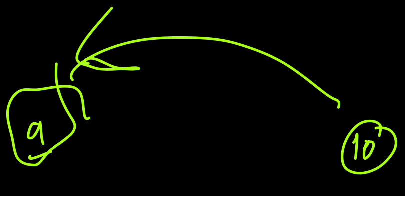
Today Agenda :

1). constructors
   - → Default
   - → Manual / custom
   - → Copy
2). Destructors.
3). Deep copy vs shallow copy.
4). Introduction to Inheritance

Class :- Blueprint of an idea / entity

Object :- Real Instance this class.

Student :-



```
Student st = new student ();
```

Student → Data type

st → variable name

new → keyword in java use to create objects.

Student st = new student ()

student st = new student ()

```
Student {

    String name
    int age
    int roll
    double psp
    String univName
```

Automatically generated

```
Student () {
    name = null
    age = 0
    roll = 0
    psp = 0.0
}   UnivName = null
```

→ Default Constructors :-

if you don't create your own constructor will be a default constructor created.

The default constructor sets the value of each attribute to default value of that data type

→ Manual Constructors :-

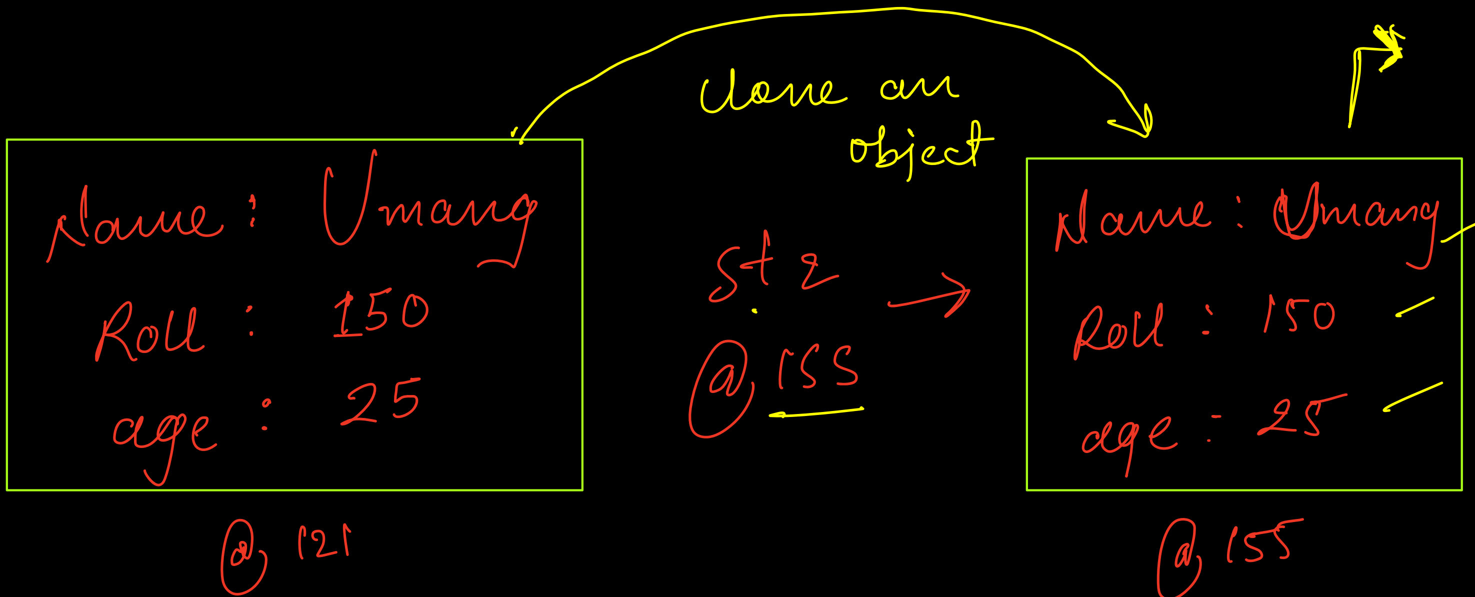
① it initializes the Object with

default value of that datatypes.

