Inheritance
Ⓑ Polymorphism Ⓓ Encapsulation

you can apply this concepts on all object oriented languages (OOL)

Ⓐ Abstraction → Hiding internal (structural) details an showing only features which can accecessed from outside.
Ex:- just use TV with remote without knowing it's internal.

Ⓑ Encapsulation: In java all things functions and data members are inside class itself so that is encapsulation, from outside all thinks look a single body even it it has many no. of internal components

Ⓒ Inheritance
Bring previous facility modify them & genrate new model from it, this is inheritance
Ex:- Taking features from old Black and white TV & modify it will colour grading & create color TV.
Using older version to derive newer version of something
Ex: Iphone 6 $\xrightarrow[\text{feat}]{\text{Extra}}$ Iphone 7 $\xrightarrow[\text{feature}]{\text{Extra}}$ Iphone 8
→ Reusing of code is main goal of Inheritance

#Specilization & Genralisation

↓

Taking Previous things adding new things in it & create a new Special thing
⇓ (Developing a particular)
Inheritance

↘

Generalisation is if you have few things of same similar type try to take them or under a common ground
⇓ (grouping multiple things)
Polymorphism

→ All student learning java ⇒ Java codes ⇒ Genralisation
Television of company like LG, Samsung, Samsui all are them can come inside one flag Television

→ Grouping things under something Perticular = Generalisation
To handle multiple thing can ~~used for~~ be done with Generalisation

⊕ Class & Object

Every object has two things
Ⓐ Properties (Data members)
Ⓑ Behaviour (function) methods)

- Behaviours will manipulate properties (Data membes can be changed with use of member functions).

- class, it is a blueprint of object, on top of which a object will be get created.

→ objects were constructed on basis of class.
→ You can Create many objects from a single class.

Ex class television[a]

```
    Private int channel;
    Private int volume;     ]→ Data members
    Public void changeChannel(){
          . . . . . . .              ]→ member methods / functions
    }
  public void changeVolume(){
          . . . . . . . . . .        ]→ member methods/functions
    }
```

⇒ Here a class called Television is declared

[3]

```
class Test {
    television a = new television();
}
```

a → object of class television

a → object

new television() → Creating a object with new keyword (constructor)

this is how you can create a object

the object will get created in heap
→ In heap object will get created when you use keyword new

⇒ local variable ⇒ Create stored in stack

⊛ Writing Classes In java

Every class has two parts (A) data members (B) member function — member functions, method

Ex:→ circle — Radius Area ⇒ Circumference

Radius → Data member / Property

After that for each property choose correct datatype related to it.

so you have main class there so create another class

```
class   circle {
    public double radius;
    public double area() {
        _ _ _
    }
}
```

⇒ such like that you can able to create class & use them

```
public class Filename {

    Circle c1 = new Circle();
    — code

}
```

→ creating & using object

⇒ so while declaring you have to specify access specifier for each member function & data member

→ such like that more of you can write and use the data element & It is eas

→ In java there exist by default function & that will get invoked on creation of new object creation.

→ object can able to fetch such functions/methods

too

you can access class from inside or outside of the class.

#Datahiding

So you can see that data of object is hidden while you still able to access methods

Ex:→ Inside of a television set is hidden but you can still manipulate data by simply using buttons

Basically Data is hidden & objects are visible

Access modifier which we often use in classes while declaring class variable & class function can easily used for data hiding

Ex: Public int abc; → can be accessed outside class

Private int def; → can't be able to access out of class

→ Private members can be accessible to only that class i.e function inside that class can be able access them.

So how does we manage private data, so as method can able access private members so just create a method & set values Ex

```
Class rectangle {
    private int l;
    Private int b;
    Public void changeVal(int ll, int bb) {
        l = ll;
        b = bb;
    }
    Public int getval() {
        return l;
    }
}
```

→ setting value (write)

→ getting value (Read)

→ such like that you can assign values without directly accessing

→ you also can read or write internal values with methods

get & set are method for using & handling hidden data.

⇒ Data hiding is very important to make data secure

→ The set method which write through data can acts like filter to avoid incorrectly inserted data.

- * Types of properties

(A) Read & writable → can have both set & get
(B) Read only ⇒ have only get functions
(C) write only → can have only set function

Ex ⇒ roll no. of student / something unique which set only once after that only be readable.

Constructors in java

~~the~~ constructor is used to assign values to object when they are created.

⇒ SO If you create a object

```
Rectangle a = new Rectangle();
```

↓

value to it's properties will be Zero but it should have some value related to them so how can you done it.

→ we want something such like

```
Rectangle a = new Rectangle(10,15);
```

and 10, 15 will assigned to length & breadth.

& should be called when object is created

⇒ Constructor is function who has same name like class

→ Every java class has predefined default constructor if you not defined one

⇒ you can write your own constructor by creating a function which has same name that of class.

Ex

```
class Rectangle {
    private int length; private int breadth;
    //defining constructor
    public Rectangle() {          ⇒ constructor without parameters
        length = l; breadth = b;
    }
    public Rectangle (int l, int b) { ⇒ constructor with parameter
        length = l; breadth = b;        i.e parameterized constructor
    }
```

⇒ you can define multiple overlaped constructor

⇒ The constructor should be functions without return type only

```
Access specifier classname() {
}
```

⇒ that's how you can able to define the constructor

⇒ Constructor don't have any return type.

So now see code

```
Rectangle a = new Rectangle();
Rectangle b = new Rectangle (10,5);
```

⇒ for both case diffsent constructor will be called

Ⓐ Types of Constructor

Ⓐ Default Constructor

Ⓑ Parameterized constructor

Ⓒ Non-Parameterized constructor.

⇒ Constructor function can be overloaded with multiple Ⓐ Constructor function as defined above

you're also able to create objects array and you can be able to store the large amount data at the same time

how the way to create array which is dynamically allocated

datatype variable Name [] = new datatype[Size];

↑ this can be of primitive datatype or any type predefined or user defined classes

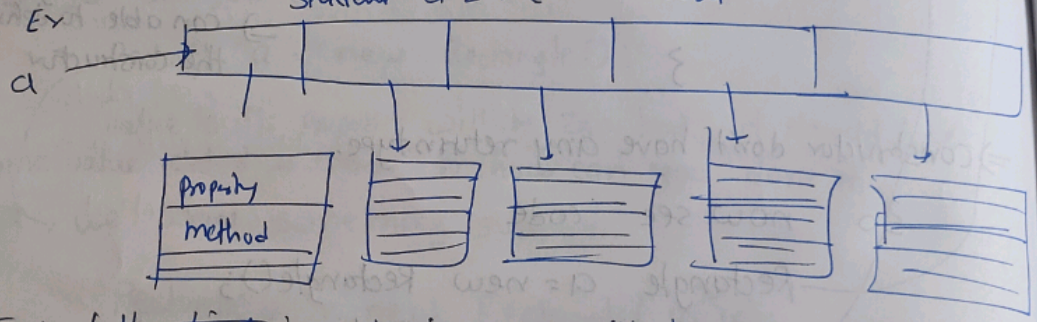
↑ new keyword

size ⇒ you have to specify how ~~many block~~ you want to allocate

new keyword to and allocate memory inside the heap

array[j].function();
array[j].property; ⇒ by using array inde you can access methods and Proper

size 5 ⇒ Student a = new Student[5];



Each of the ~~object~~ is index in array will denote a object + as a whole they will packed & stored in array entity

→ you also able to apply loops on ~~func~~ array