

Design Patterns

: Creational Design Patterns

Agenda

- ① what are design patterns
- ② Types of design pattern (3)
- ③ Creational Design Pattern
 - a.) Singleton
 - b.) Builder
 - c.) Factory

```
graph TD; A[Practical factory] --> B[a.) Singleton]; A --> C[b.) Builder]; A --> D[c.) Factory]; C --> B; D --> C;
```

Design Patterns

d.) Prototype  → Abstract Factory



→ Solution to common software design problems

→ Types of DP



→ how to create an obj:

→ where to mate an obj

- how many objects to create
- Singletone, factory, Builder

② Structural DP

- How to Structure our classes
- what all classes will I have
- what attributes will be there in every class
- How are > classes going to be connected
- Decorator, Adapter

③ Behavioural DP

- How will classes interact with each other



Other

→ Strategy,

Why learn DP

- ① Common Terminology b/w SWE
- ② Give you the worth of good software design.

Creational DP

① Singleton Design Pattern

You have a project

↳ consists of hundreds of files /
classes

↳ I want to interact with DB

In a normal codebase

In every file

I Database db = new Database();

≈ also going to invoke a TCP co

Class A↑ Ø

Database db = new DB();

db - Save()

Class A↑

Database db1 = new DB();

db1 - Save()