(2) Runs the code inside the constructor.

Copy Constructors :-

we already have an Object of student.
we want to create a new object of student with exact same value as elders object.



clone an object

Name : Umang
Roll : 150
age : 25

@121

St 2
@155

Name : Umang
Roll : 150
age : 25

@155

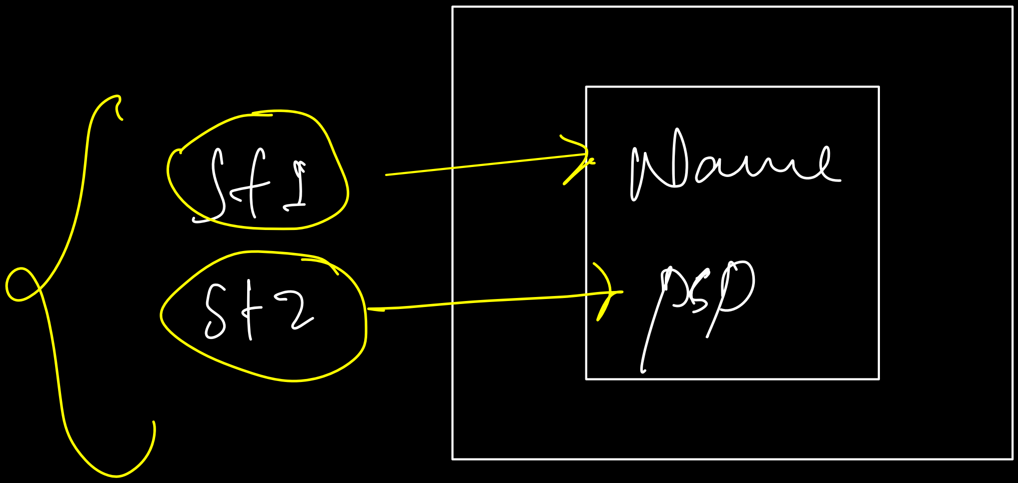
```
Student {

        age                         new — ();
Private name;
        UnivName
        roll
                                    ↓
    student ( student st) {
      this. age = st. age
      this. name = st. name
    }


    Client {

            student st1
                                    }

          student st2 = new student
                                   ();
```

x st2. name = st1.name

st2 . roll = st1.roll

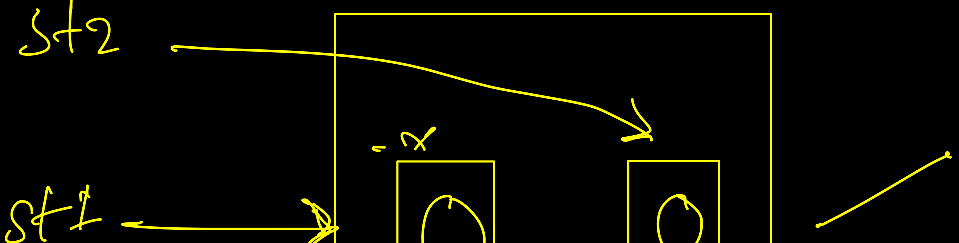
student st3 = new student(st1)

Shallow copy.



Student st2 = new student(st1)

Deep copy.



Contineous after the break Train

⇒ Shallow copy :-

whenever I have Created a new object but behind the scenes the new object still refers to a few attirubutes of old object.

