

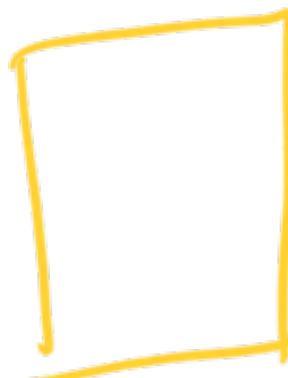
SOLID

- ↳ SRP \Rightarrow One responsibility
- ↳ O/C \Rightarrow Open for ext
Closed for modif.
- \Rightarrow LSP \Rightarrow no special handling
for subclasses
- \Rightarrow ISP \Rightarrow no fat interfaces
- \Rightarrow DI \Rightarrow high level modules
should not depend
on each other

Ready made Solutions

for more complex
use cases

Design Patterns



def calculate_all(A):
 ac = []
 for i in A:
 ac.append(i)
 return ac

→ through on loop

Java → SRP

→ utils

Design patterns

Creation al

- ① Singleton } \rightarrow Double check locking
 - ② Builder \rightarrow Telescopic const.
anti-pattern
 - ③ Factory
 - ④ Prototype
-

~~A { a
b
c }~~ A(a,b)

$c \quad d$ → $A(c, d)$

Polymorphism

— Ctor overloading

Design Patterns

→ Solutions to commonly existing
problem

→ Gang of Four

Do or

Don't repeat

Try yourself

How can I create a single instance
of class?

