L : Liskov's Substitution Principle

I : Interface Segregation Principle

D : Dependency Inversion Principle

Problem Statement

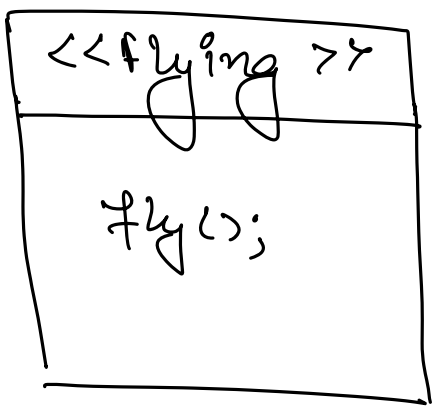
Some birds demonstrate a behaviour, while others don't demonstrate a behaviour.

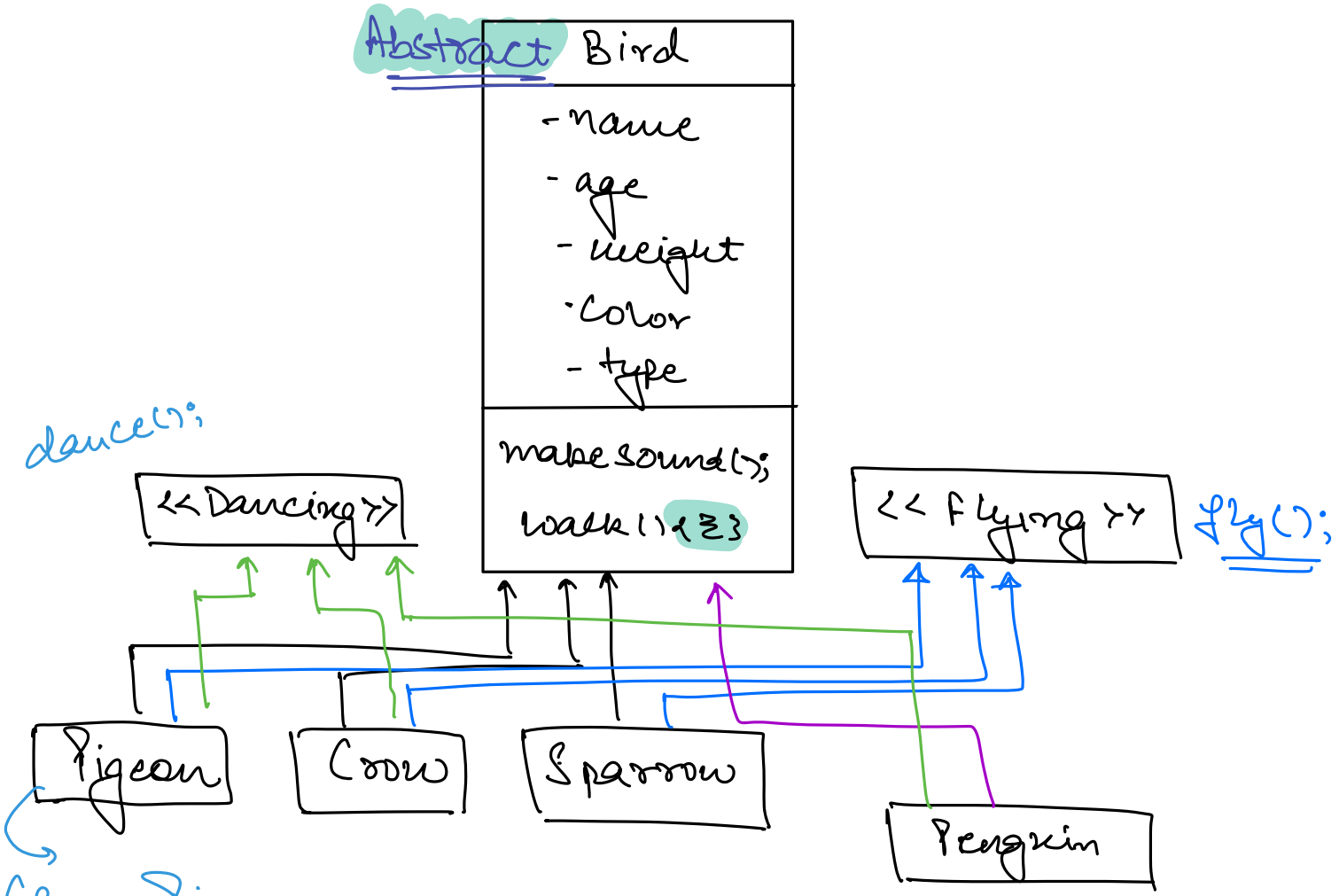
1) Only the birds having the behaviour should have that method.

