

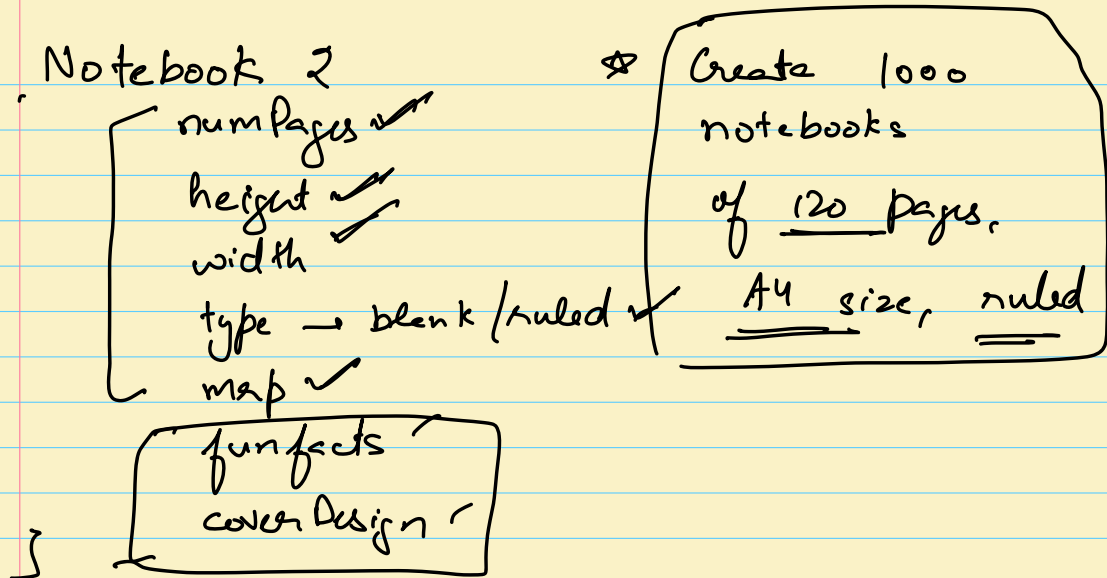
1. Good Evening
 2. Lecture begins at 9:10 pm
 3. Topic
 - ↳ Prototype & Registry Design Pattern.
-

Agenda

1. Prototype Design Pattern
2. Registry Design Pattern

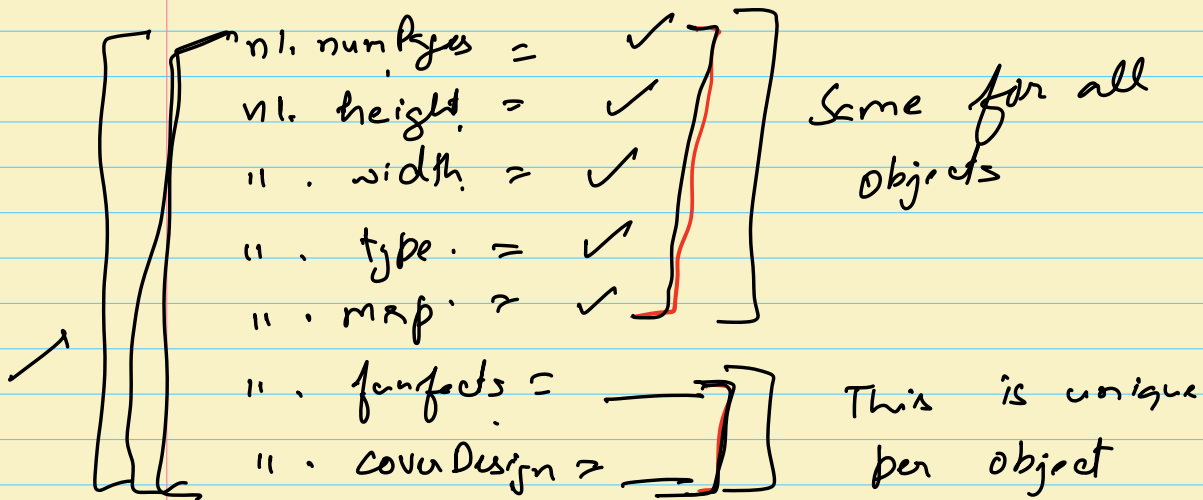
Problem Statement

1. Given an object of a class
2. We need to create a copy of the object



Wdy 1

Notebook n1 = new Notebook()



→ Create each object independently. & set all the properties

Weg 2

1. Create a prototype ✓
2. Store the prototype in registry ✓
3. On object demand ✓
 1. Get prototype from the registry
 2. Clone it.
 3. Make changes { for unique }
properties
 4. Return the object

Chrent

```

class Notebook {
    Notebook orig = NotebookRegistry.get(
        "A4_120_ruled");
    Notebook copy = orig.clone();
    copy.funfacts = 9;
    copy.coverDesign = 9;
}

```

A3 - 200 - blank

★ Registry can be implemented as singleton.

