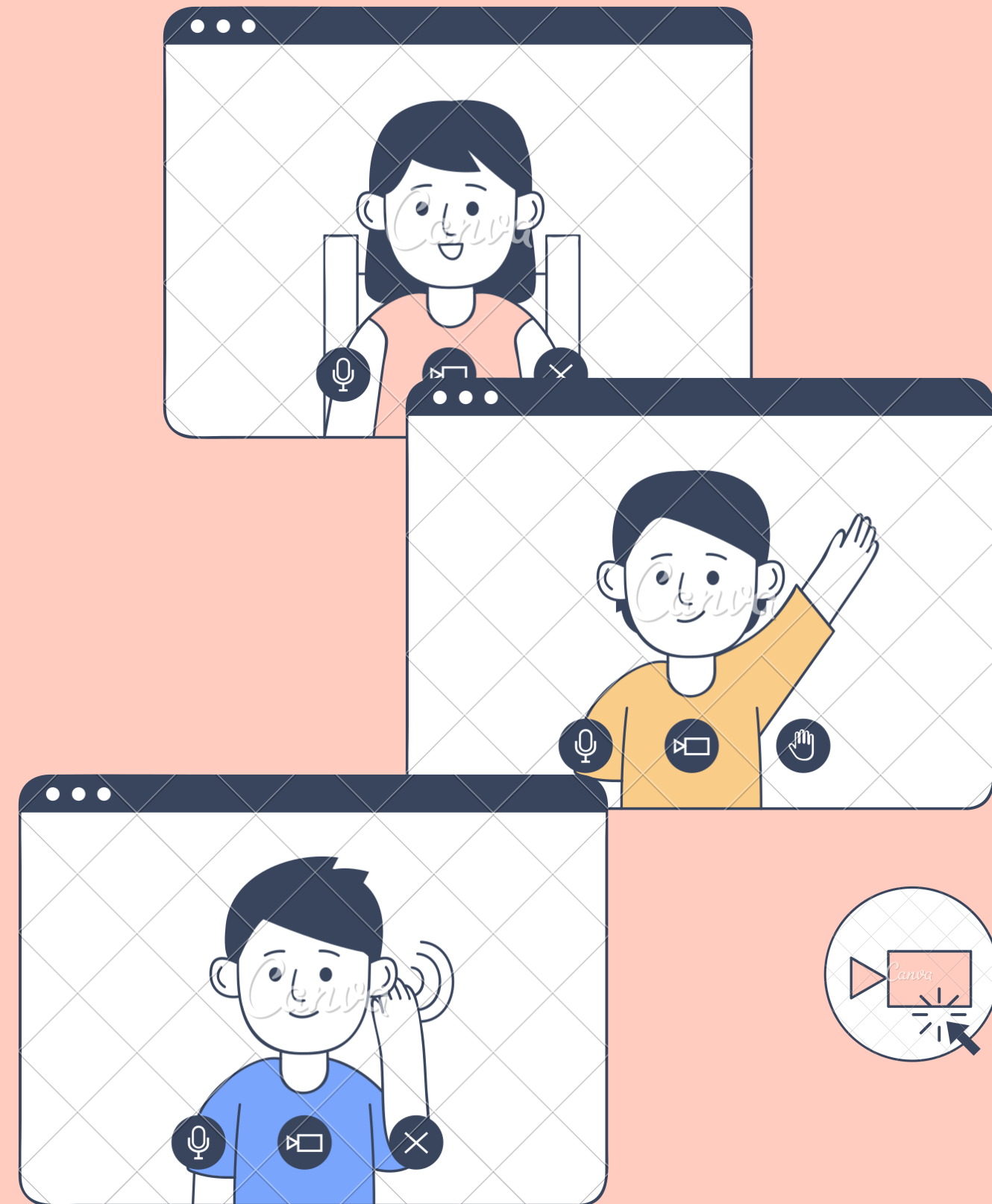


Factory Pattern

Group Member:
Đào Thế Hiền
Nguyễn Trức Linh



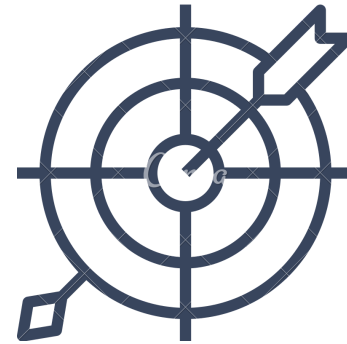
Content

1



Introduction

2



Usage

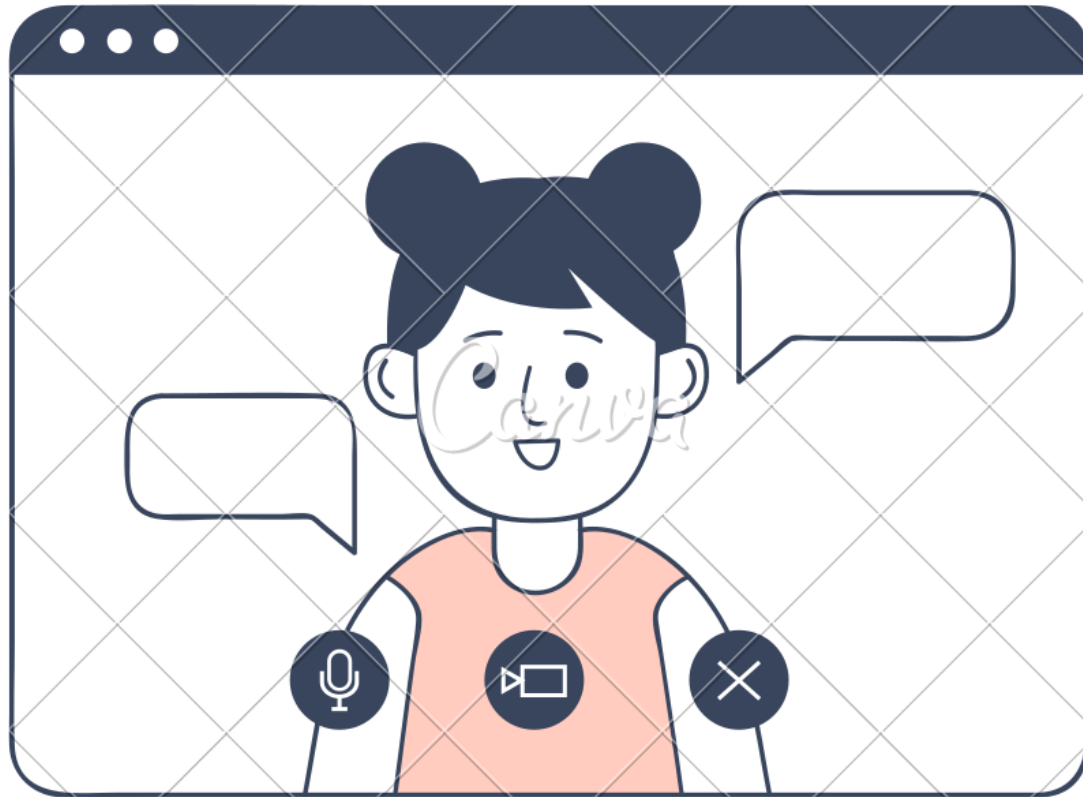
3



Benefits



Introduction



DEFINITION

It is an interface to create objects, but allows subclasses to decide which class to instantiate based on some business logic.





Createpizza()



