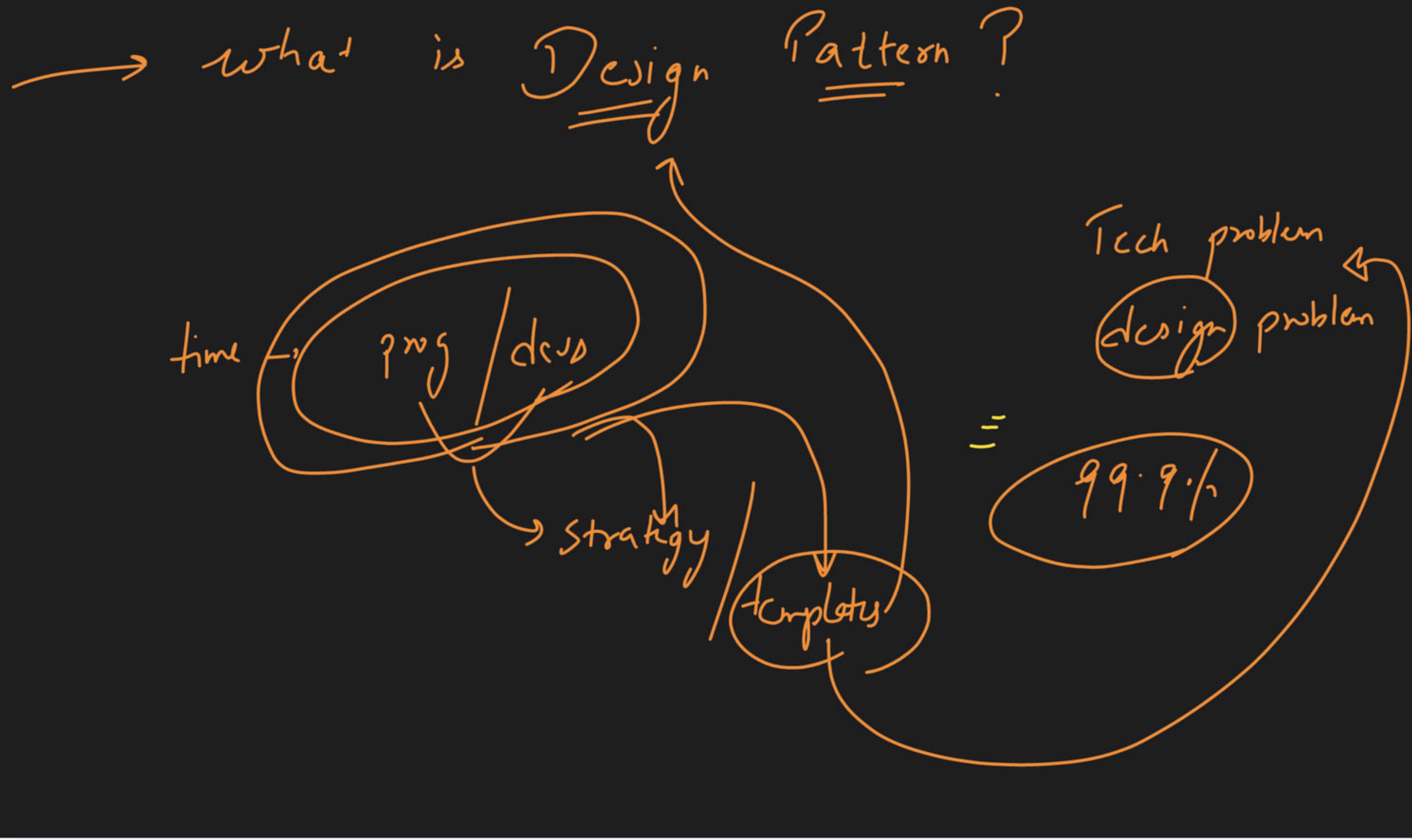
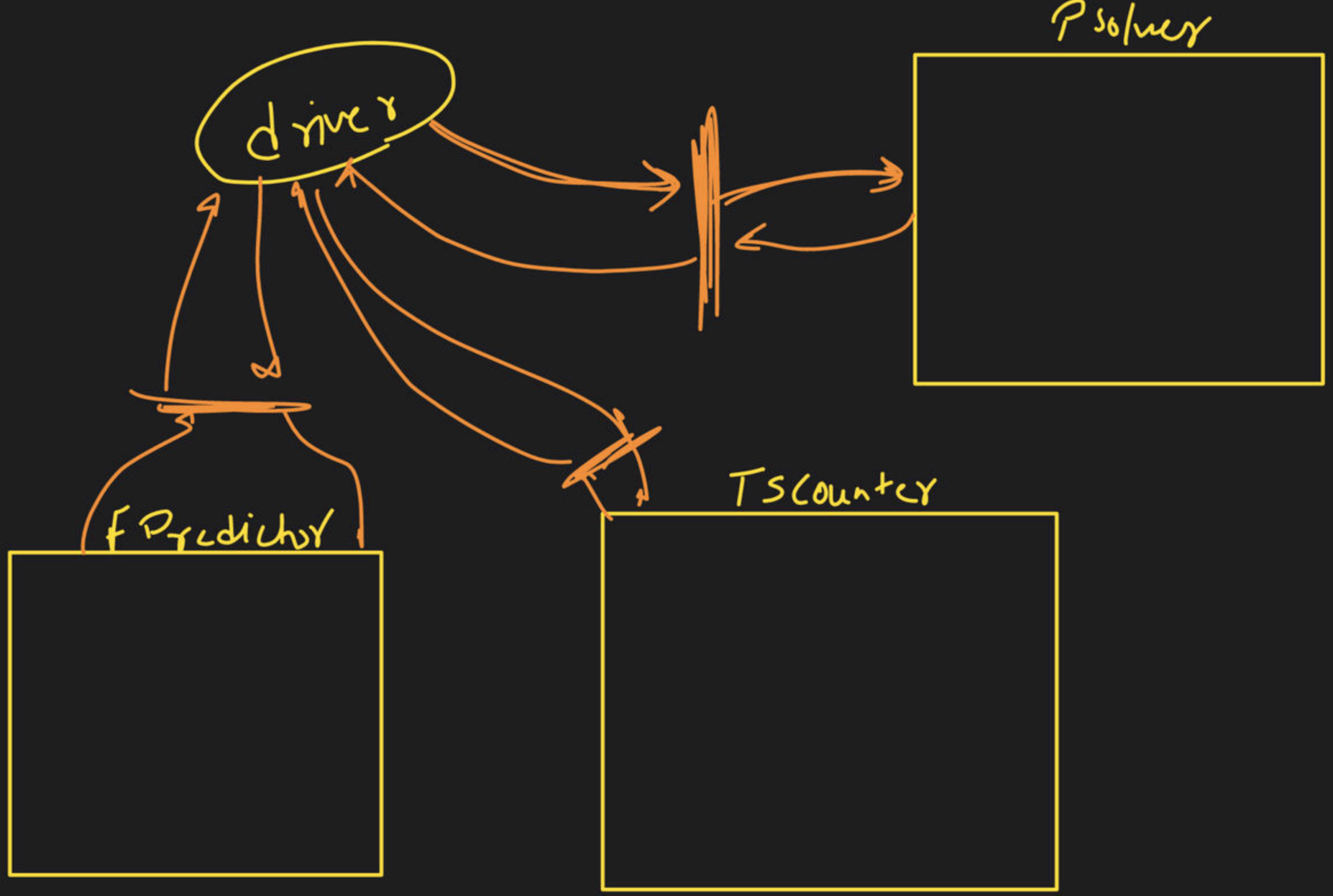
Introduction to Design Patterns, Factory & Abstract Factory Pattern

