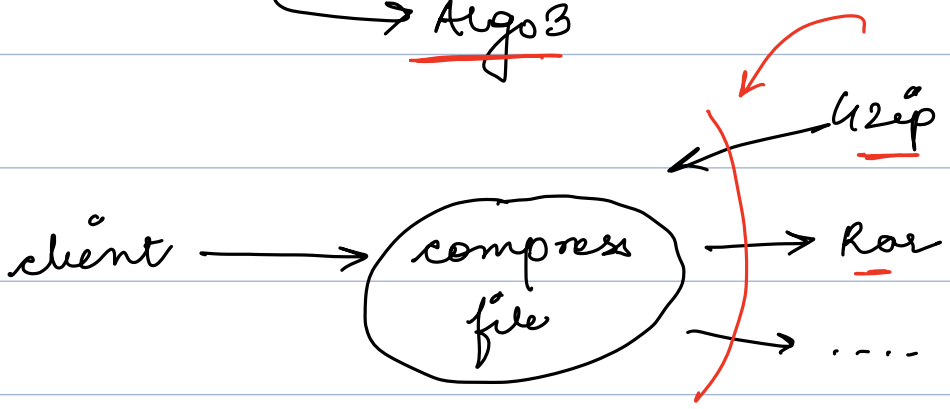
Behavioural → Strategy, Observer.

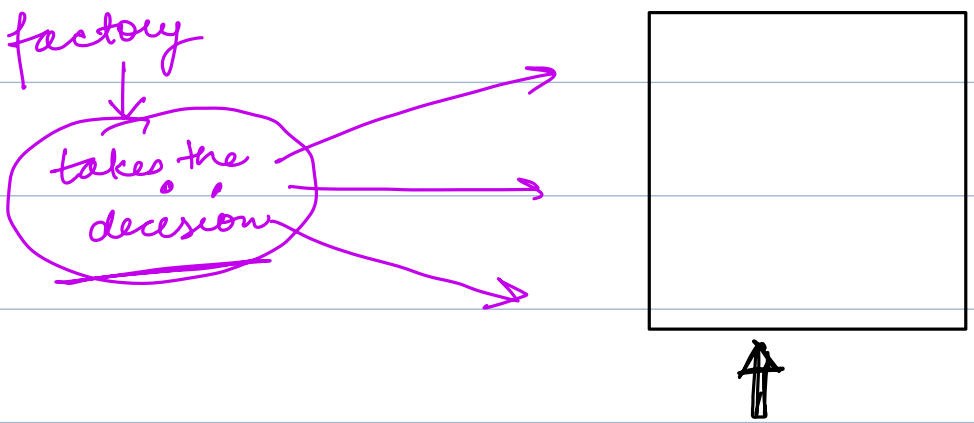
↓

concerned about behaviours or algorithms and assignment of responsibility

Strategy design pattern

Strategy is a behavioral design pattern that lets you define a family of algorithms, put each of them into a separate class, and make their objects interchangeable.