}

↓ YOB

100 DB Objects

,

Linear sum → 360 ↗

100 * 100 ~~KB~~ \Rightarrow 2000 00 B

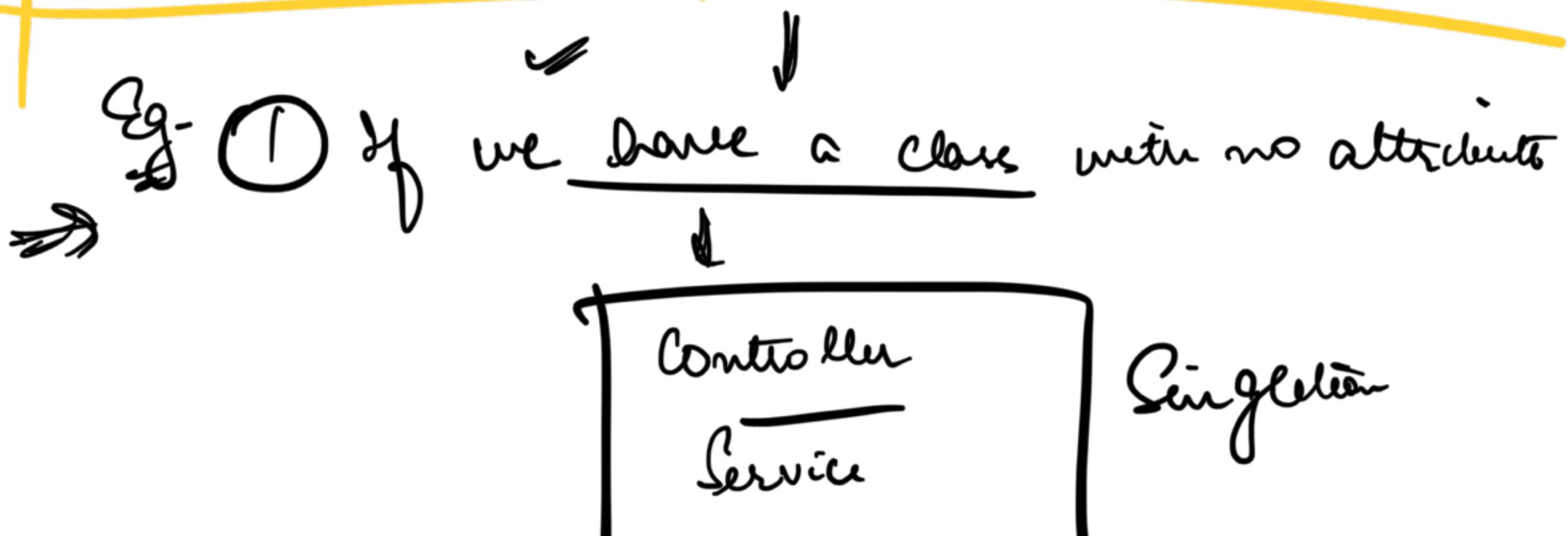
\rightarrow 200 KB

\Rightarrow 0.2MB

\Rightarrow wastage of resources

SITUATION

We have multiple objects of a class where every object is going to behave EXACTLY same. In that case, it makes no sense to have multiple instances of that class.



Repository

class UserService {
 register() {}
 getDetails() {}
}

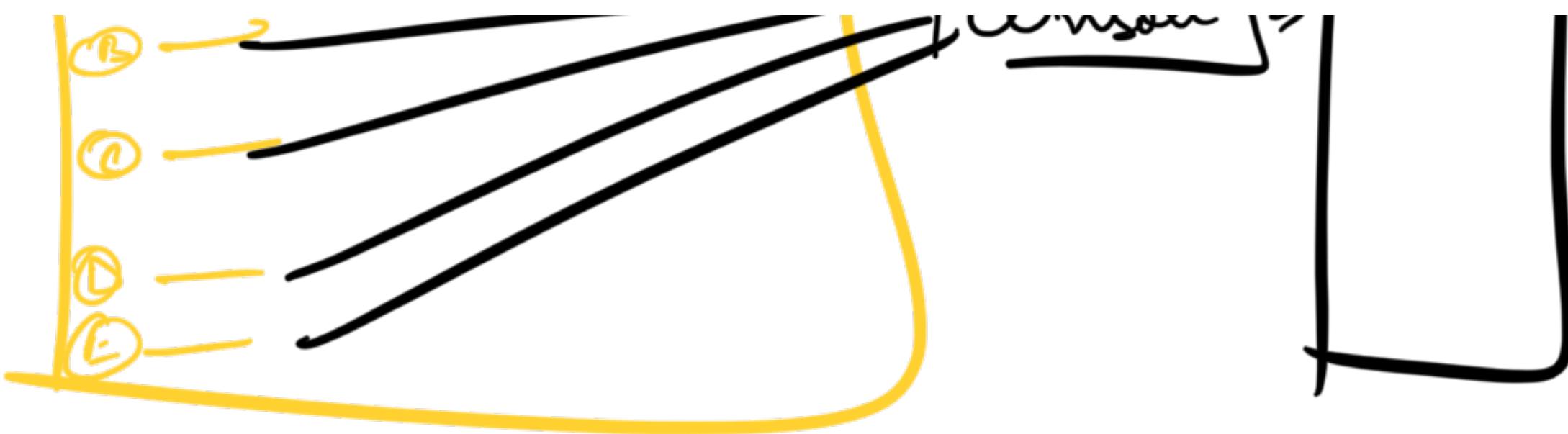
② ~~Shared Common Resou~~

behind the scenes.