FACTORY PATTERN

Example

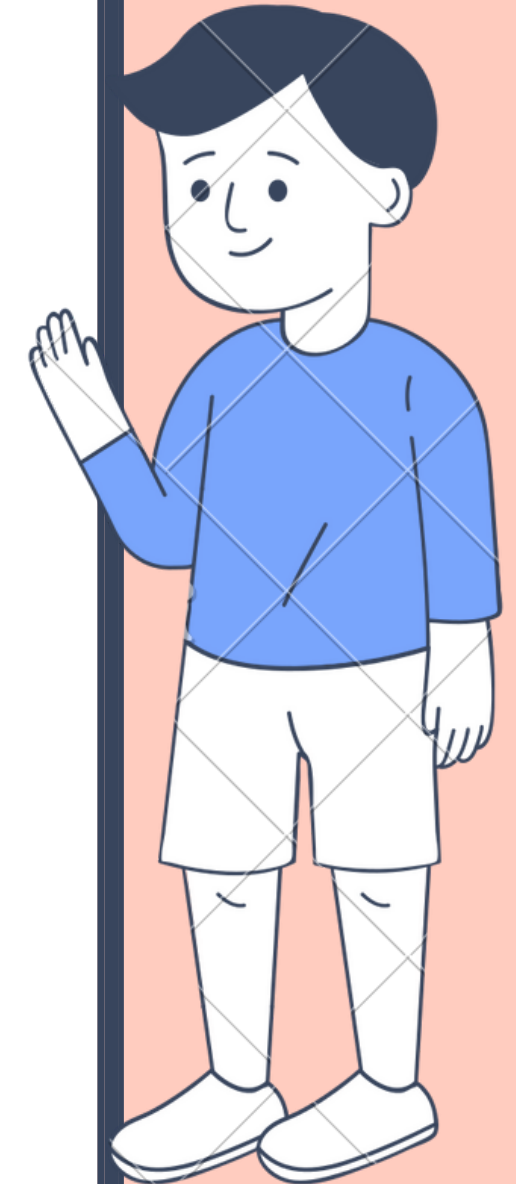
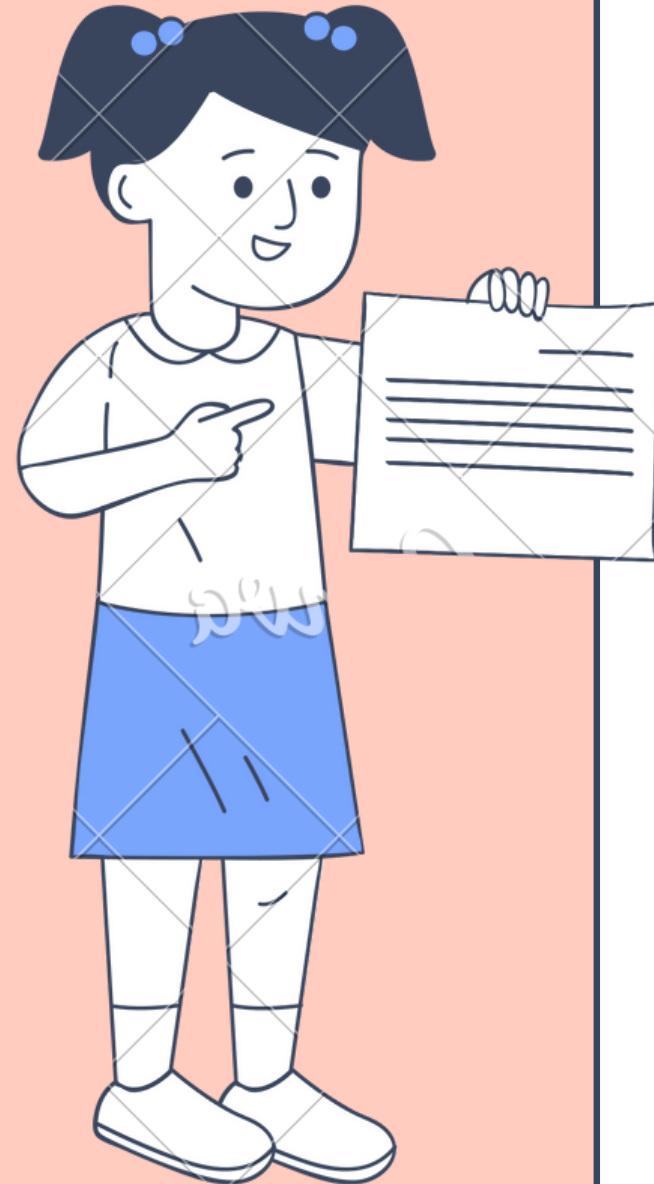
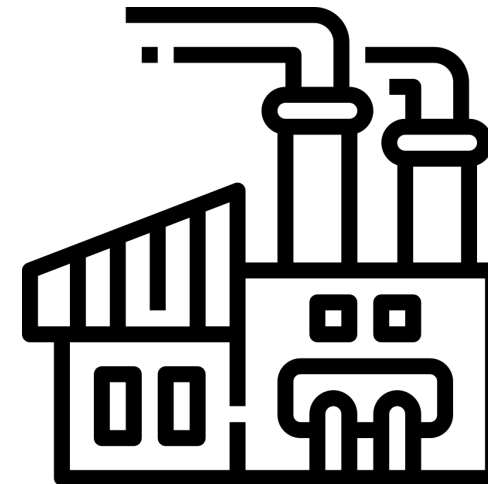