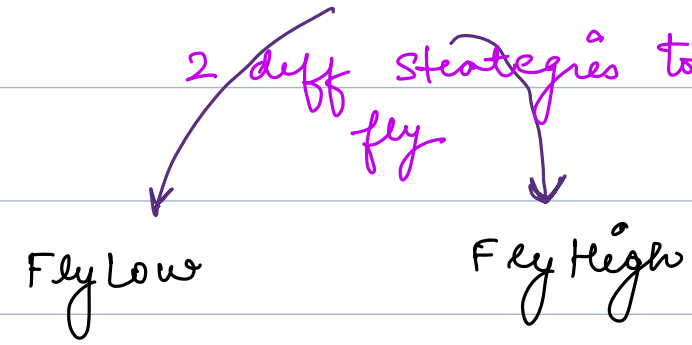




Bird

client wants to fly

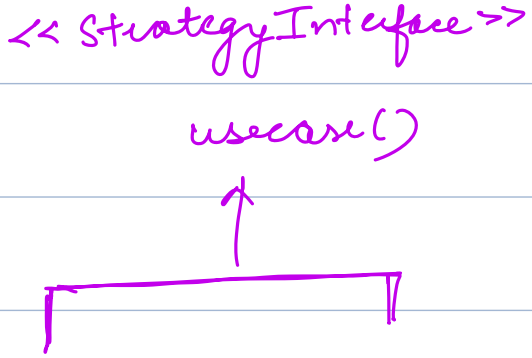
2 diff strategies to fly

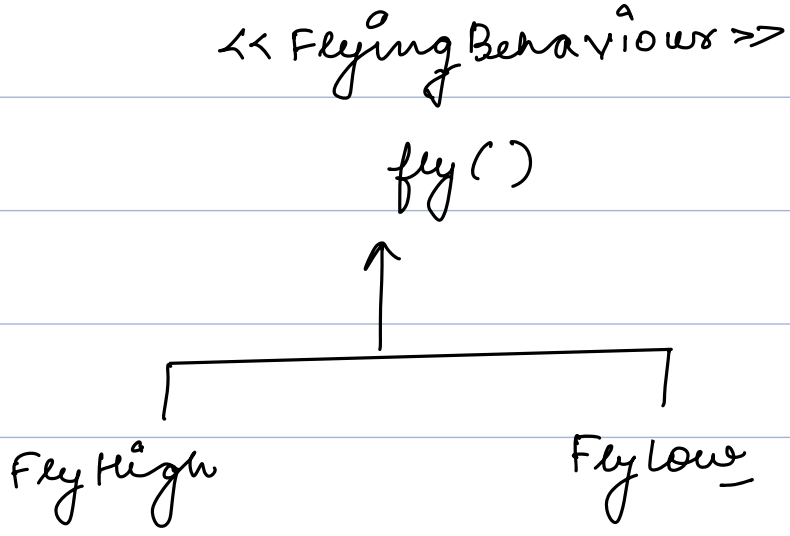


① algo were in the classes

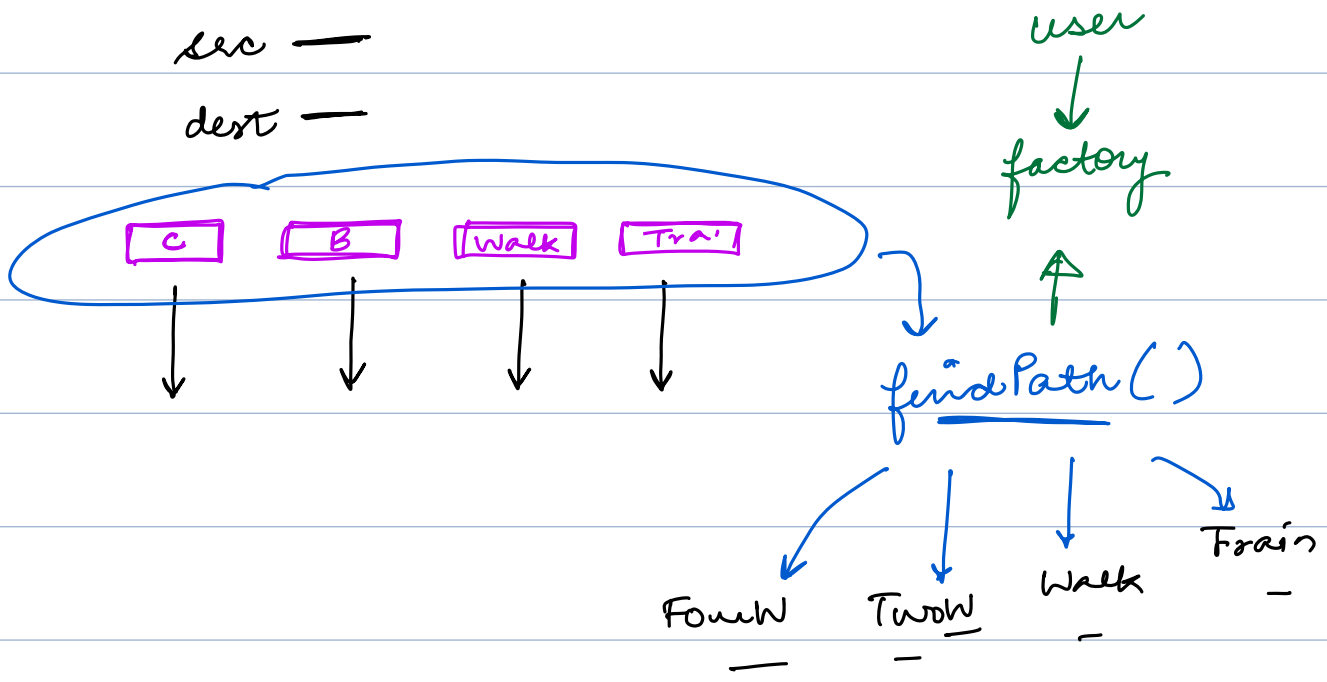
② Two classes
↓
Reusability ✔

③ Interface.