logging

ORM





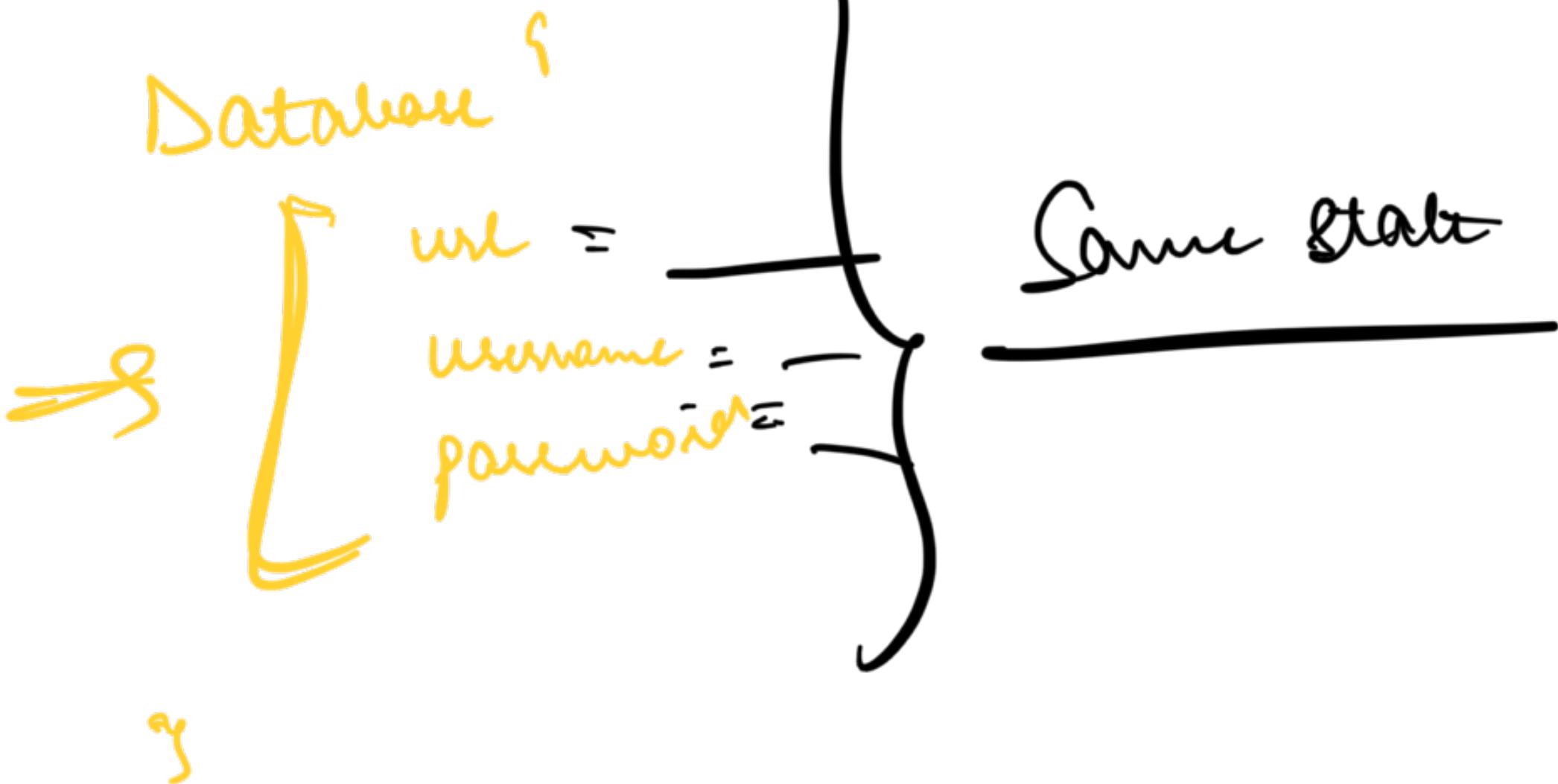
Singleton : A Class which can have AT MOST

One instance

③ Every instance of that class is my

App[~] will have EXACTLY Same

attribute



HOW WILL I CREATE SINGELTON

Q: If I have a public constructor of a class.
Can I ever make the class Singleton?