Usage

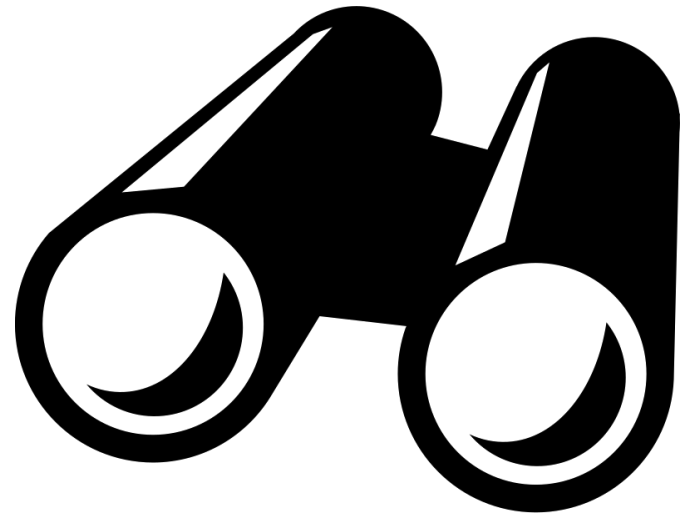
FACTORY METHOD



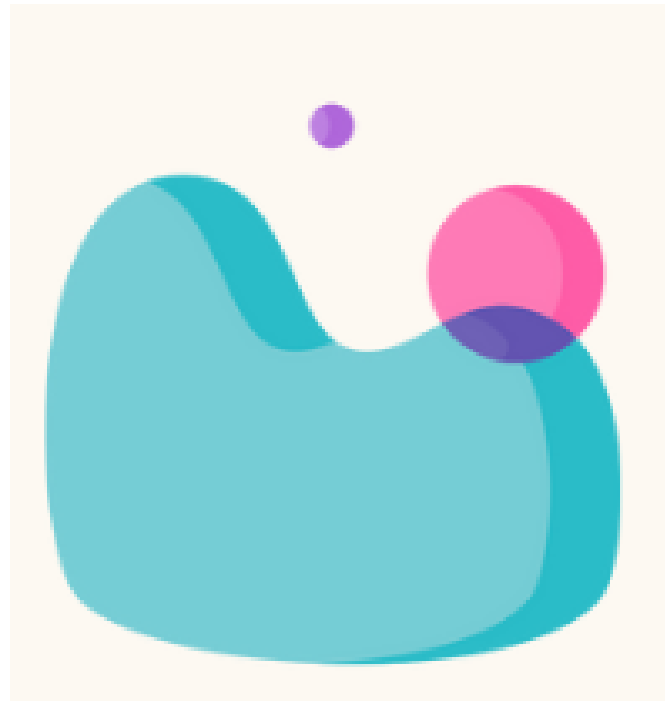