2) We should be able to create a list of birds with that behaviour.

Interface => Blueprint of Behaviours.

Interface =>





Class Pigeon extends Bird implements Flying, Dance

```
Bird b = new Pigeon();
b.fly(); ✓
b = new Penguin();
b.fly() X
```

⇒ 10 types of Birds. ⇒ 10 interfaces

No class | interface Explosion.

Liskov's Substitution Principle.

Object of any child class should be as it is substitutable in the parent class reference without requiring any extra code changes.

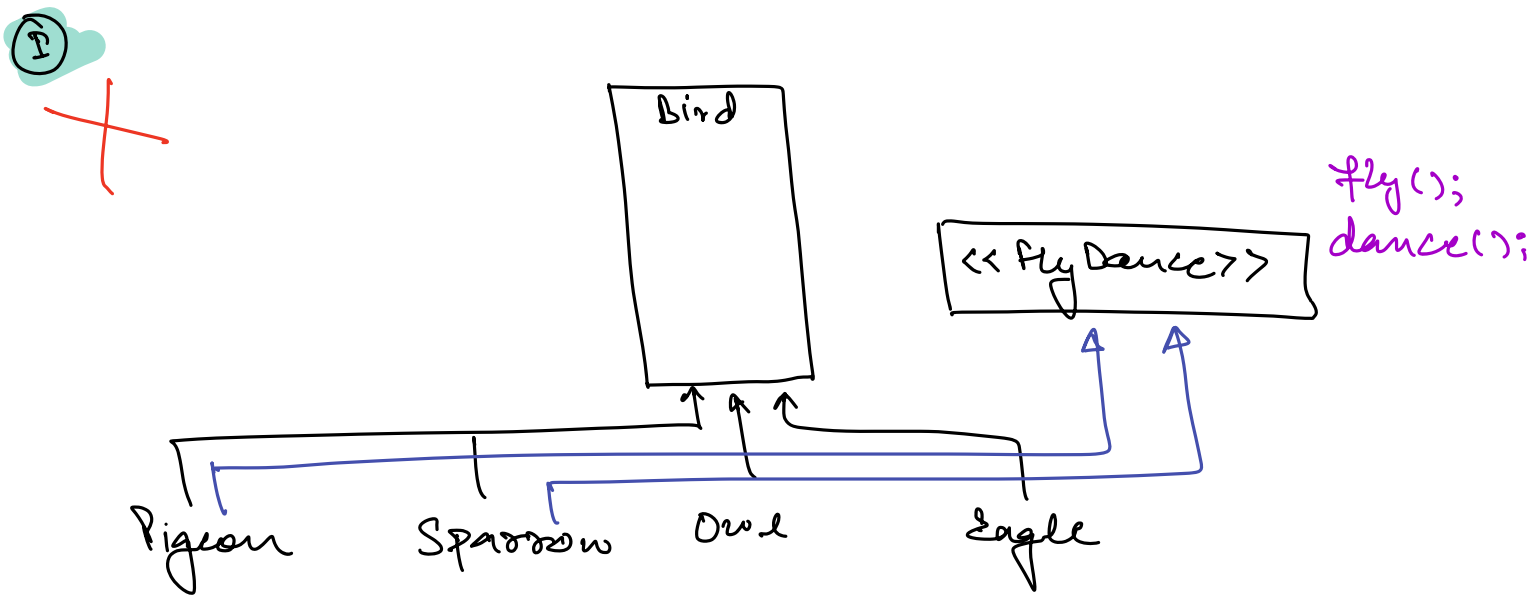
Interface Segregation Principle.