NO → Everytime a constructor is called,
a new object is created.

```
Database {
```

```
    private Database() {  
    }  
}
```

→ Now I can't even
create 1 instance

}

Database.getInstance()

Database {

~~private~~ static

Database instance = null ;

private Database () {}

static

Database getInstance() {

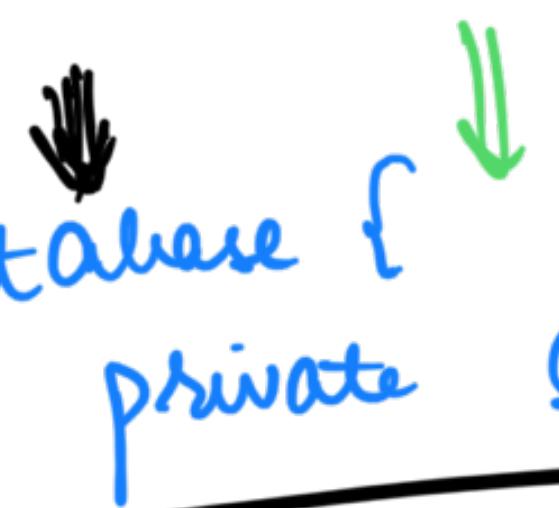
if (instance == null)

```
    =>     instance = new Database()  
    }  
    }  
    return instance;  
}
```



This method needs to ensure
that only one instance of
Database is created ever

```
Database {  
    private static Database instance = null;  
    ① private Database() {}  
        not called
```



~~public static Database~~ ^{synchronized} ~~get instance () {~~

~~if (instance == null) {~~

~~instance = [new Database]~~

~~}~~

~~return instance;~~

}



instance = new db @ 123

Thread 1 Thread 2

```
get Instance () {  
    => if (instance == null) {  
        => new db@456  
        instance = new Database();  
    }  
    => return instance;  
}
```

```
Client {
```

```
    Database db = Database.getInstance();
```