Singleton

Creation



How to create
object

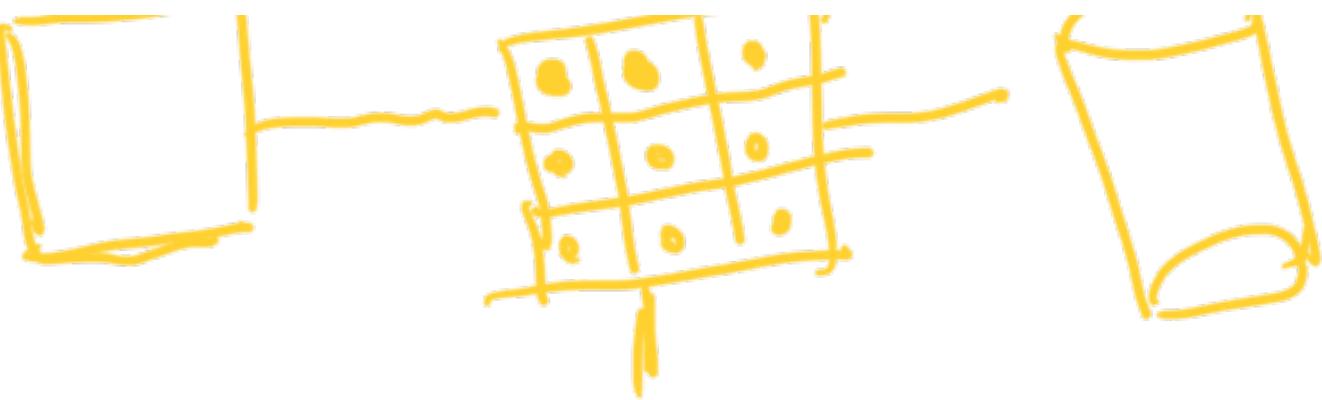
How many objects
to create

Motivation

- * Shared resource

Connection pool





Connection Pool

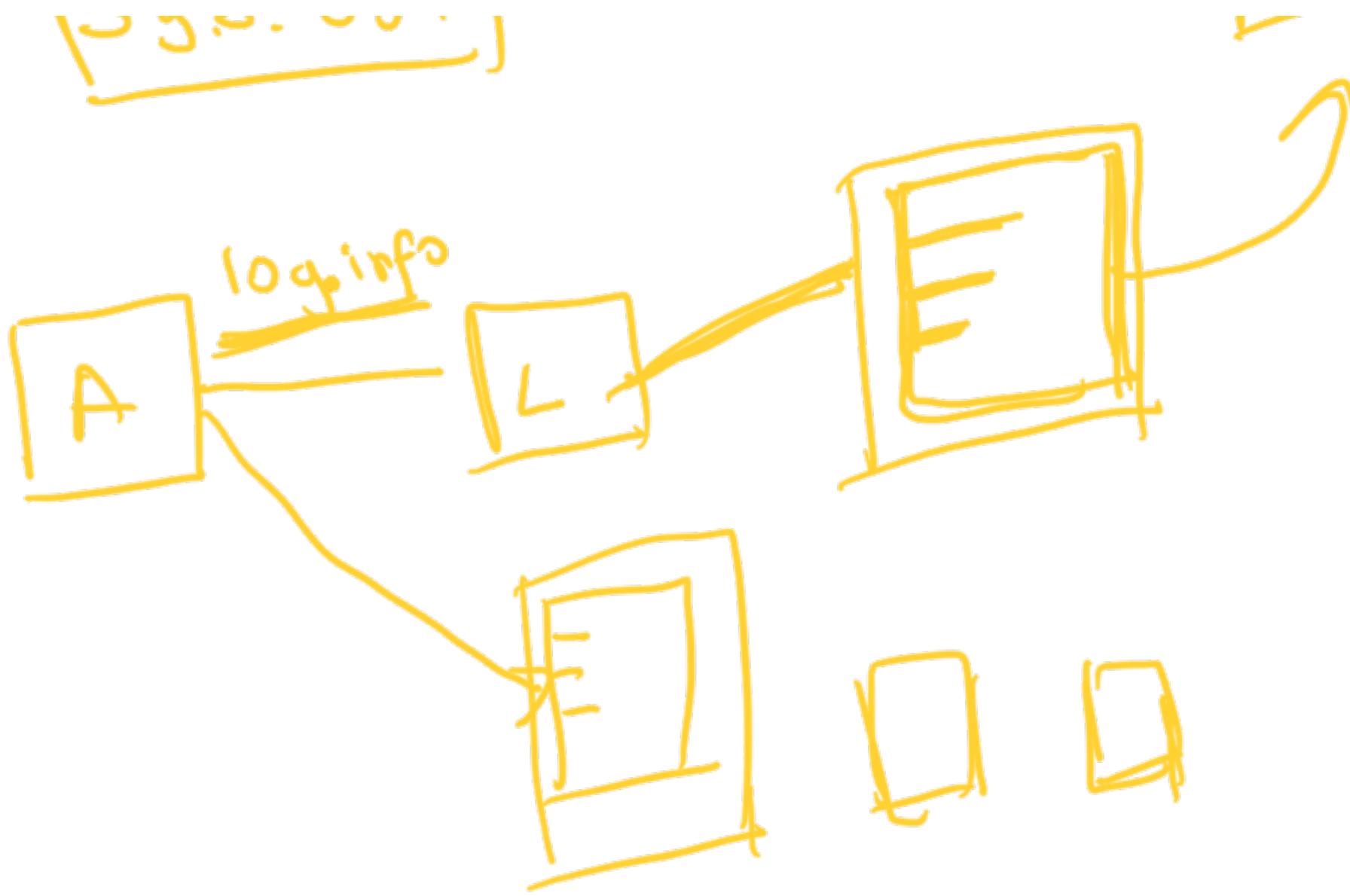
List of Connections

} 10
100

Logging - Logging library

Cust.out

S3



Shared resources

→ Only write once in store



→ file → file

~~Con pool~~

jj v'

info

error

configurat

} host :
url :

Connection pool size :



Single access point

Singletton

→ Create only one instance
of a class

Spring → instance

→ singletton
→ Beans

How)

①

Initialisation

↳ constructor

②

Ctor hiding

+

class Connection {

private C P () { }

L 3

② Static initialiser

Connection Pool {
private static CP instance
→ private CP() { }
→ static getInstance() {
return new CP();
}}

Connection Pool {

private static CP instance

= new CPC();

get Instance(); {

return instance;

};

① Other hidden

①

get is static (not static)