ABSTRACT
FACTORY METHOD



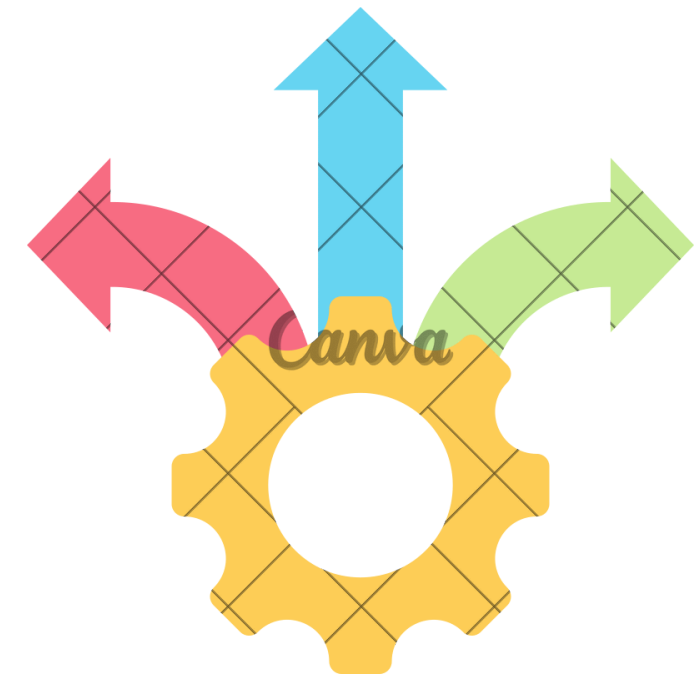
Key Differences



SCOPE