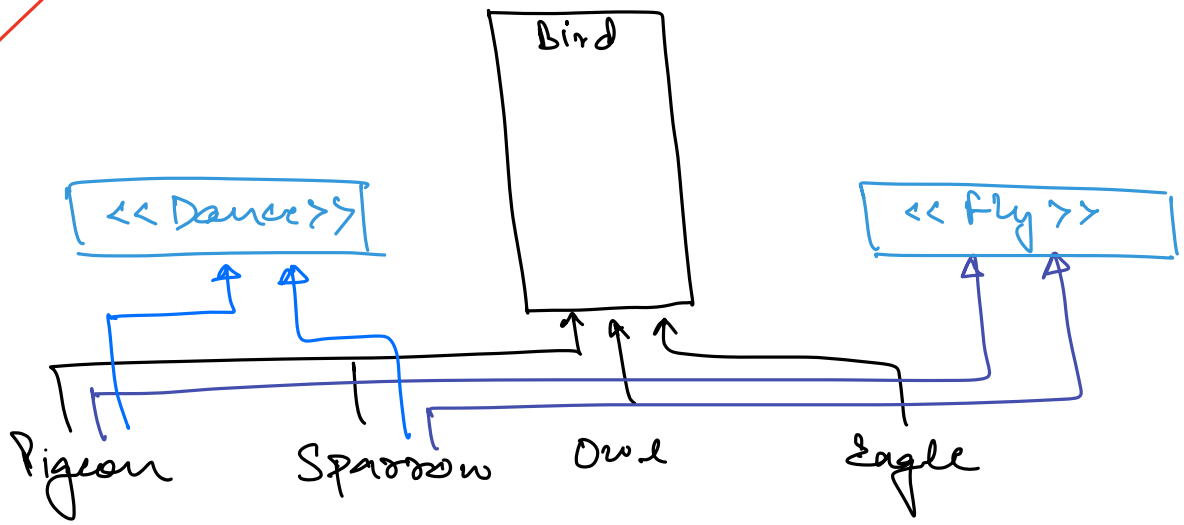
Requirements:

Some birds can fly

Some birds can dance

All the birds who can fly can dance as well & vice-versa.





⇒ Interface Segregation Principle.

→ Interfaces should be as light as possible.

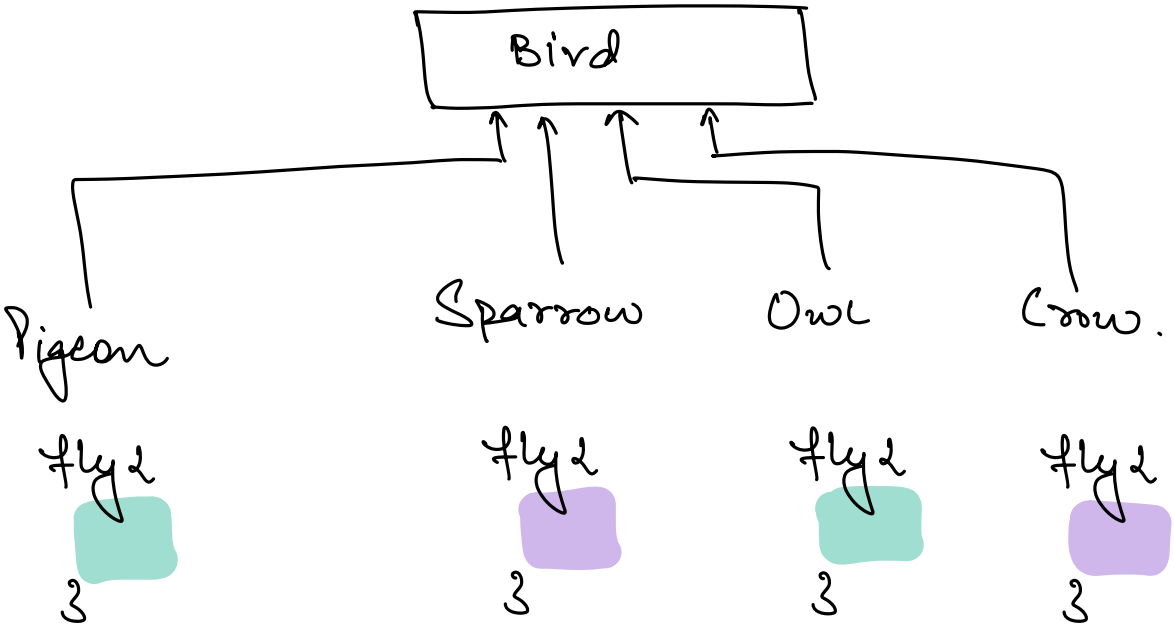
→ Ideally every interface should have a single behaviour.