②

static variable

A {

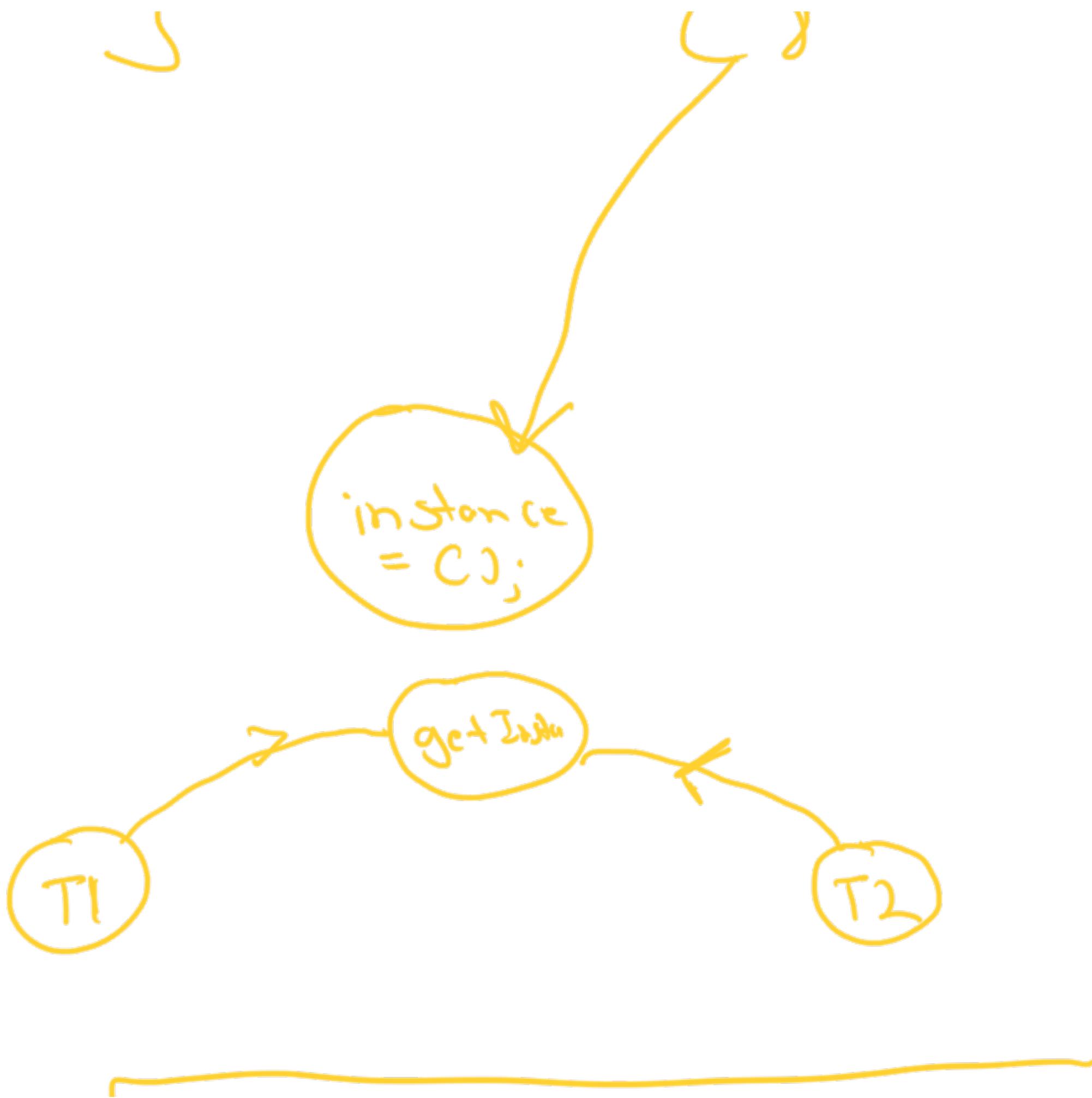
static b()

}

new A().b

A. 5 C

```
get Instance () {  
    if (instance == null) {  
        T2  
        => sync () {  
            if (instance == null)  
                instance = new CP()  
        }  
    }  
}
```



Public check locking

Singletion

Singleton

- ⇒ Share of resources → DB, log file
⇒ Single access point → config

How?

- ① CTOR hiding

- ② get Instance (Static)
- ③ static member (in static)

✓1 → Eager initialisation

✓2 → Lazy | on - demand

instance = null

...
if (i == null)

get Instance

↳ Create

VB → Thread Safety
→ Syncronizing

VB → Performance
→ Double check locking

if C)
else sync
if C i = n ()