GoogleMaps

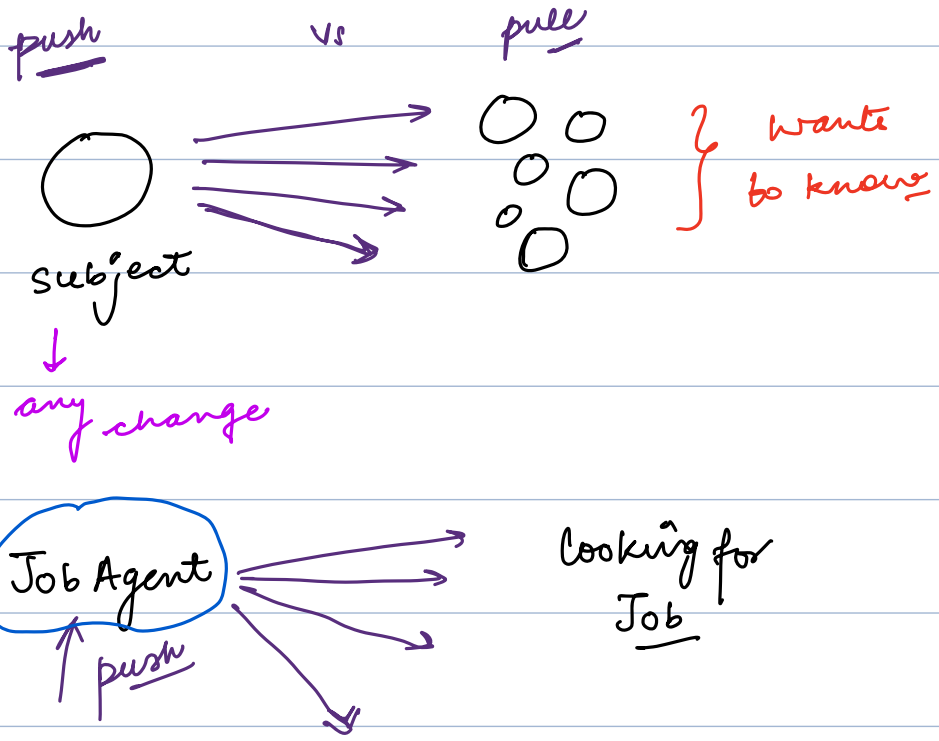

src —
dest —
C
B
Walk
Trai
user
factory
findPath()
FourW
TwoW
Walk
Train

Observer Design Pattern

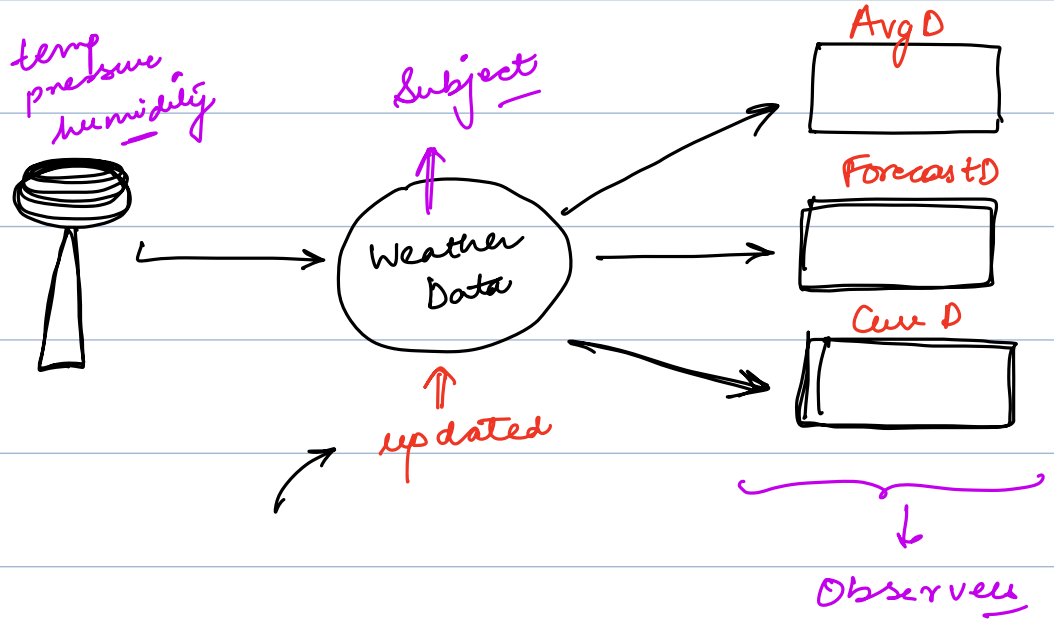
↓

some kind of observation

The Observer Design Pattern is a behavioral design pattern that defines a one-to-many dependency between objects. When one object (the subject) changes