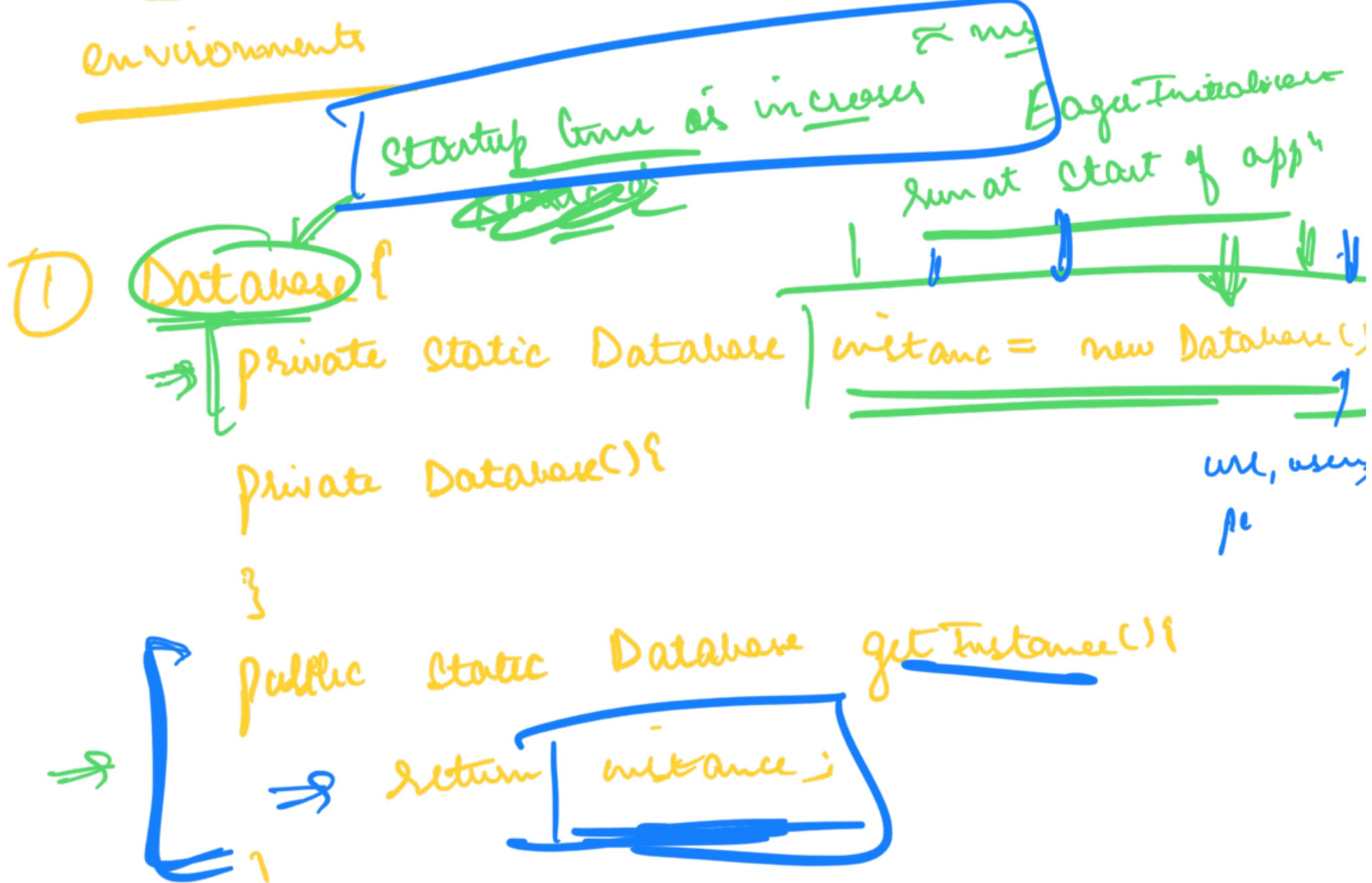
```
,
```

11. ... small. A solution in multi threaded

- ① -> Tim
- ② + Tim
- ③
- ④
- ⑤
- ⑥

How well I move things

environments



3

2nd con: what if to initialize the obj

I need to give few paramet



DOUBLE LOCKING

Database {

```
private static Database instance = null;  
public Database() {}
```

~~Synchronized~~

① → public getinstance () {
② ↗ Cyn changing {
if (instance == null)
instance = new Database ();
return instance;
}
} ↗ 1st check w/o lock
↗ 2nd check after lock

}
Database {
private static Database instance = null;
} Double Check locking

get Instance () {
if (instance == null) {
Synchronized {
if (instance == null) {
instance = new Database ();
} }
} }

```
(  ) --> instance = new Database()
           |
           v
instance;
}
}

? }
```

Washroom

P1: ① Is anyone inside?

P1 got a phone call

P2: ② Is anyone inside

~~Phone~~

P2: ③ P2 went in washroom

P1's thread might
be paused by
Scheduler

P1 ended the call

PI: ① PI went

Google about
how to implement Singleton classes that
allow serialization

Builder Design Pattern

- Class has multiple attributes (12)
- All objects of the class "should be created" ||

How will you create that class.

Can't change

Value of

attribute once

object is

created.

Observation

- ① All attributes of class should be private
- ② I should be able to set the value of those attributes at time of creation
(construction)

⇒ } should take all values via constructor

Class A {

int a;

 int b;

int c,

String d;

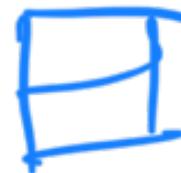
Change e)