* Enums

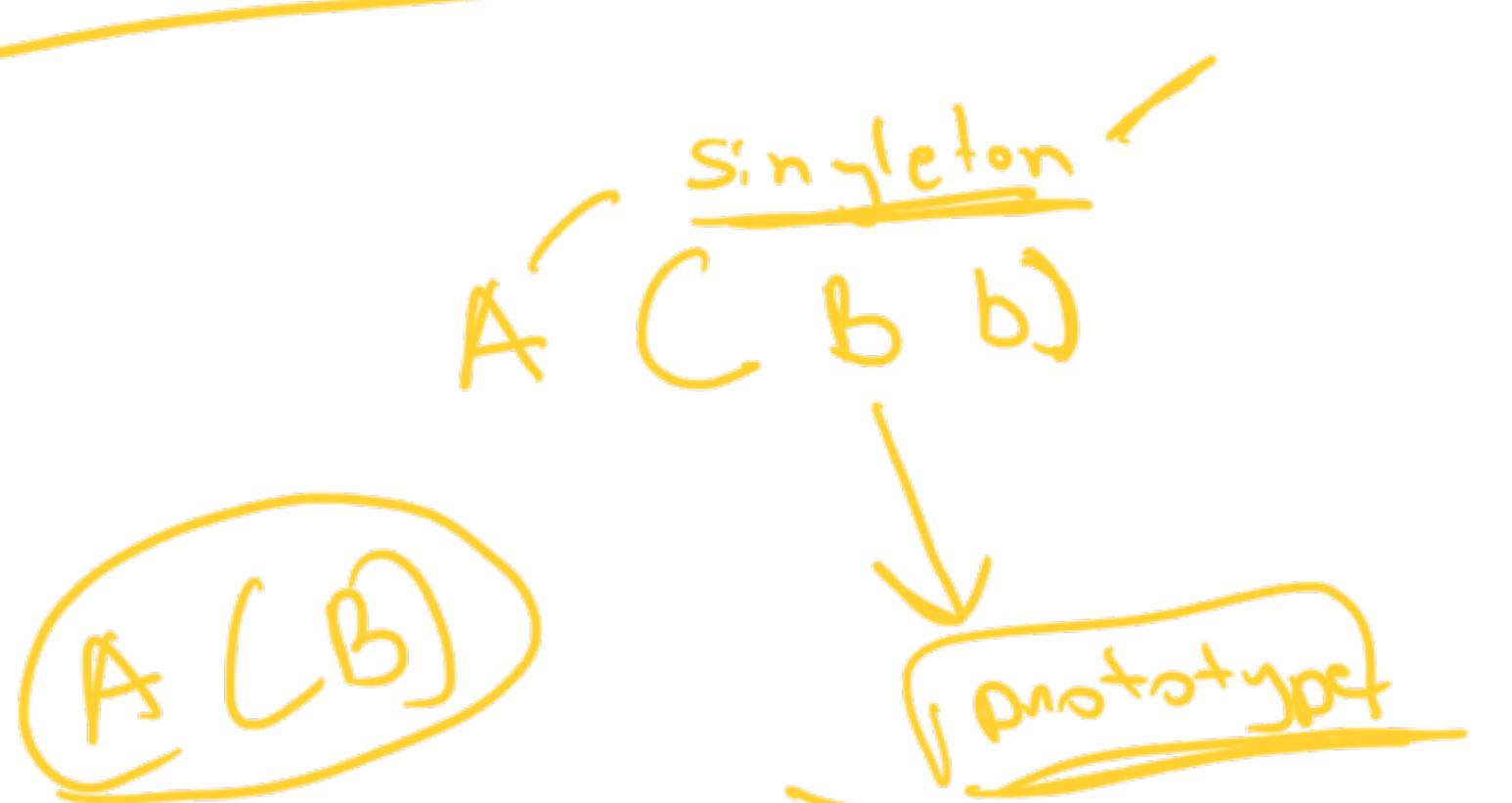
36

* Inner classes

- Yoda's CP }
- C3PO -

5:57 - 6:02

- 10:32





Factory - Simple

- Factory metho.
- Abstract Factor.

Builder

Design Pattern

Lombok, Builder

Create and assign values

User

→ name

→ email

↳

①



User (name, email)

②

Default contr + fields

User = new User()

User.name = " "

User S

name

email

phone

addr

⋮
⋮
20

Job

new Constructor, "email", "phon"

adms, ...