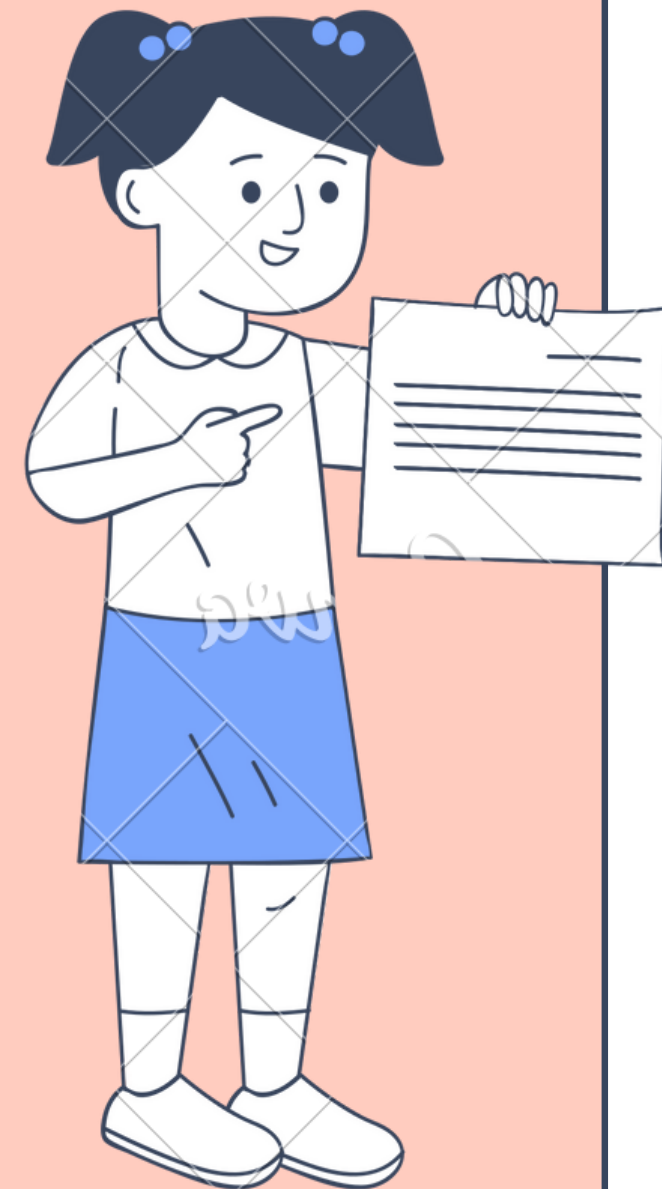


ABSTRACTION

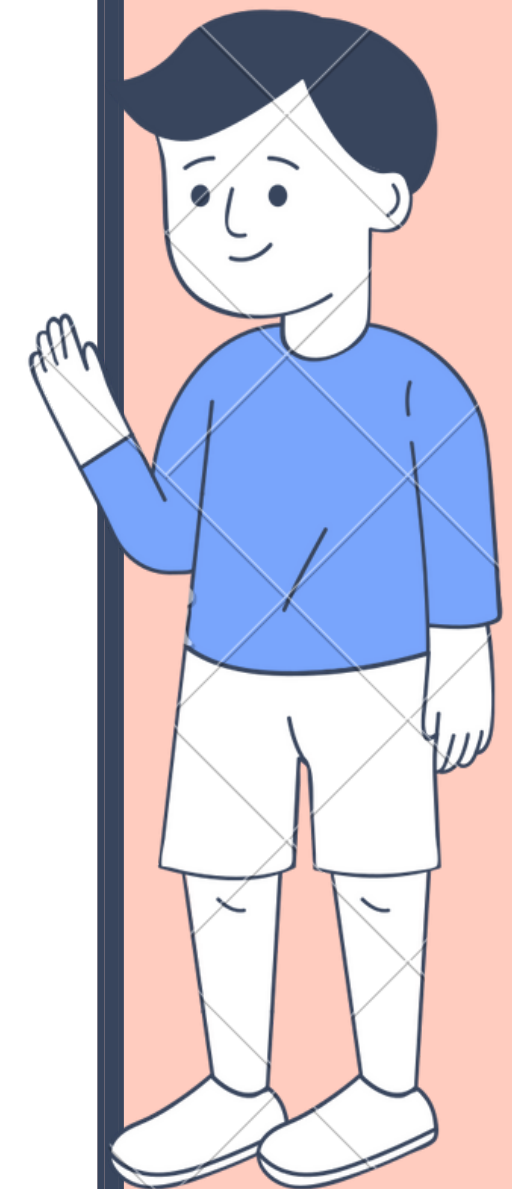


FLEXIBILITY