Special class

Love Babbar • Mar 19, 2024


what is Design Pattern ?
time
prog / dev
Strategy
Template
Tech problem
design problem
=
99.9%


driver
P solver
FPredictor
TSCounter

Babbar → CodeHelp → Article →

template

title -> < 150

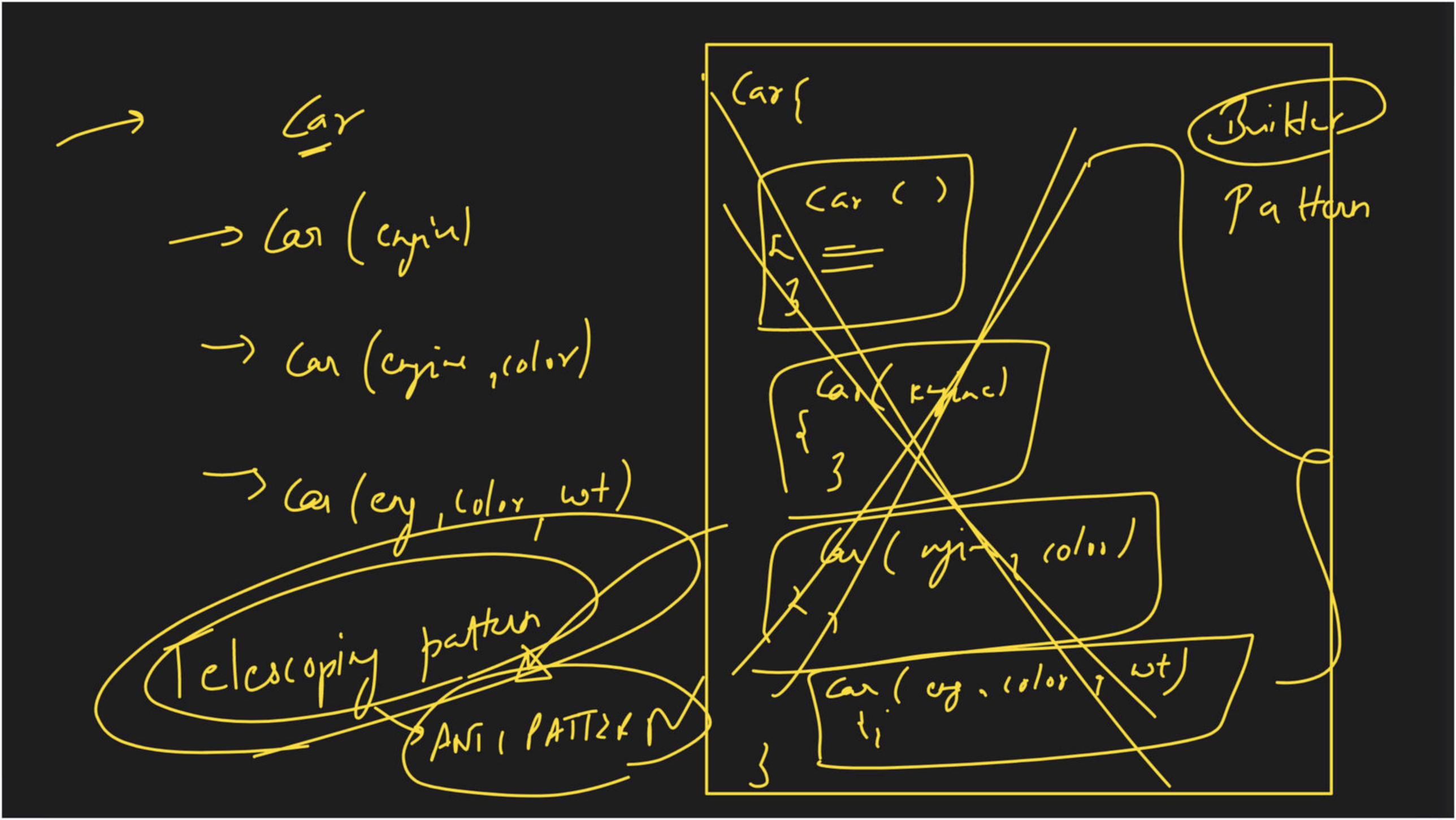
desc -> < 3000

tx -> 3

ck -> 3

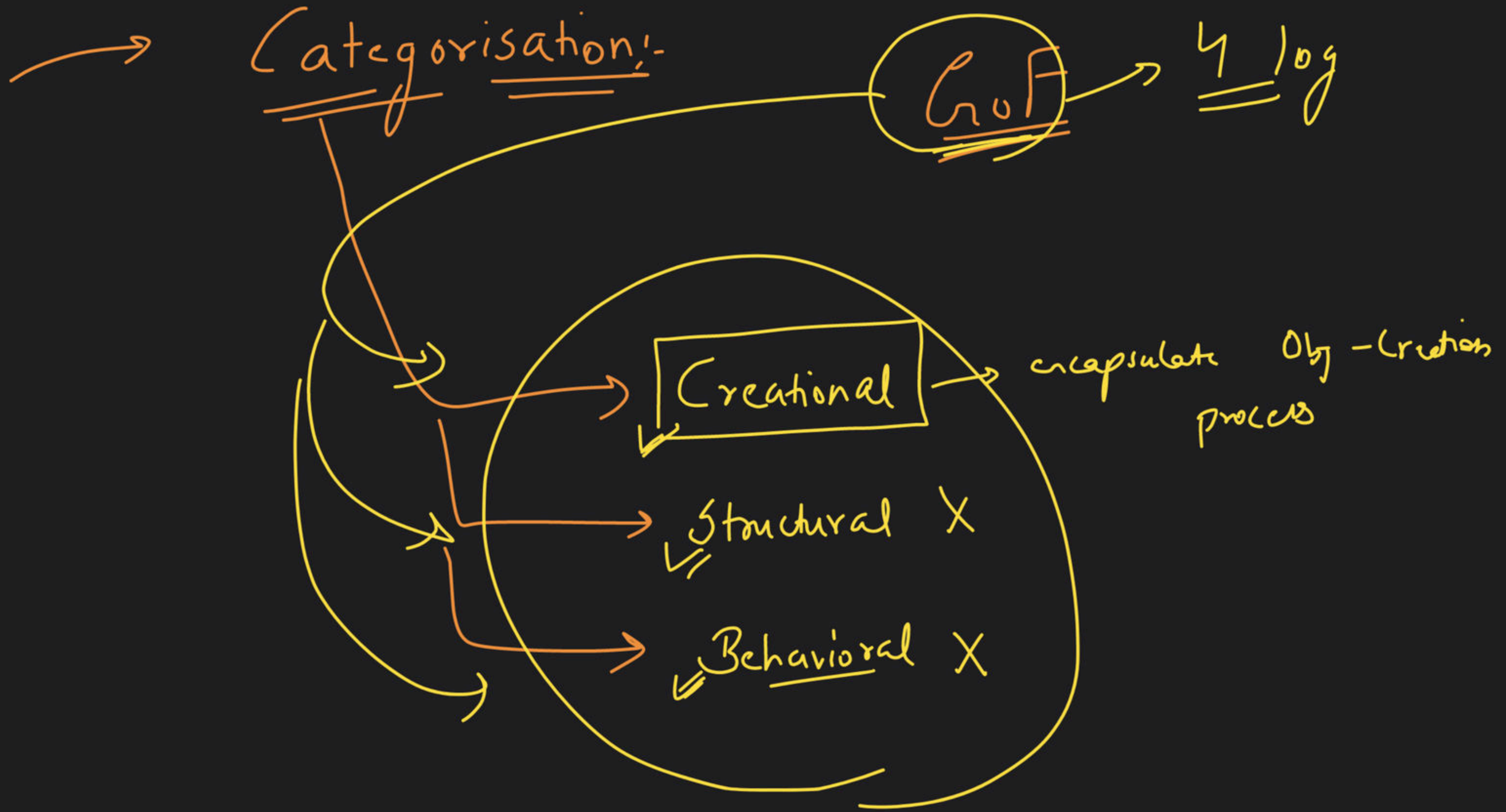
10 → approved

algorithm


Car
Car()
Car(engine)
Car(engine, color)
Car(eng, color, wt)
Telescoping pattern
ANTI PATTERN
Car {
Car()
Car(engine)
Car(eng, color, wt)
Builder
Pattern

Why Design Pattern?