A(

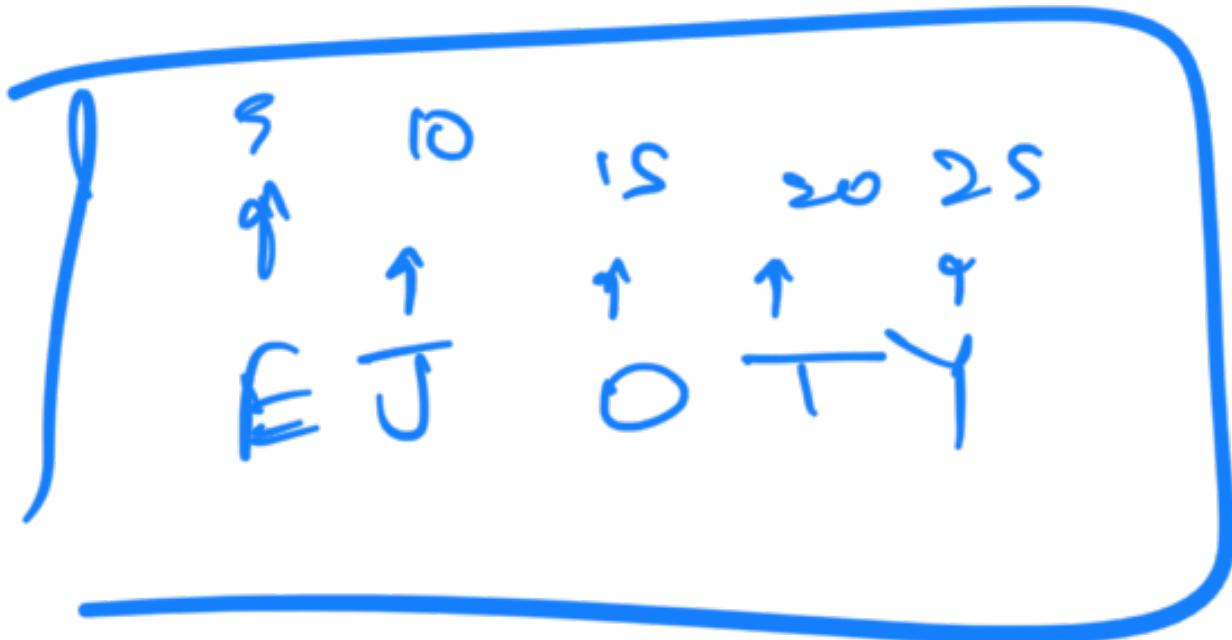
int f;
 int g;
 int h;
 int i,
 int j,

int k
 int l;

)



5



Client 1

A a = new A(1, 2, 3, Ave, Hello, 5, 6, -);

1

class Sandwich { }

 int noOfBread,

 int noOfPickle,

 int noOfOnion;

x x

↓

Sandwich(int noOfBread, int noOfPickle, int noOfOnion)

}

Sandwich S = new Sandwich(5, 2, 1)

n

e.g. other via constructor / passing

Passing multiple
Attributes of some type via Constructors

phone to 

→ I Can't use constructor

- ① Class with multiple attrs
- ② Objects should be immutable
- ③ ~~Look~~ Can't do use constructor because

unreadable word

- Instead of passing every param separately,
- O → DS that allows us to store multiple values
- FlashMap