Need for prototype & Registry

1. Scenarios when multiple objects are required, which have some unique properties, and a lot of properties are similar

How to create Prototype & Registry

(✓)

```
Student ? pvt id;
{
  * name;
  * age;
  * psp;
} public

{
  # batchName;
  # instName;
  # avgBatchPSP;
} private
```

Client

psvm

```
Student org = ✓
[Student copy = org;]
copy.age = 10 [org.age?]
```

copy in this case is just a new reference for the same instance. If something changes in copy, org also changes.

V2

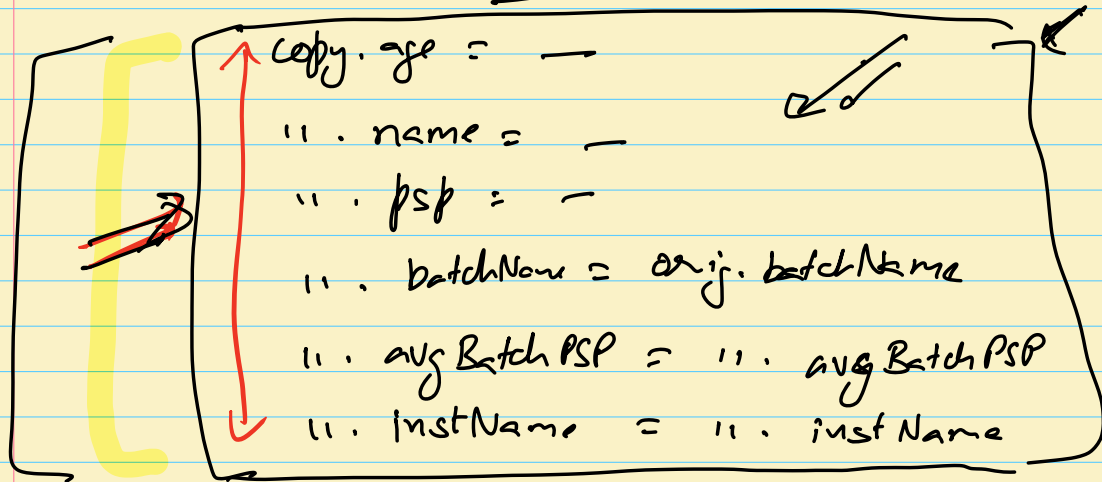
1. Same student class

Client

psvm

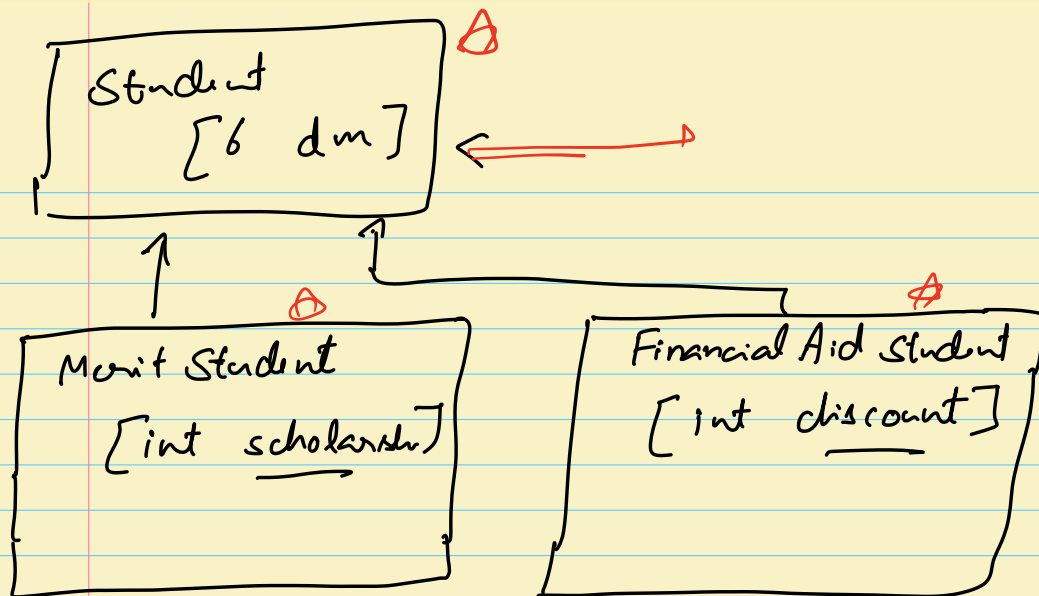
Student orig =

Student copy = new Student();