didn't want to reinvent the wheel


Categorisation:-
GoF
4 log
Creational
encapsulate Obj - Creation process
Structural X
Behavioral X

Factory method

Calendar.getInstance()

NumberFormat.getInstance()

Example → Abstract factory

→ DocumentBuilderFactory → newInstance()

→ TransformerFactory → newInstance()

① Factory Method Pattern:-

It provides an Interface for obj creation but delegating the actual instantiation of objects to subclasses

2 min Break

Ex

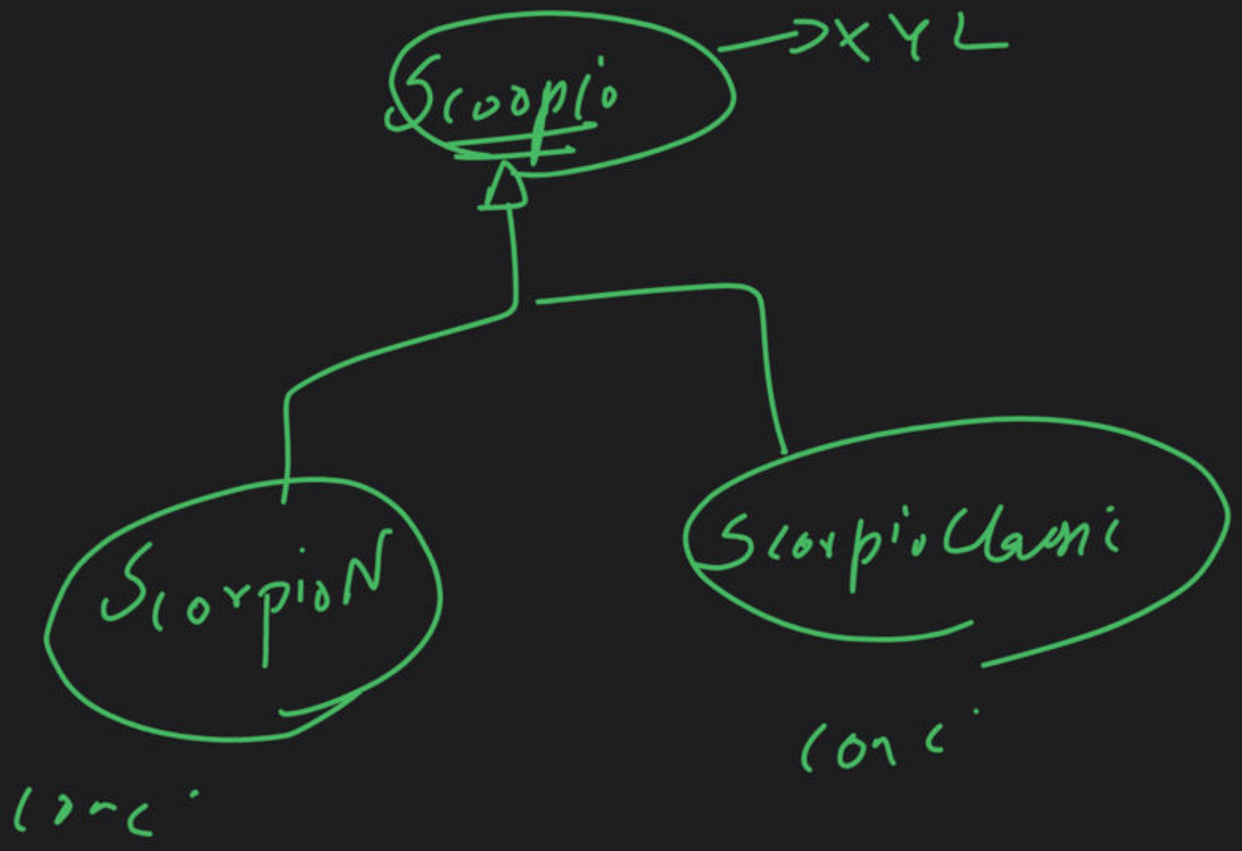
Collection c = new ArrayList<>();