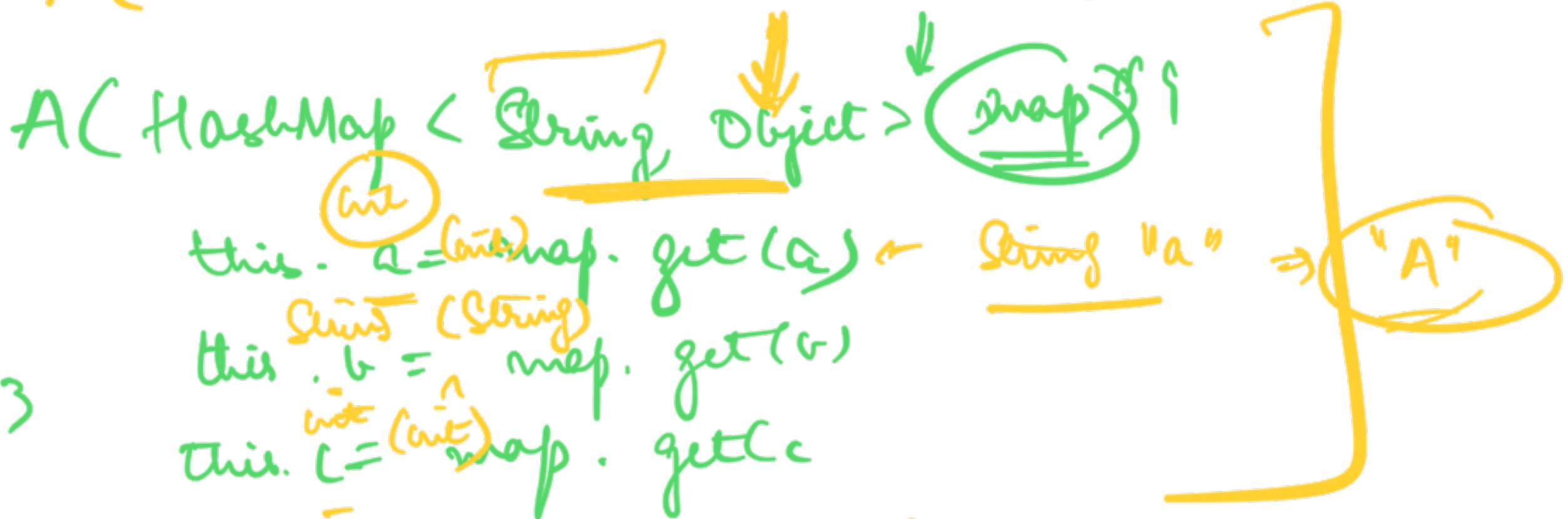
A {

 bit a; intcc;

 bit b;



A (int a, int b, int c,)



① ClassCastException while creating obj

② Case typos in the keys

Do you know a language construct which

allows to store multiple value of diff types

and be able to fetch / set those values

separately via some Name

```
class X {  
    String a;
```

```
    String b;
```

```
    int c;
```

```
    double d;
```

{ }
Open

}

X x = new X()

x.a = "Hello"

x.b => hi!

x.c = 1

x.d = 2.

x.A = "Hello"

class Sandwich {

class Builder {

pvt

int noOfBreads)

pvt

int noOfOnions)

pvt

int noOfPickles;

Sandwich (Builder b

this.noOfBreads = b.noOfBread

this.noOfOnion = b.noOfOnion

} this.noOfPickle = b.noOfPickle.

}

int noOfBread -
int noOfOnion
int noOfPickles

1 - int

{



b.noOfOnion

b.noOfPickle

Builder or new Builder.

↳ no of breads = 10

↳ no of Dres ↗

↳ no of Pickles = 2

Sandwich c = new Sandwich (b)

Immutability → Object in the various can
change. Don't do ok

Sandwich {
 wt no of bread
 wt no of Onion
 wt no of Pickle

(not)

Sandwich(Builder a)

this .no of Bread = b . no of Brea

Builder {
 no of bread
 =

Never make attrs public

Always use getters setters methods

Builder {

 private int ~~noOf~~ bread

 private int ~~noOf~~ pickle

 private int ~~noOf~~ Onion

 public int getNoOfBread() {
 return ~~noOf~~ bread;

 } *1st One*

}

 public void setNoOfBread(int noOfBread)