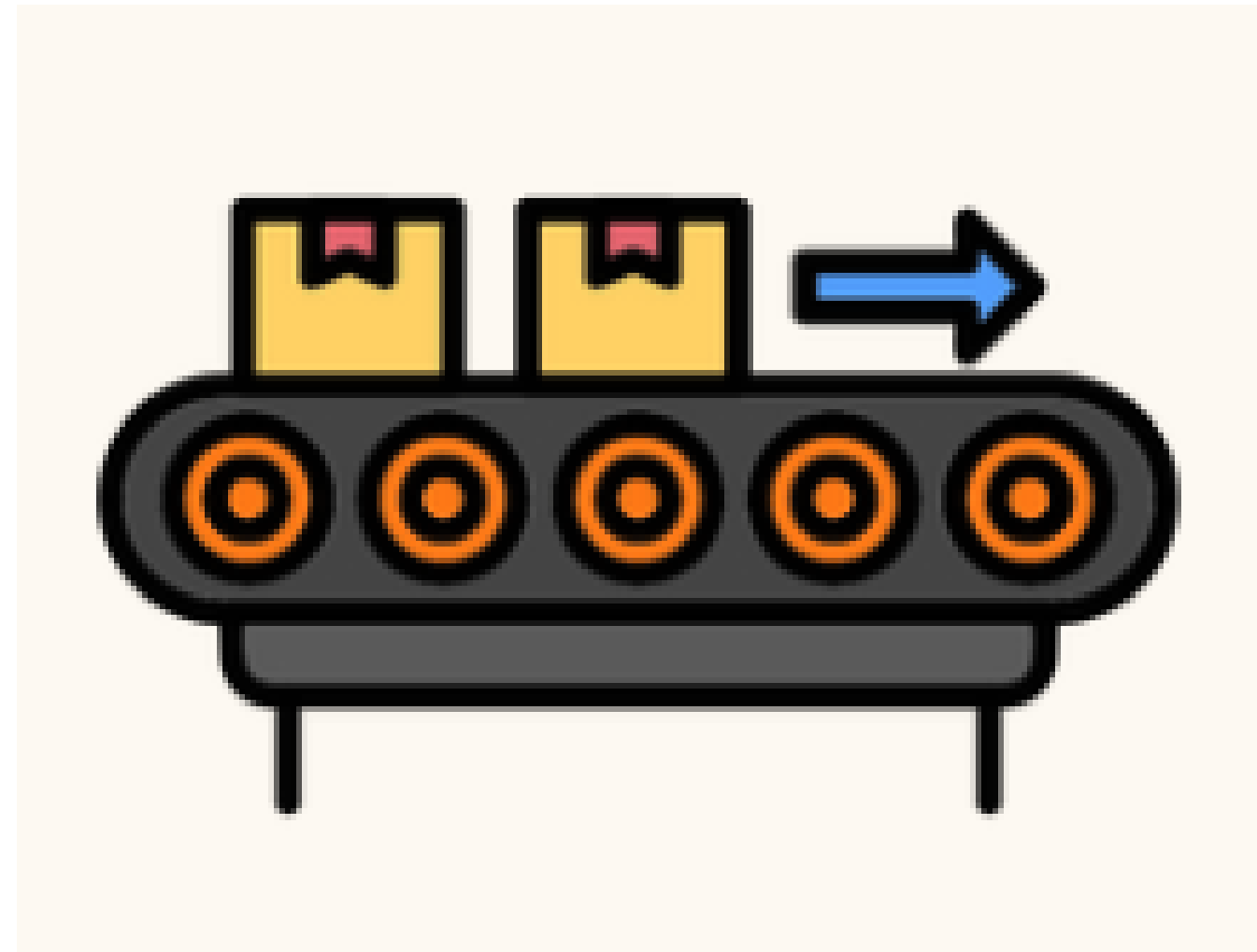




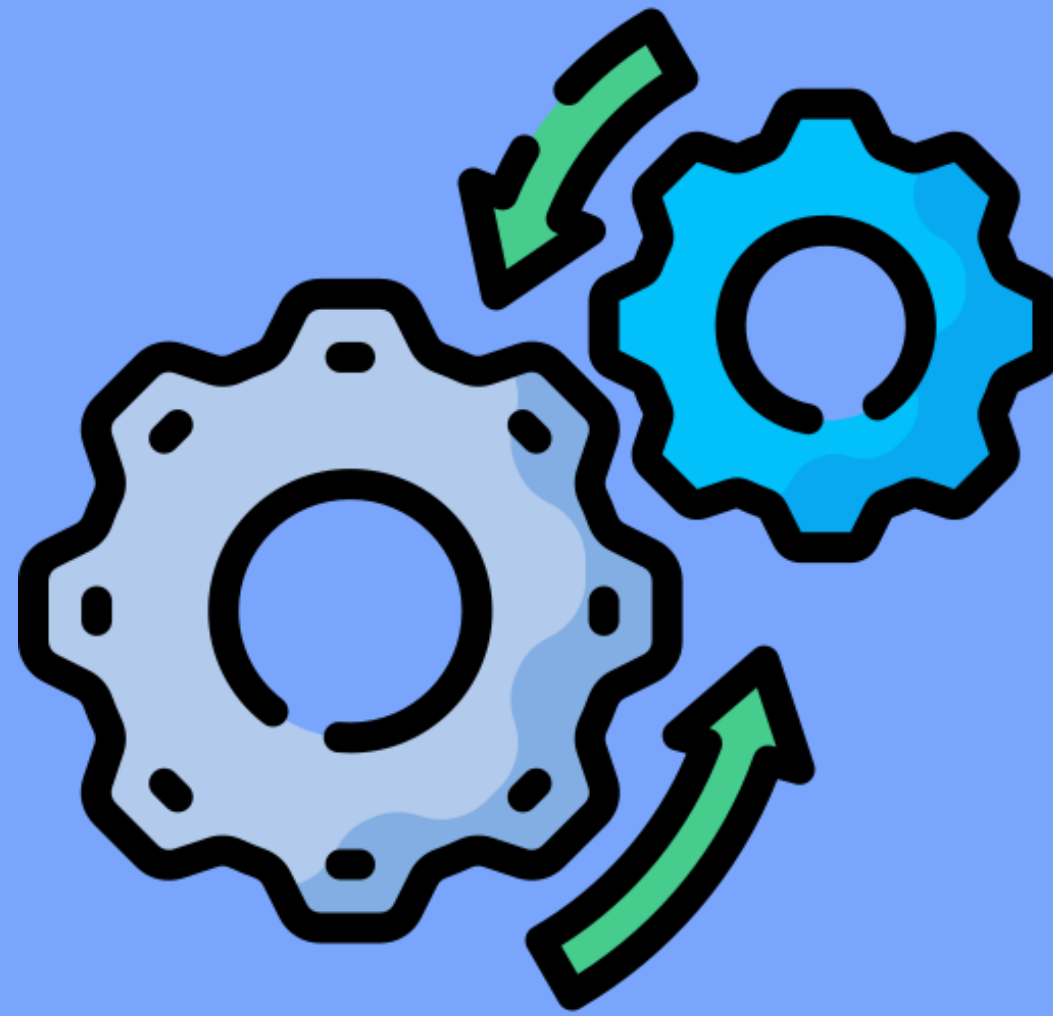
When do we use it?