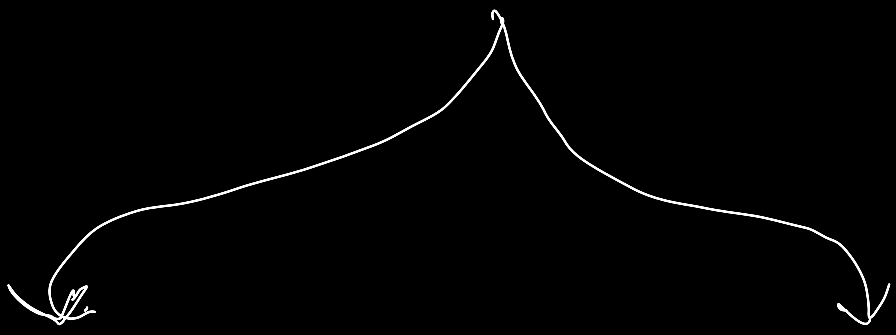
New and Old object still share some Reference

Deep Copy :- No shared Reference

sf2 }
sf1 }

In java, Deep copy is not even practically possible

Types of data & where are they stored in memory ?



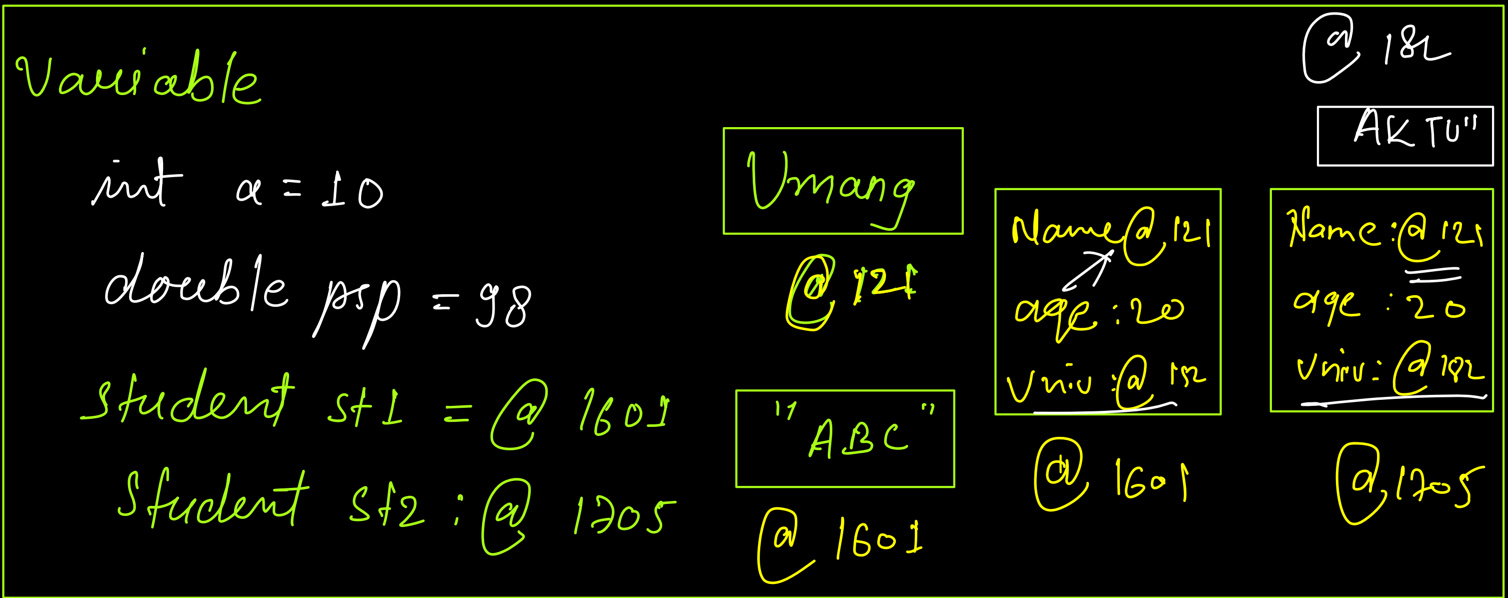
Primitives types
(int / double / float)

Exist as variable which store the values directly

Object
(string)

Object are stored memory variable are store the adress of the

Object.