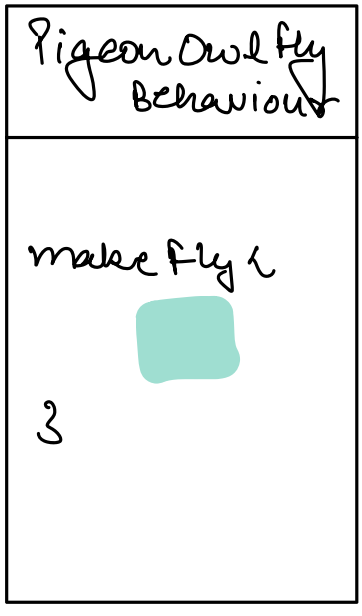
→ SRP on Interfaces.

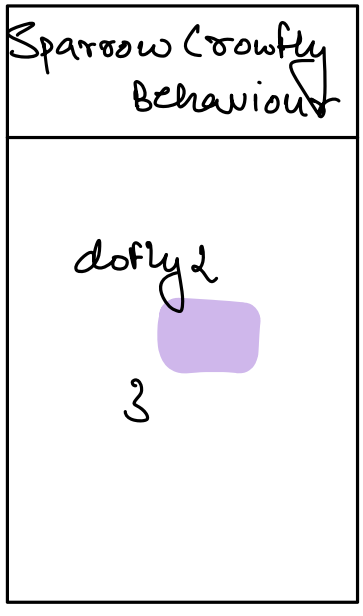
Functional Interface ⇒ Interface with single method.

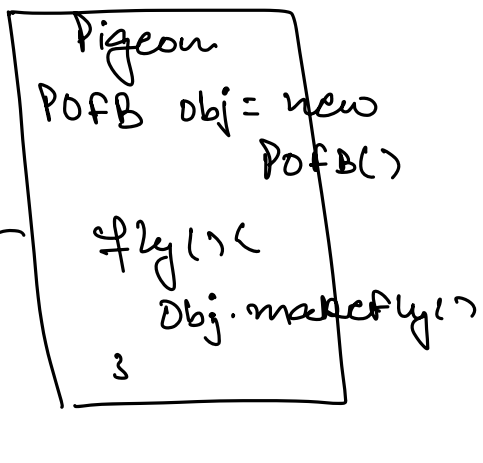
Dependency Inversion Principle

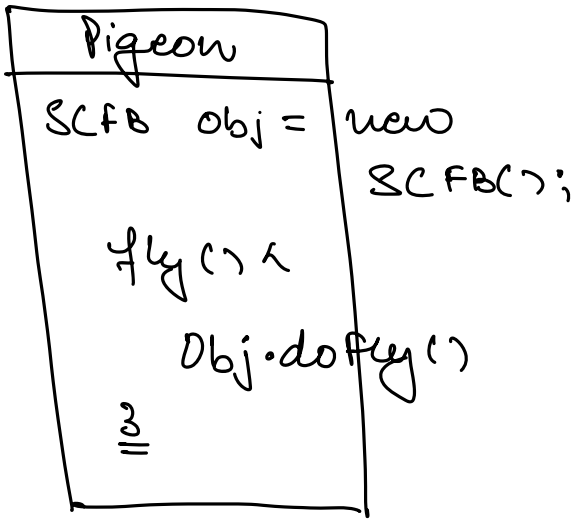
DRY.