**Decouple object
creation from
object usage**



Adding Dynamically

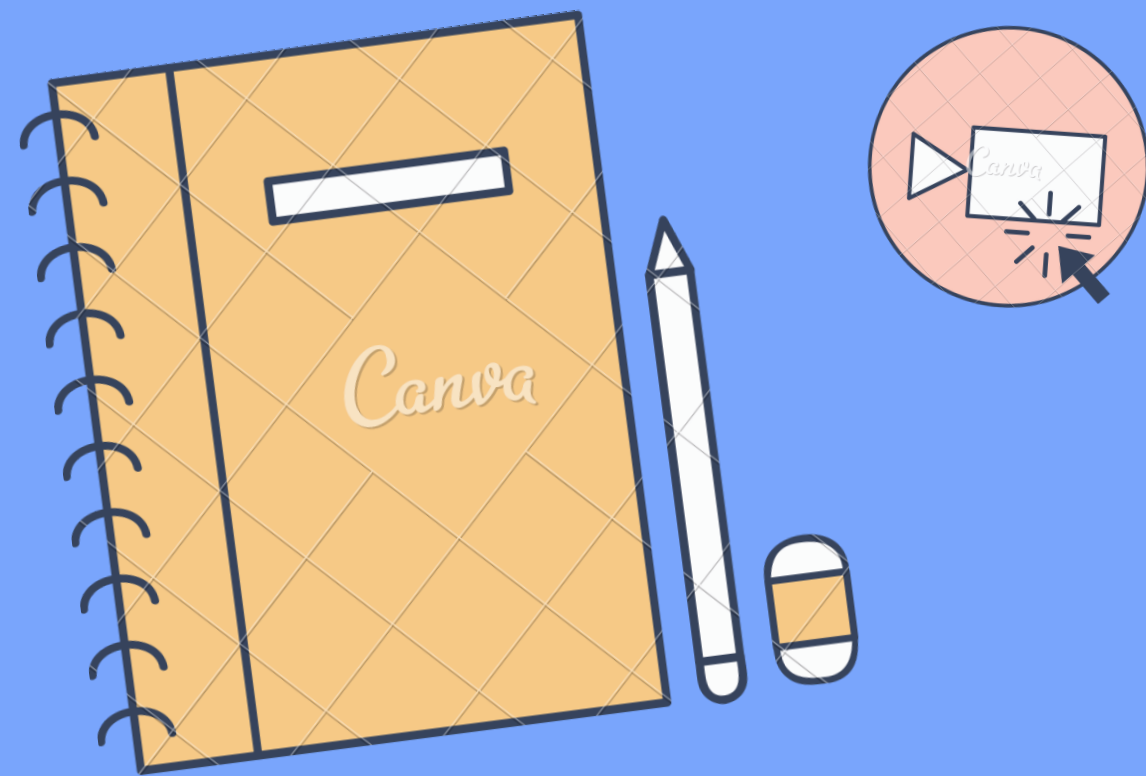




**Create object
based on runtime
condition**



Advantage



- **ENCAPSULATION**

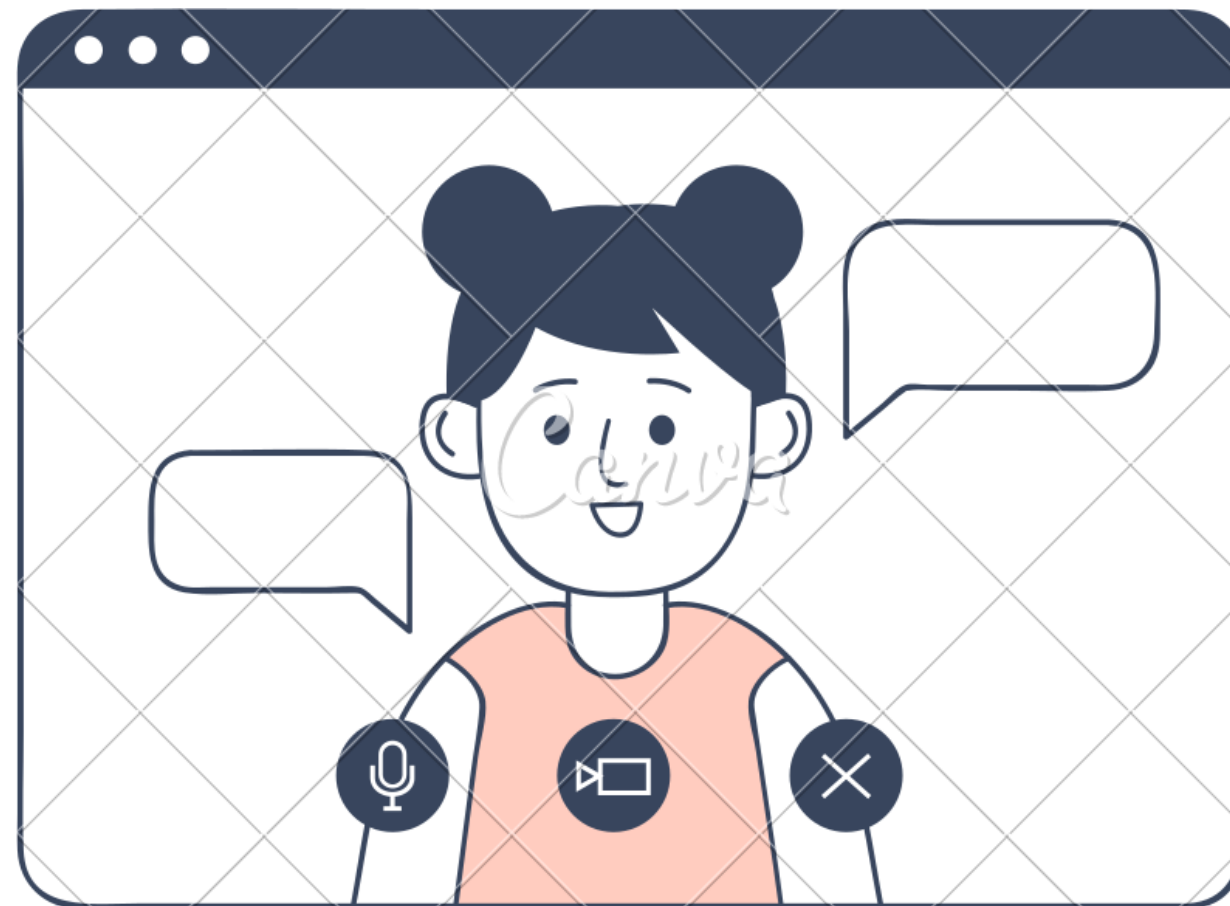
- **FLEXIBILITY**

- **DECOUPLING**

- **EXTENSIBILITY**



Code Showcase



Thank You

Do you have any
questions of us?