Problems

1. Private Data Members ✓
2. Client's SRP is being violated ✓
3. OCP violation in case of inheritance ✓



Client

Student org =

Student copy = null

if (org instance of Student) 6 dm

copy = new Student() ✓

elif (org instance of MeritStudent)

copy = new MeritStudent() ✓

6 + scholarship

elif (org instance of FinancialAidStudent)

copy = new FinancialAidStudent() ✓

6 + discount

elif

V3

COPY CONSTRUCTOR

Student

Student(st-)
{ t.id = st.id

Problem → OCP is not solved. If-else
block will still remain in
client

Client

Student obj =

Student copy = new _____

```
Student
    private id;

    Student(Student o)
    {
        this.id = o.id;
    }
}
```

Client

* private dm are visible inside
the class [ctor]

[Not only of the current obj, of
any object of the same class.]

Why OCP not solved?

Client

void fun (Student obj) ?

Student copy = new Student(obj)

or

= new MStudent(obj)

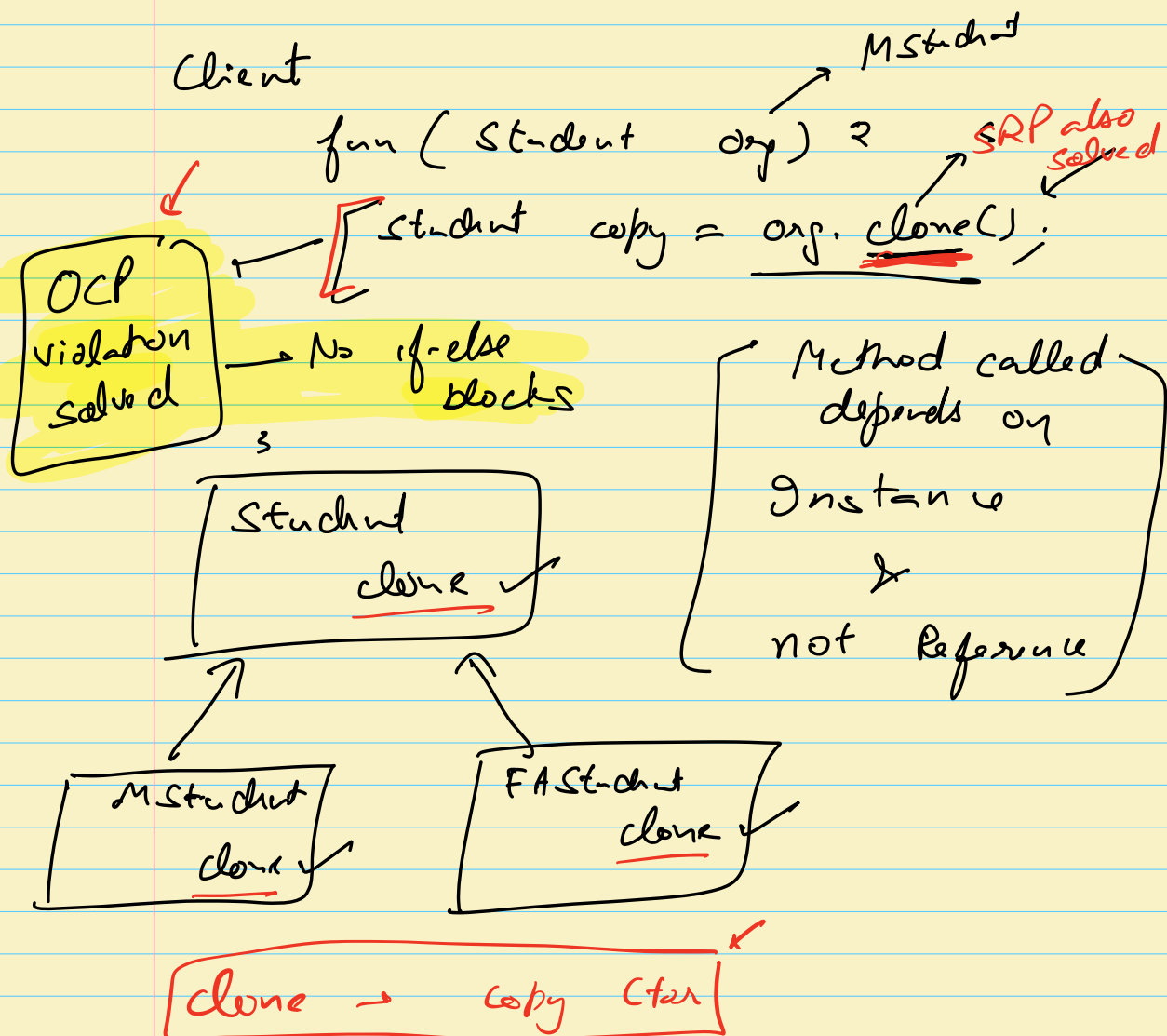
or

= new FAStudent(obj)

if-else blocks → OCP is
violated.

★ Copy ctor can't be used
of OCP violation

→ clone Method inside
Student
MStudent
FA student



Break → 10:20 — 10:25

↳ code [clone & copy ctor]

Deep Copy vs Shallow Copy

class A {

int x; ✓

int y; ✓

B obj ✓

class B {

int z; ✓

C oc ✓

}

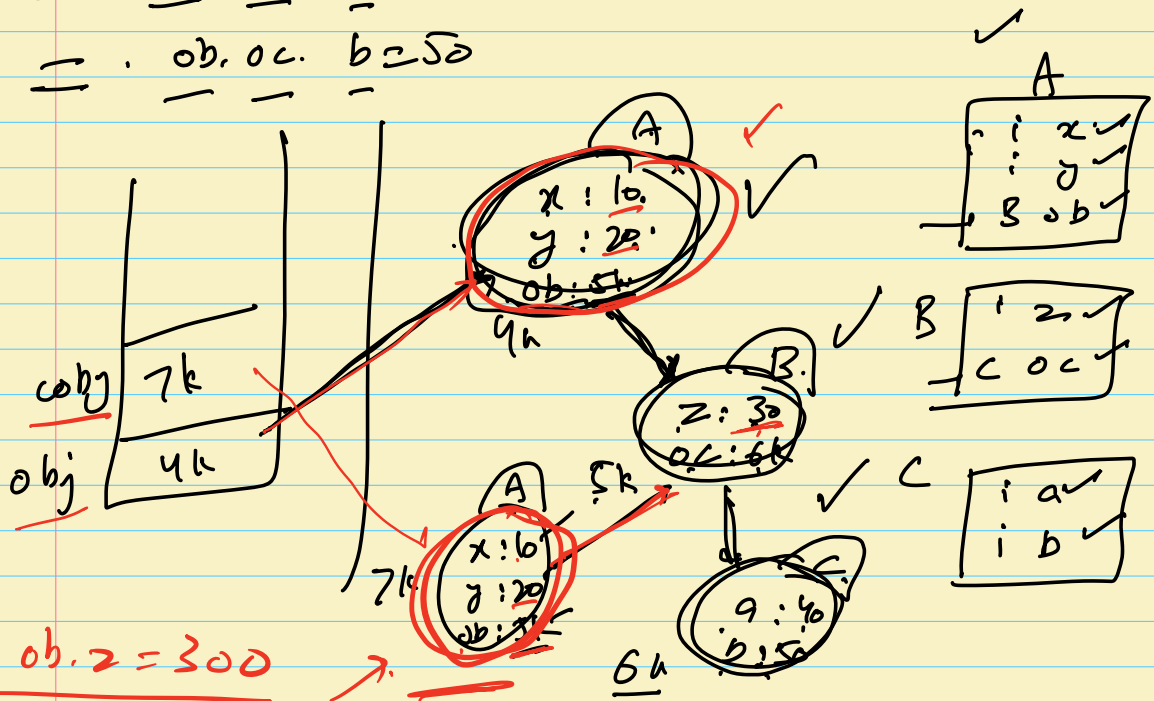
class C {

int a;
int b;

}

A obj = new AC()

obj.x = 10 ✓
 — .y = 20 ✓
 — .ob = new B() ✓
 == .ob.z = 30
 == .ob.oc = new C()
 — .ob.oc.a = 40
 — .ob.oc.b = 50



copy.ob.z = 300

[obj.ob.z ?]