Variable

int a = 10

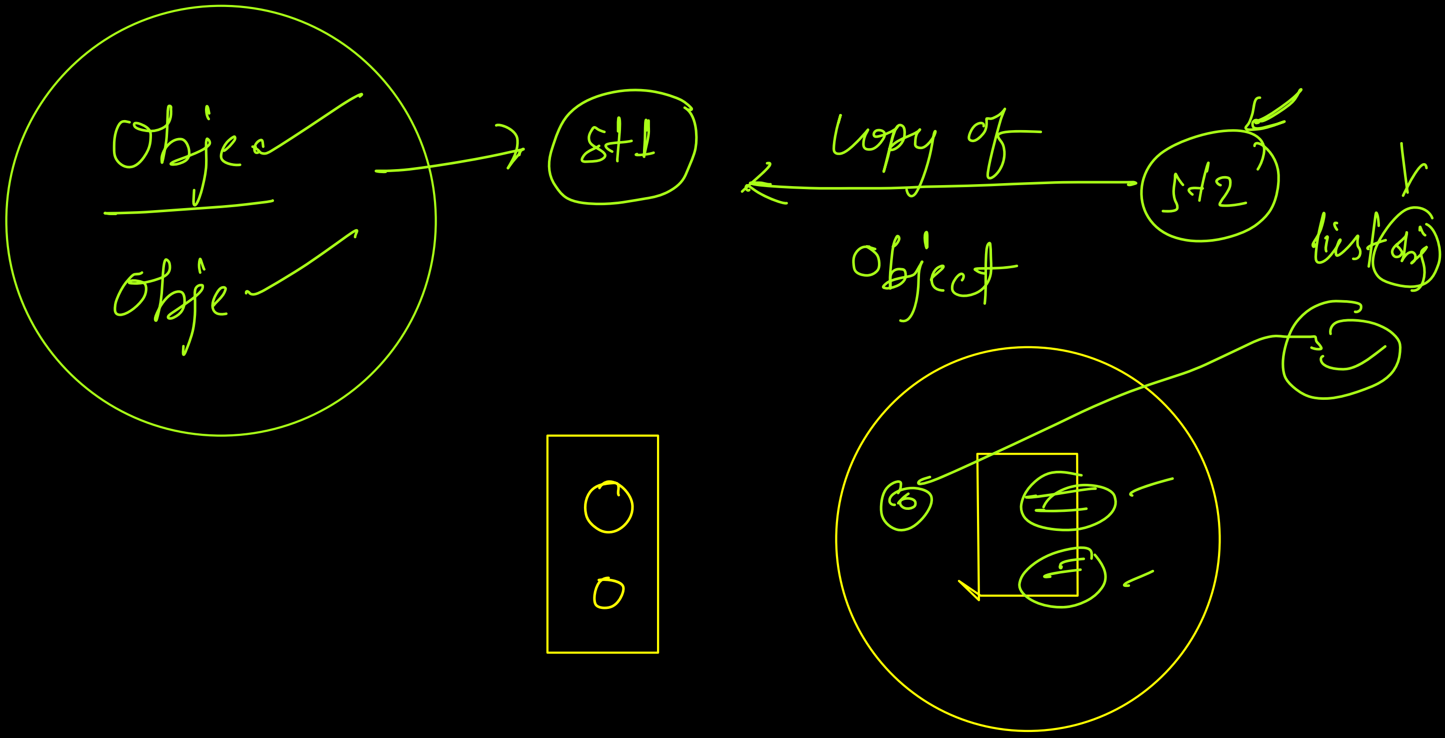
double psp = 98

student st1 = @ 1601

Student st2 : @ 1705

St2. univ = " AKTU "

st2. Univ. add (" EF ")



Shallow Copy.

Object → obj → obj'

Recursive

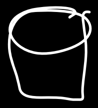
Destructors :-

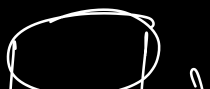
↳ Opposite of Constructors

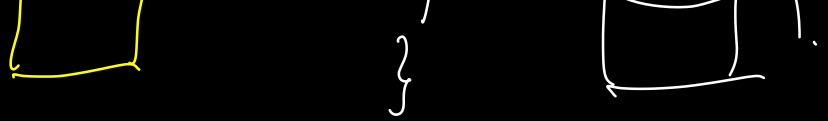
↳ will be automatically called if Object is no longer needed to free up the space that is being taken by the Object.