↓

* Readable ✅

* prone to errors

email phone
(String String)

(phone, email)

User (phone, email)

```
new User();
```

```
User.name = ""  
User.email = "j
```



fail object creation if name is null

- ① Unmanageable when
- ② No way to restrict creation for validation

↳ Job Q
PSP; /
Job Exp; /
Batches; /

↳

↳ User Frame



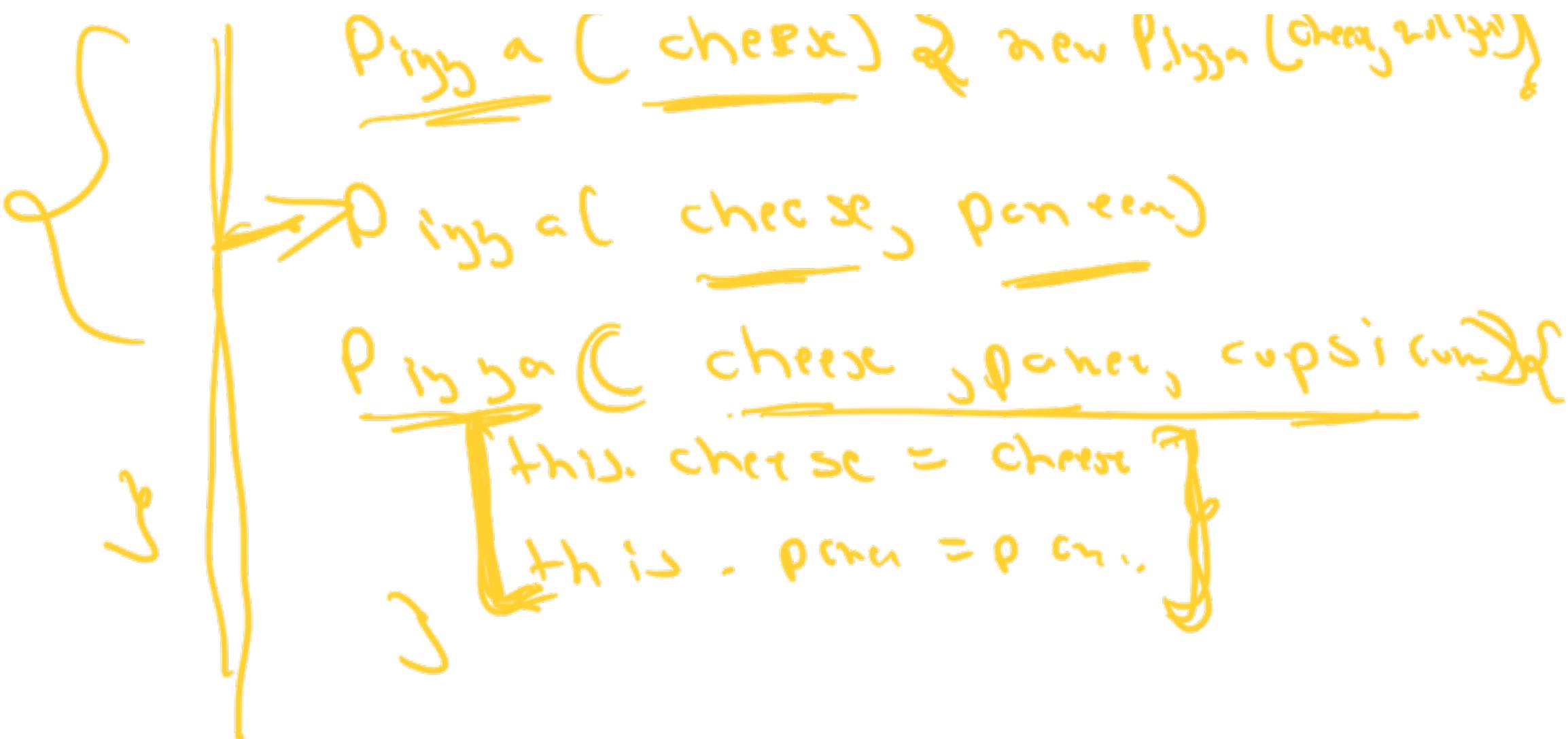
Ctor overloading

Pings Q



(10)

3



Setting Charge_{solar}

if $c_{ji} > 1^{\circ}$

$solar_j > 10^0$



Immutability

Objects cannot be modified

`Dat object = new C()`

data base & come = X

- set $\text{num}()$

get sum of APL



A. $a = 2$

- $a = 1$

$A \alpha = \text{new } A()$ ✓ - 22

$A \alpha^* = \text{new } A()$

$\alpha \sim \alpha^*$

① ~~default~~
ctor + sethui X

CD

② ctor()