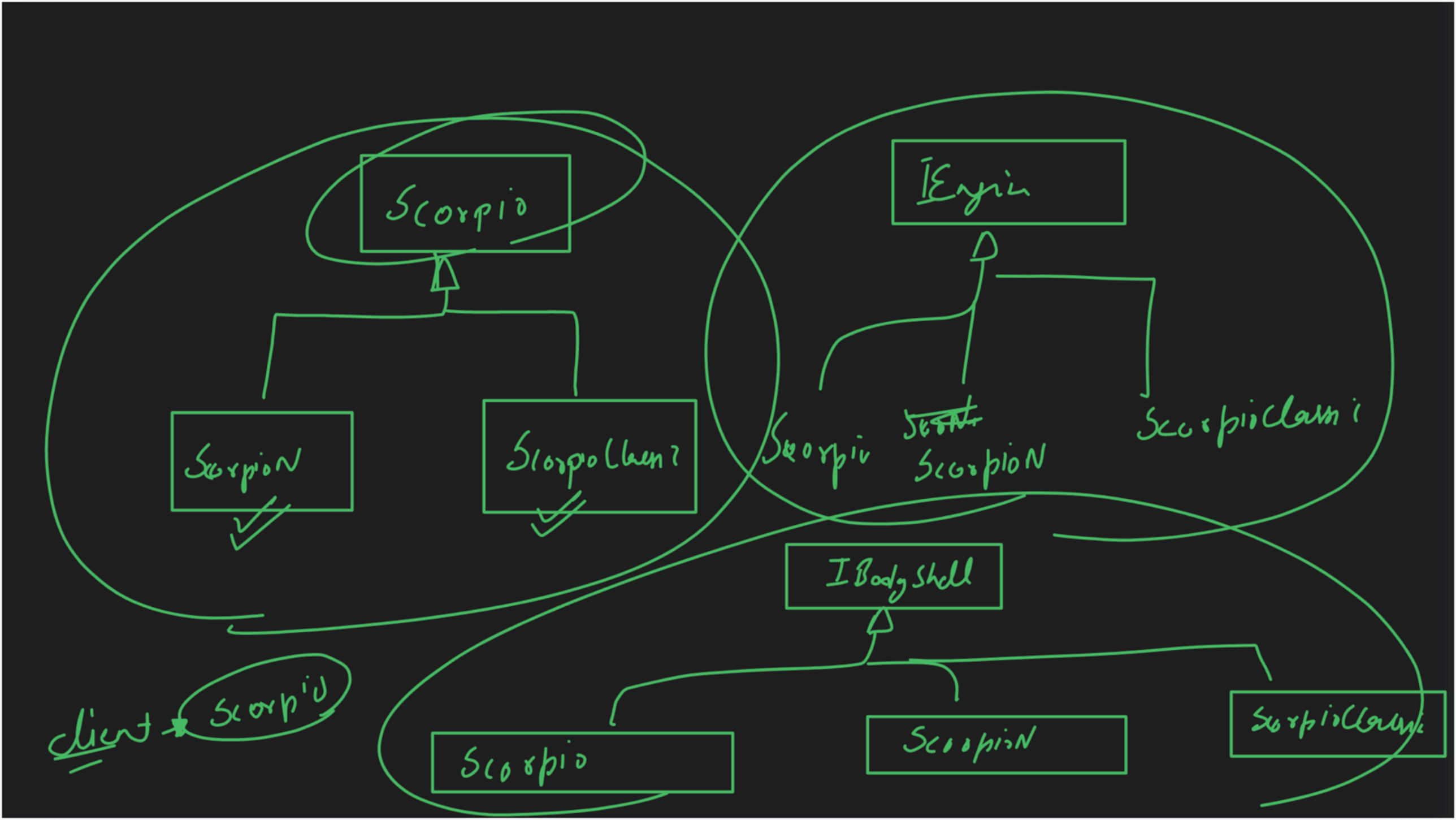
Tight coupling

A obj = new A();

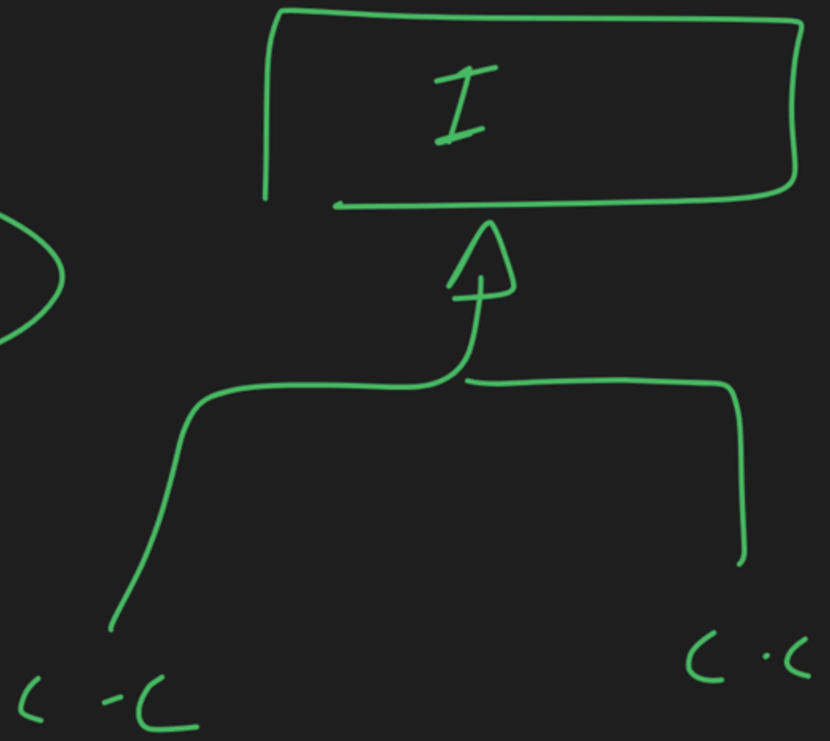
dependent

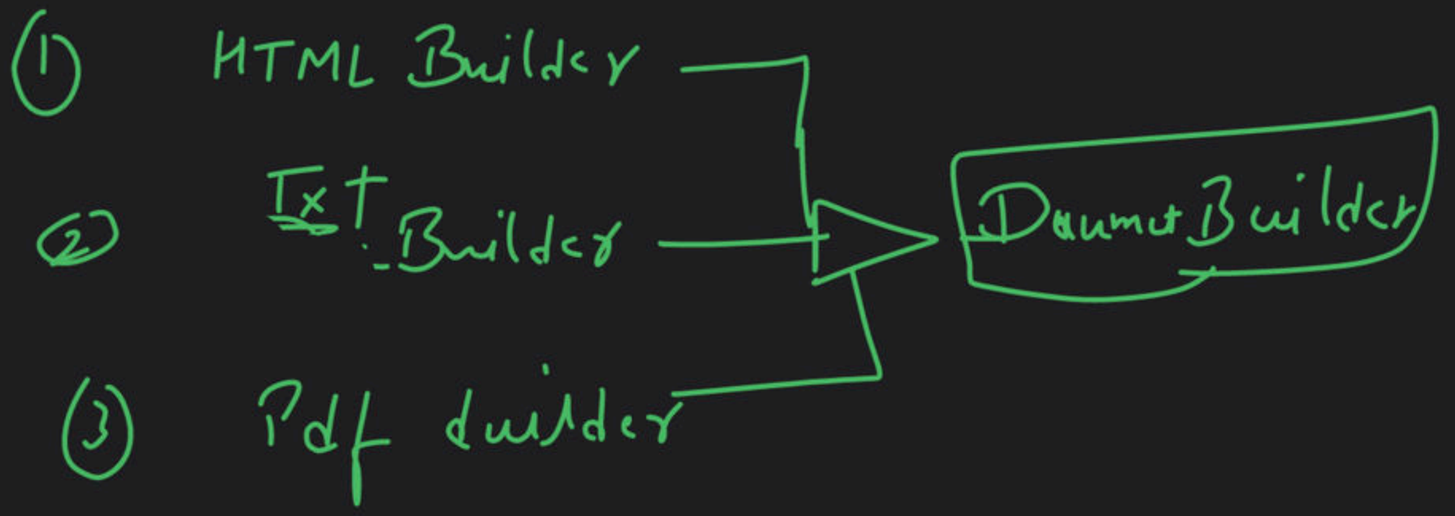
"always code to an Interface, not an Implementation"