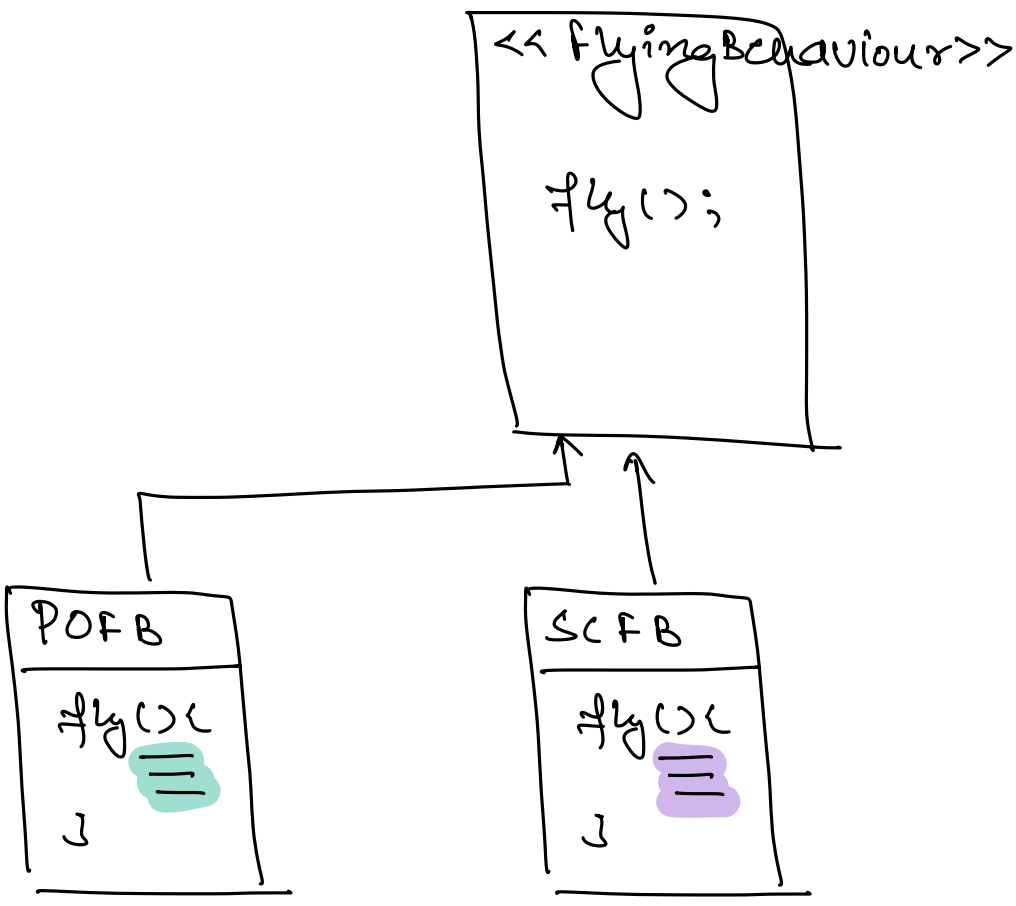


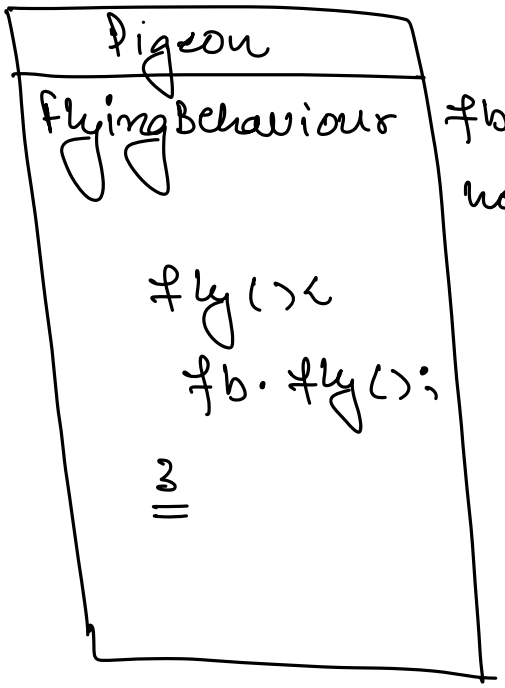




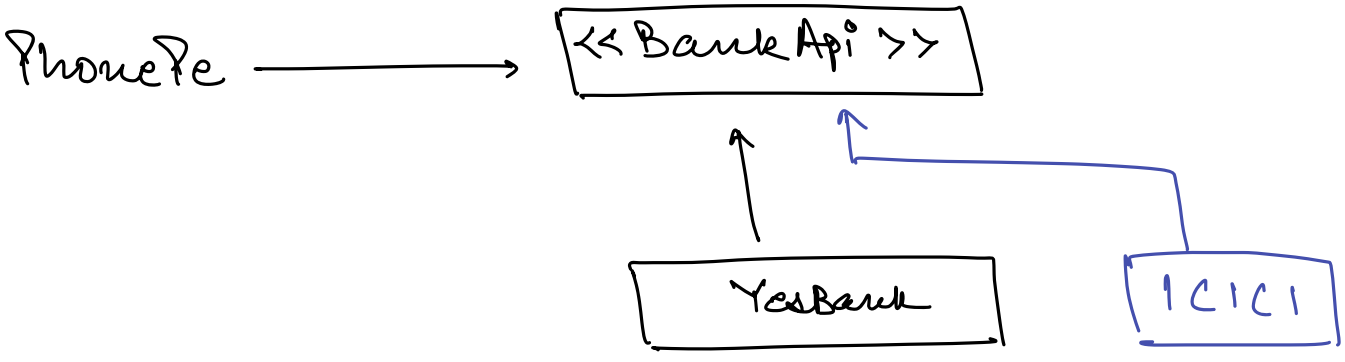


Program to Interface, Not to implementation.





PhonePe ⟶ YesBank } Tight Coupling



BankApi api = new ~~YB()~~;
ICICI

Dependency Inversion Principle

No 2 concrete classes should depend on each other directly, they should depend on each other via an interface.

Dependency Injection (Not a Part of SOLID).

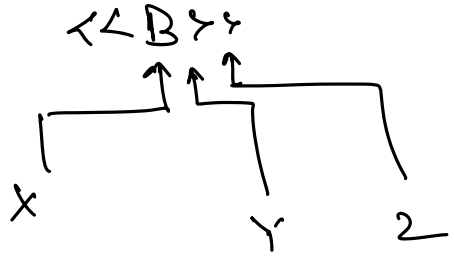