- no sethui

- ~~private~~

✓

- ① Large set of arguments
Option a /
→ Ctor.
- ② Fail object creation
- ③ Option c / construction
- Telescoping ctor.
- ④ Immutable

User {

UserCrop<String>)

for value in map

User . name = map.get("name")

}

Builden

① Immutable

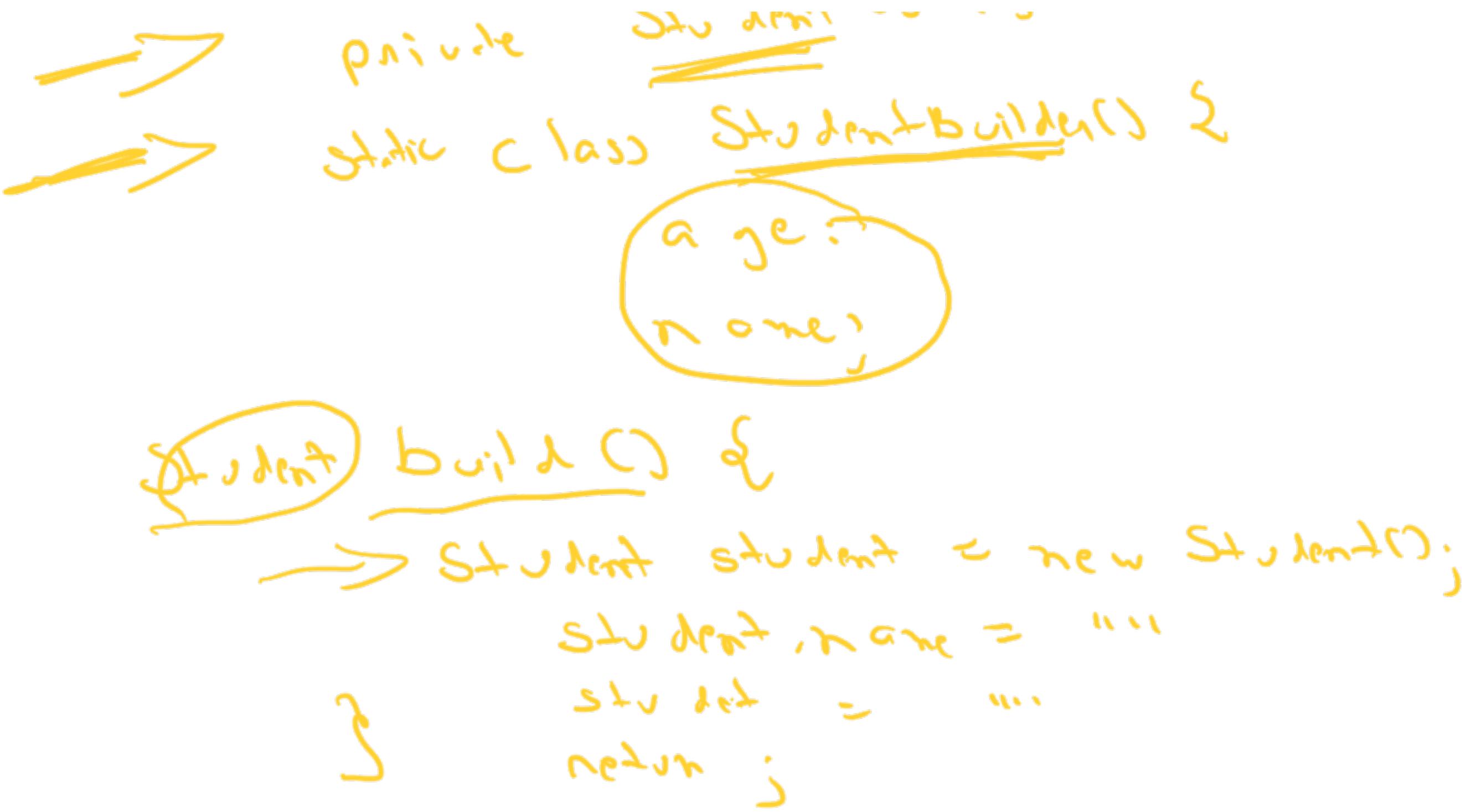
return ()