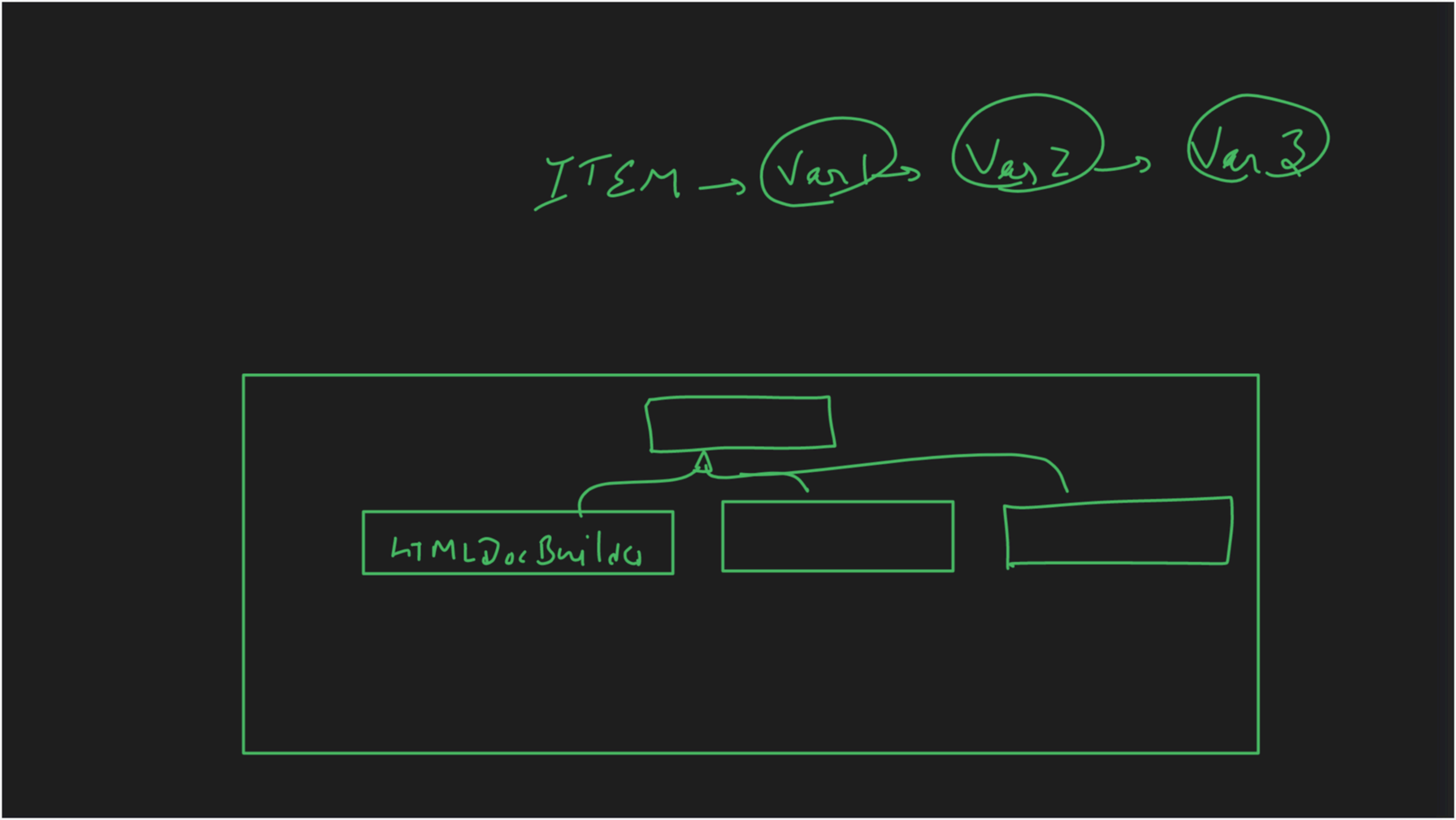

Scorpio
SUV
Scorpio N
Scorpio Classic


Scorpio
IEngine
ScorpioN
ScorpioClassic
Scorpio
ScorpioN
ScorpioClassic
IBodyShell
Scorpio
ScorpioN
ScorpioClassic
client
Scorpio

If obj = new C()