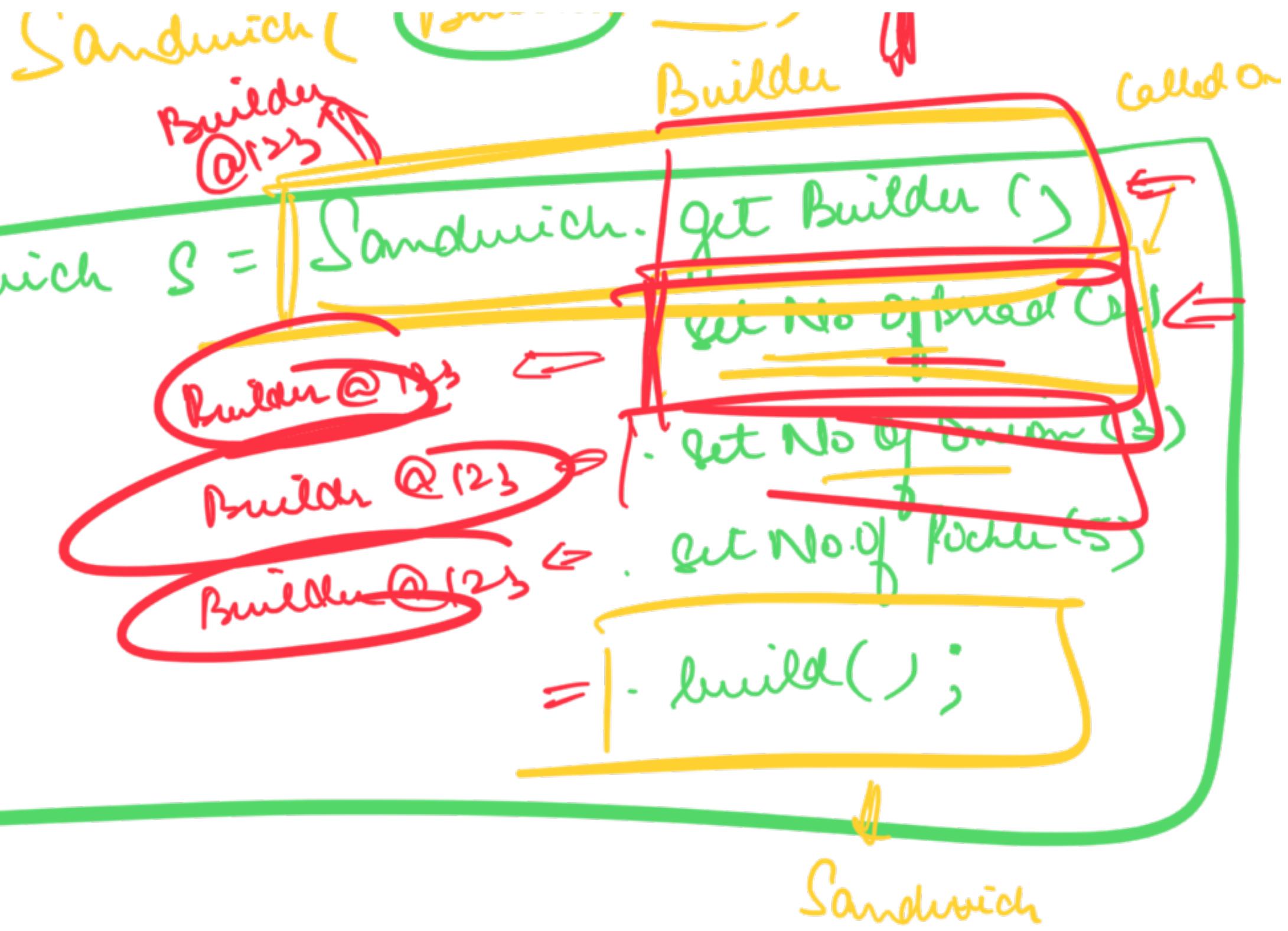
log[xx] is changing
true . no offread = no of read()

$$\times \frac{a}{a} = \frac{l}{l}$$
$$B \leftarrow \frac{l}{l}$$
$$C \leftarrow \frac{l}{l}$$

Buildin $b = -$

\equiv

Buildin



Builder {
 private int noOfBread

```
private int noOfOnion;
private int noOfpickle;
```

```
int getNoOfBread() {
    return noOfBread;
```

```
}
```

```
Builder SetNoOfBread (int noOfBread) {
    this.noOfBread = noOfBread.
    return this;
}
```

```
}
```

```
Sandwich build () {
```

```
    Sandwich S = new Sandwich (this) =
```

```
    S.noOfOnion = noOfOnion;
    S.noOfpickle = noOfpickle;
    S.noOfBread = getNoOfBread();
    return S;
```