Pass by value vs Pass by Ref ?

Pass by Ref

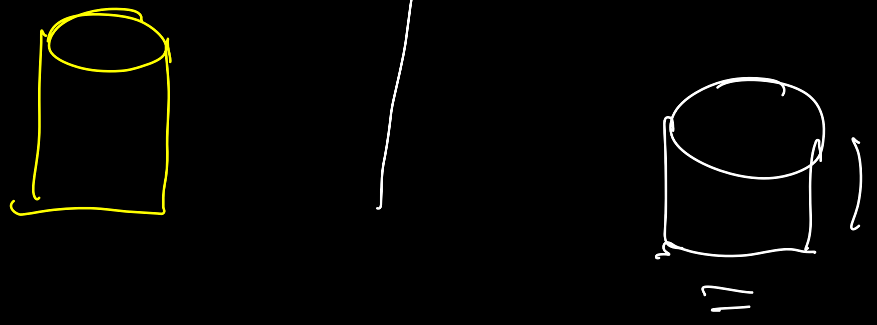
fill coffee (){





Pass by value

fill coffee (⌸){



Java is pass by value

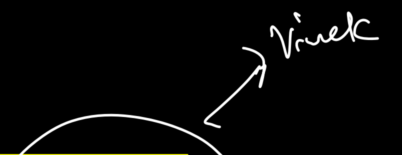
do somthing (student (st1) {
@152

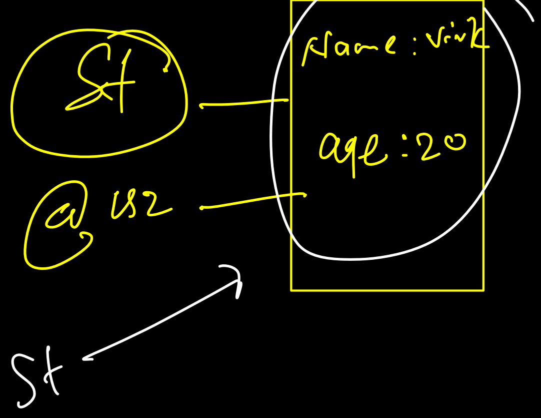
St. name = " Vivek "

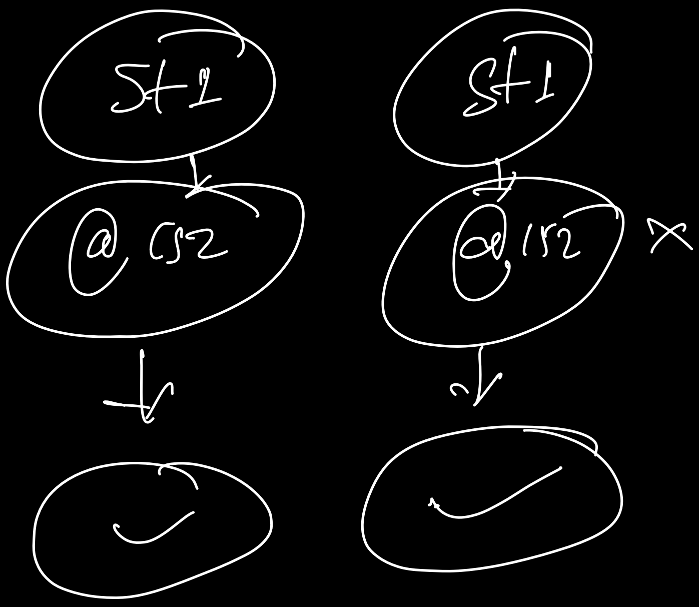
St = new student(); x

}

Client { do somthing (st1)




St
@ 152
Name : Virat
age : 20
St


St2
@ C52
St1
@ 152
X