② Hook \Rightarrow Validation \hookrightarrow
 \Rightarrow build()

Builder - builder class

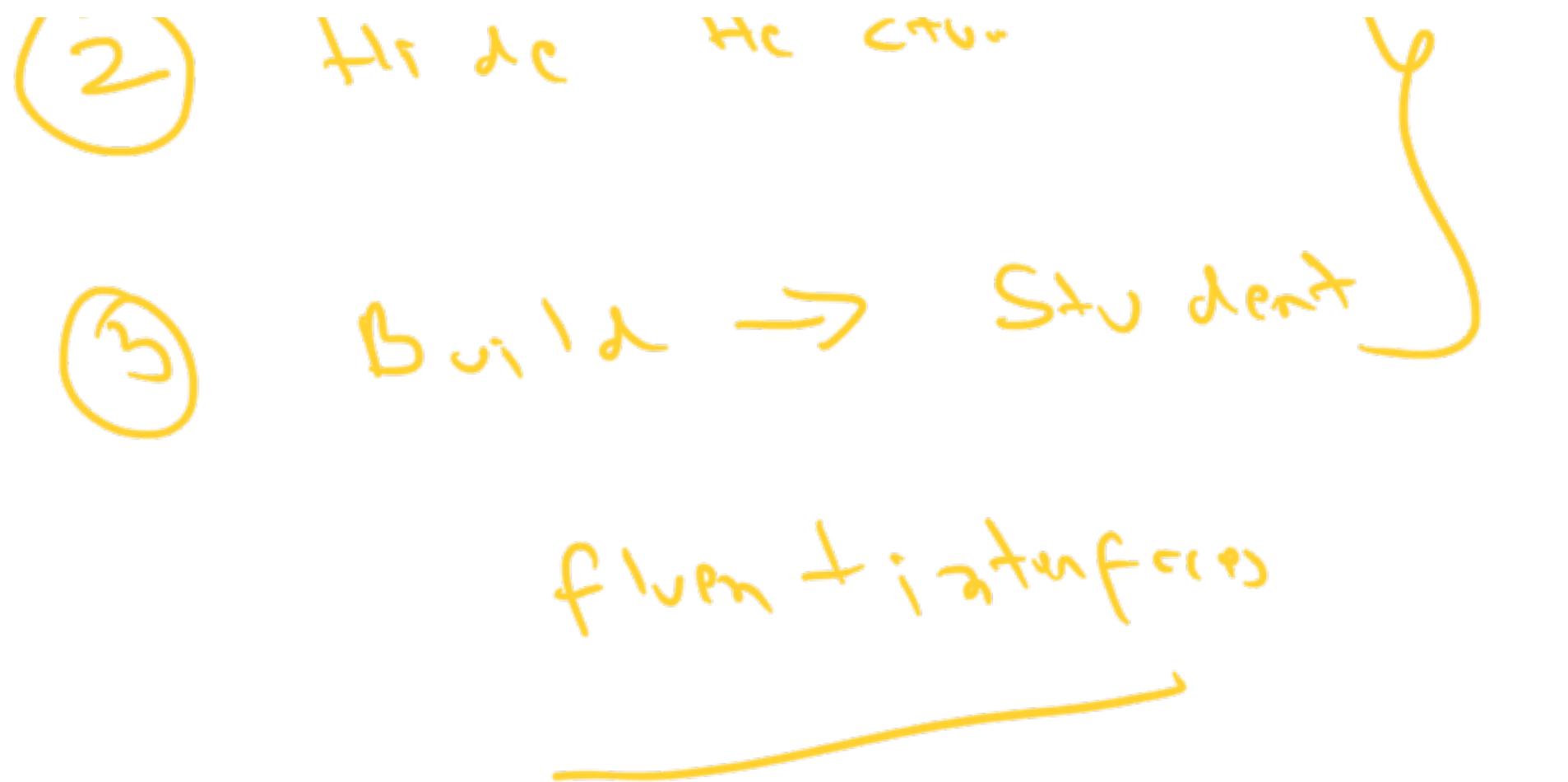
- Student Document:
- inner static

class Student {
 name
 age
 constructor () {}}



① Static inner

② ...



- (1) Immutable
- (2) Compiled object
- long set of fun.
- (3) Validate -

DTO

- OIC
- ^{not} easy to maintain