state, all its dependents (observers) are notified and updated automatically.



Weather Station



<< Subject >>

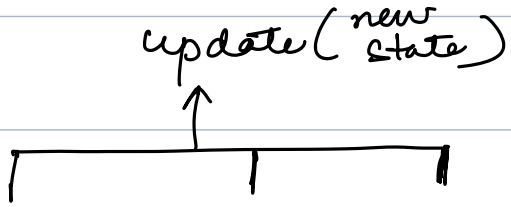
notify observers ()

register Observer ()

remove Observer ()

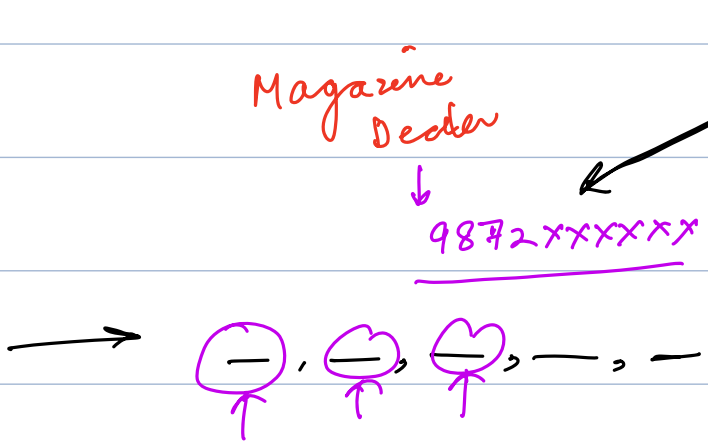
<< Observer >>

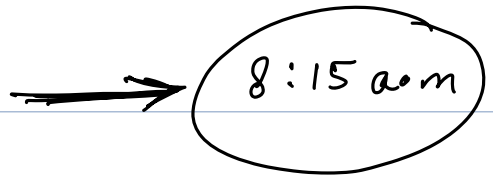
update (new state)



Magazine Dealer

9872xxxxxx




8:15 a.m