A True deep copy of A

A copy = new AC(),
 copy.x = obj.x ✓
 — .y = obj.y ✓
 — .ob = obj.ob ✓

$x: 10$
 $y: \text{"Sumeet"}$
01

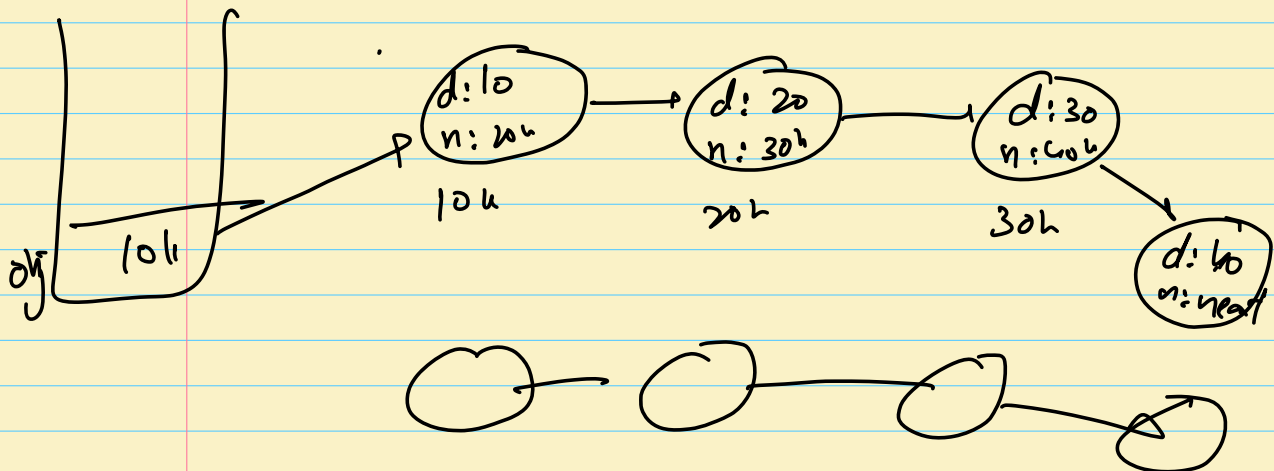
$x: 10$
 $y: \text{"Sumeet"}$
02

$02.x = 01.x$
 $02.y = 01.y$

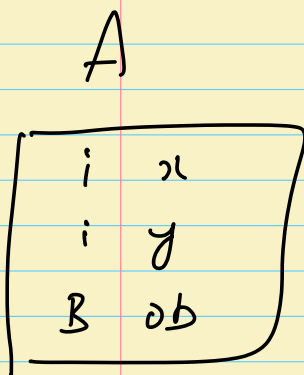
class Node
{
 d
 next

}

→



Shallow vs Deep



clone() {

A copy = new A()

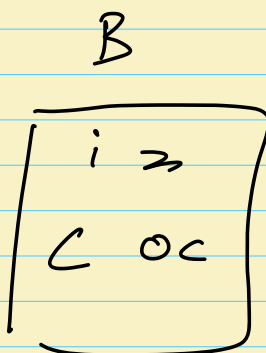
copy.x = this.x

copy.y = this.y

→ .ob = this.ob.clone()

return copy

}



clone() {

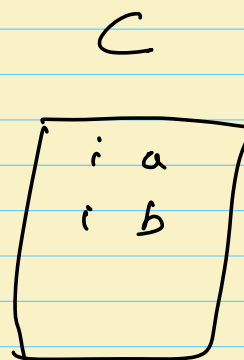
B copy = new B()

copy.z = this.z

copy.oc = this.oc.clone()

return copy

}



clone() {

C copy = new C()

copy.a = this.a

→ .b = → .b

return copy

}

A

i x ✓

clone

s y ✓

d z ✓

B ob; ✓

To implement dup copy what
should of clone method be like?

A clone () ?

A copy = new AC)

all
the
primitiv
&
string

[copy . x = this . x
 y = y]

2 [copy . ob = this . ob . clone () ;

Node ?
i d

Node left ;

Node right ;

↓

Notebook

Registry

↳ The place where
samples or prototypes are
stored

→ register (s k, Student obj)

→ student get (string key)

→ Map <String, Student>