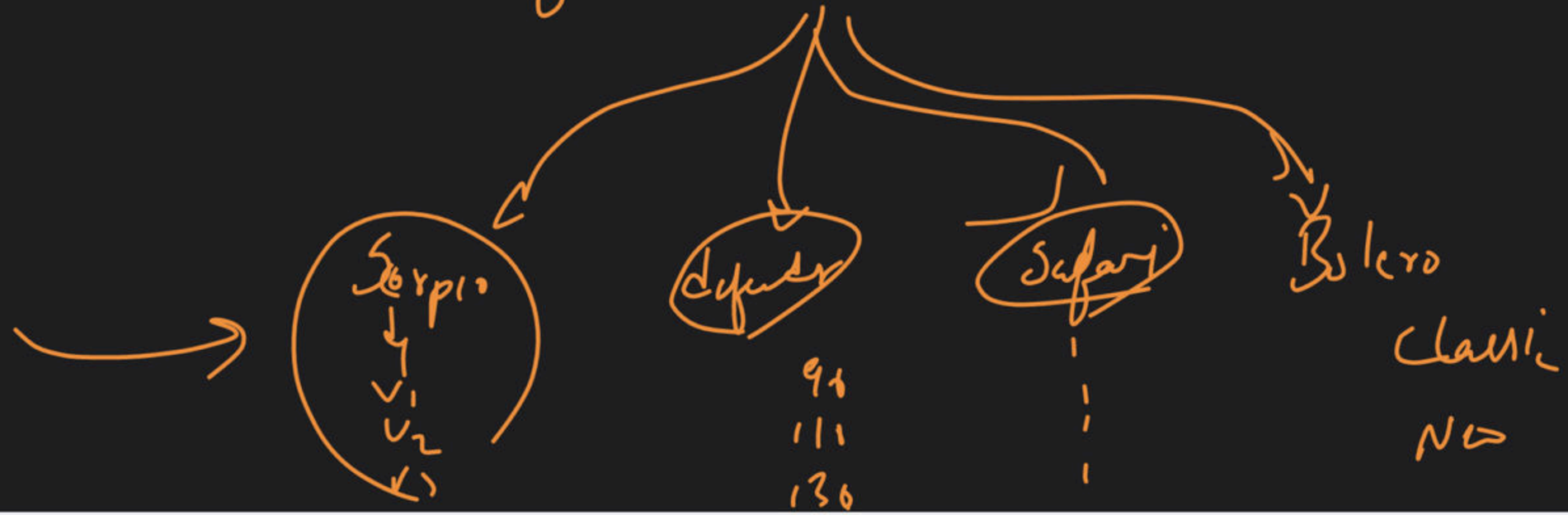

(1) HTML Builder
(2) Txt Builder
(3) Pdf builder
Document Builder


ITEM
Var1
Var2
Var3
HTMLDocBuilder

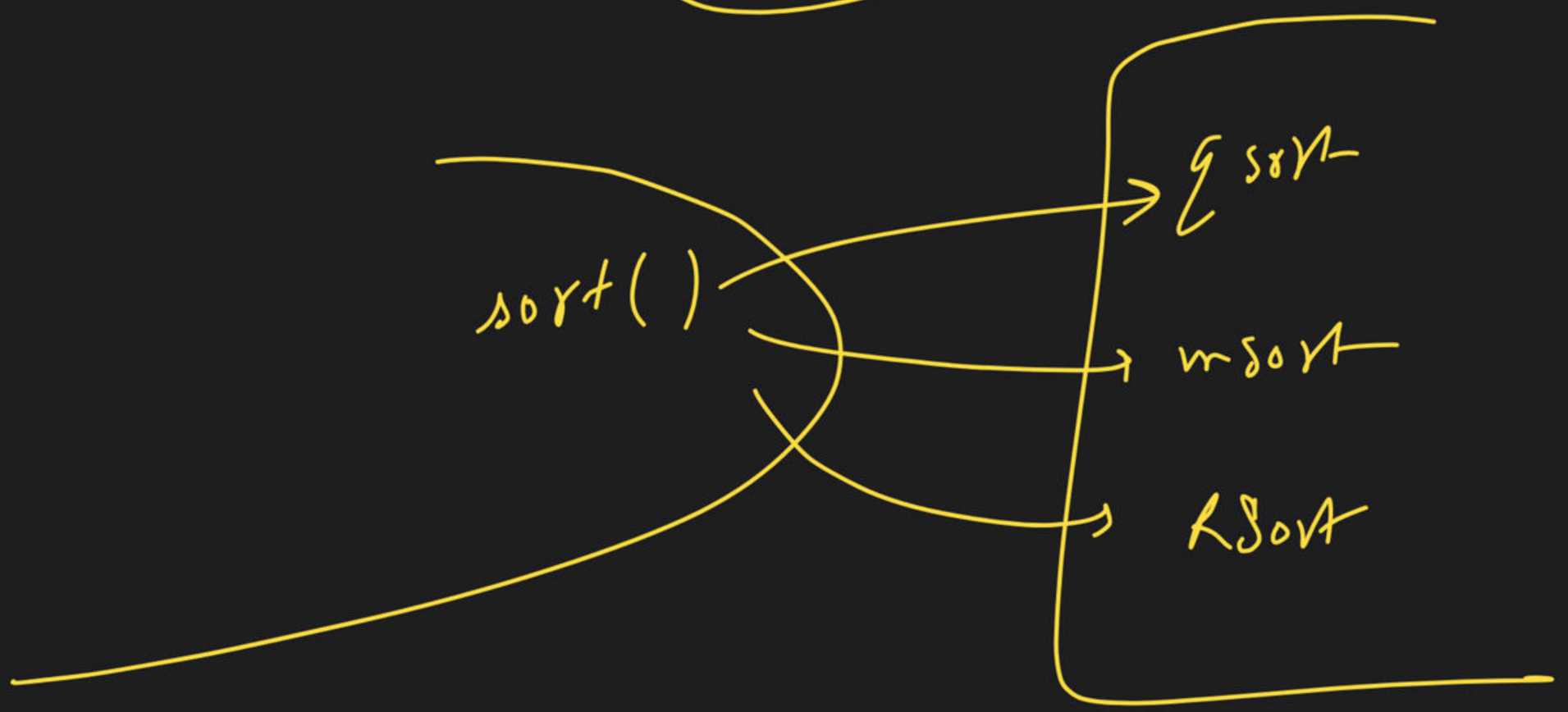