```
A {
    B b = new B();
    ==========
}
```

⇒ Class A is dependent on class B.

⇒ Dependency Injection

Instead of creating the dependency object on our own, it should be given at the time of object creation.

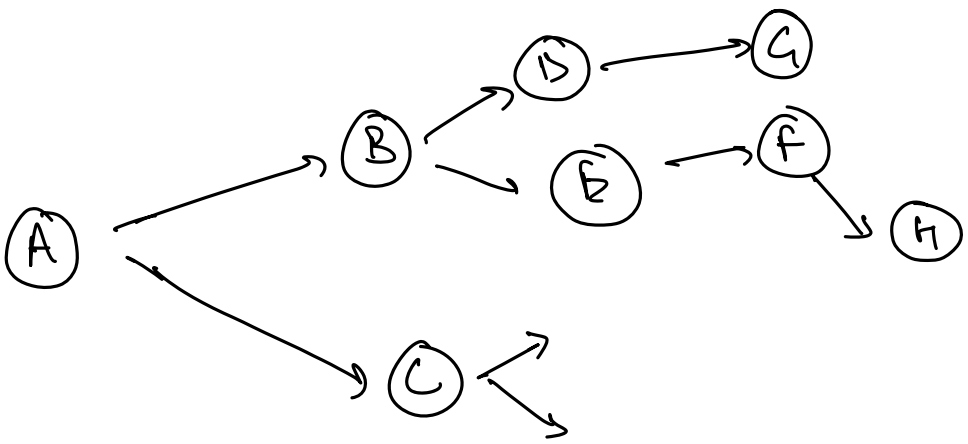
```
A {
    B b;
    A(B b) {
        this.b = b;
    }
}
```



Client

```
B b = new Z();
A a = new A(b);
```

Instead of creating the dependencies on our own, dependencies are being injected.



=> SpringBoot