Abstract Factory

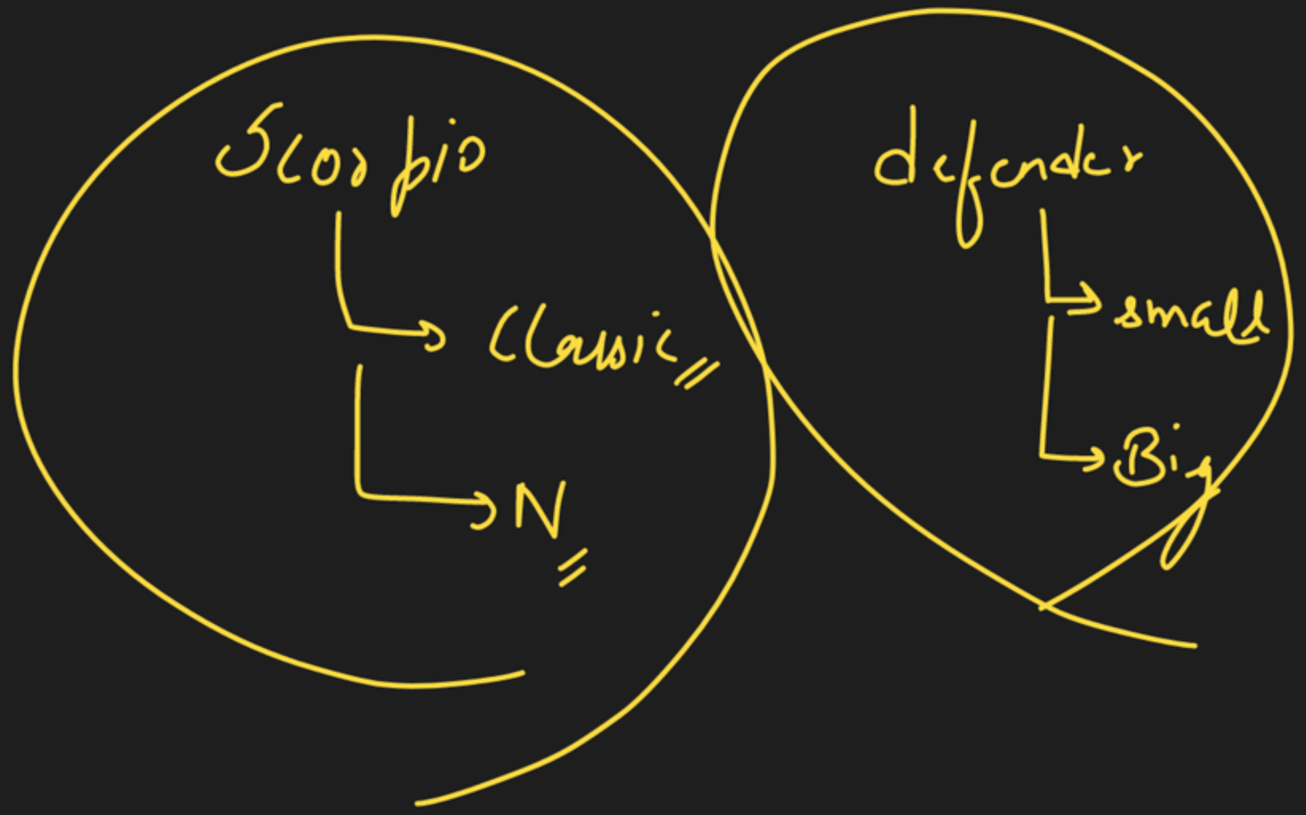
factory of factory

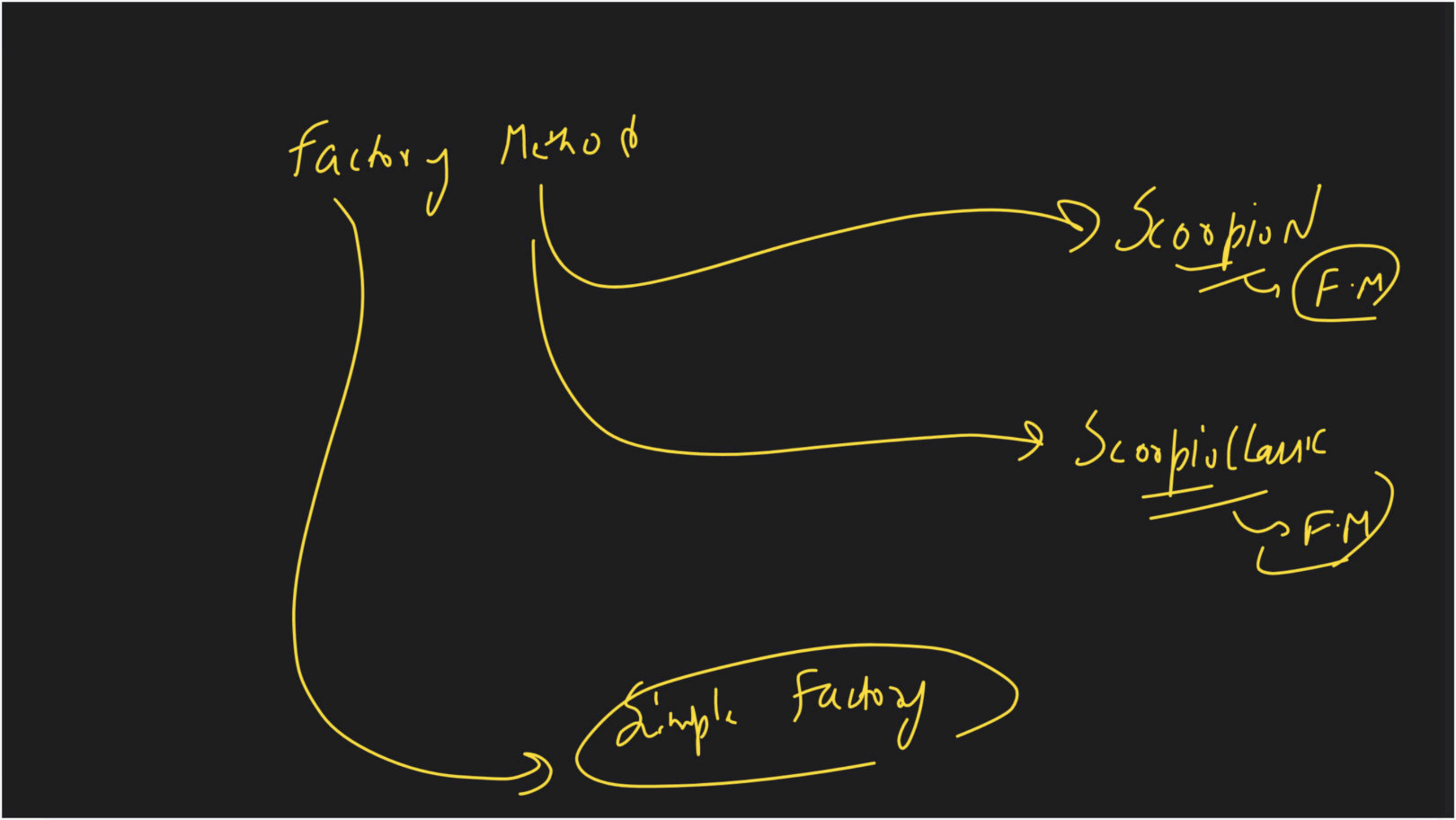
solves the problem of creating family of related products

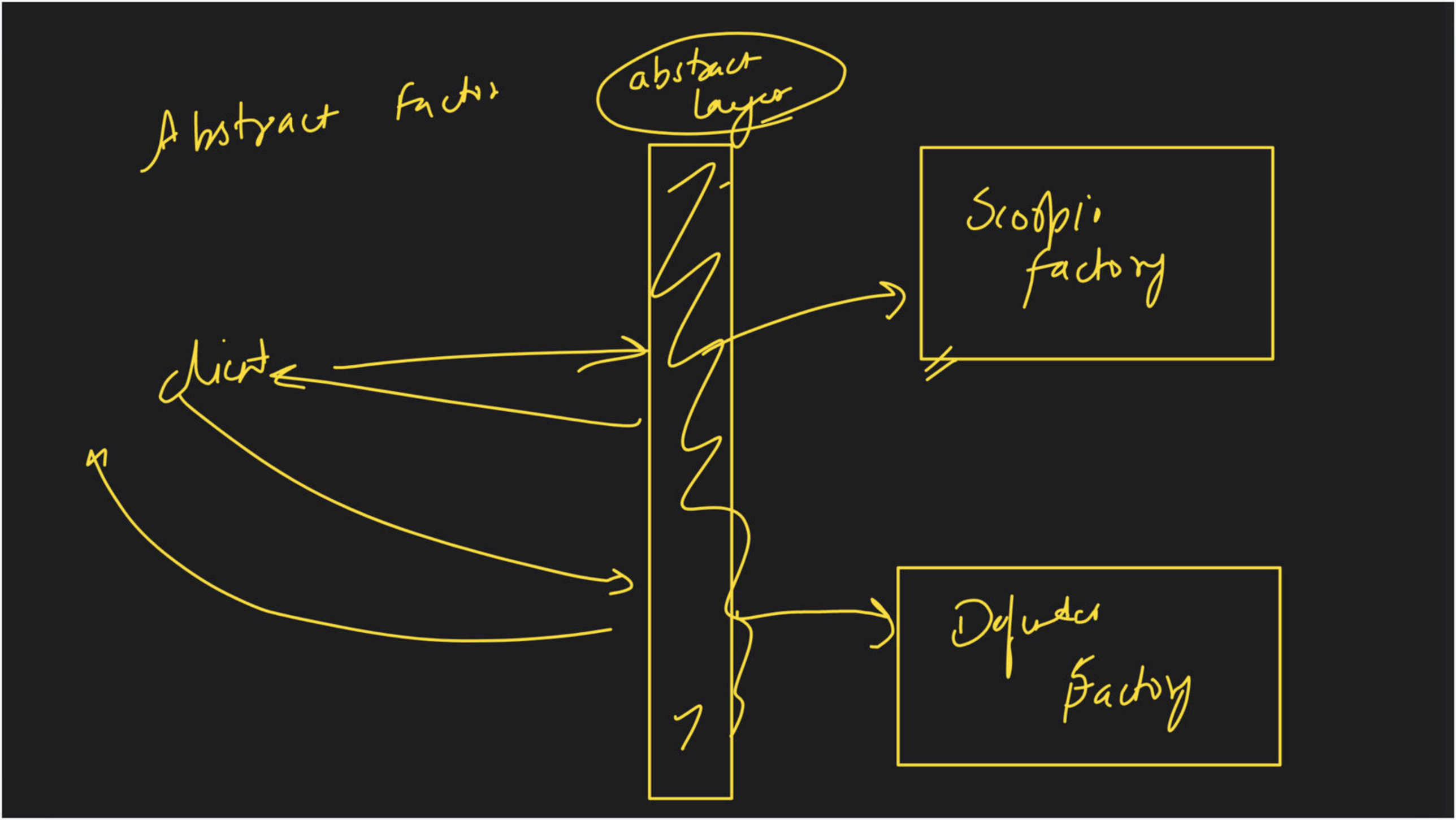


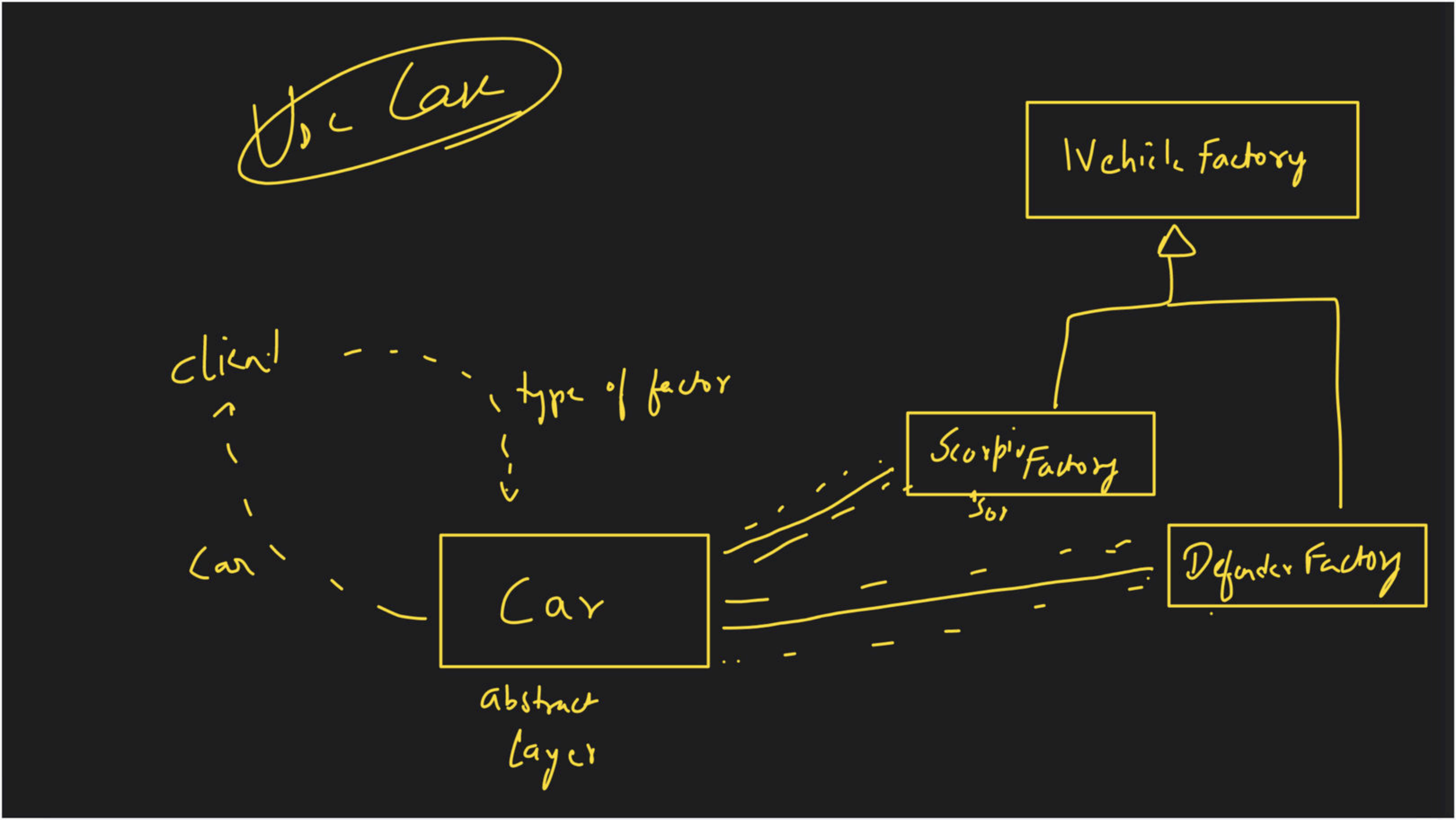
⟹ defining an interface to create families of dependent objects without concrete classes




Scorpio
Classic
N
defender
small
Big


Factory Method
Scorpion
F.M
Scorpion Classic
F.M
Simple Factory


Abstract factory
abstact layer
Client
Scoopi factory
Defender Factory


Use Case
IVehicle Factory
client
type of factor
Scorpio Factory
Car
Car
Defender Factory
abstract
Layer