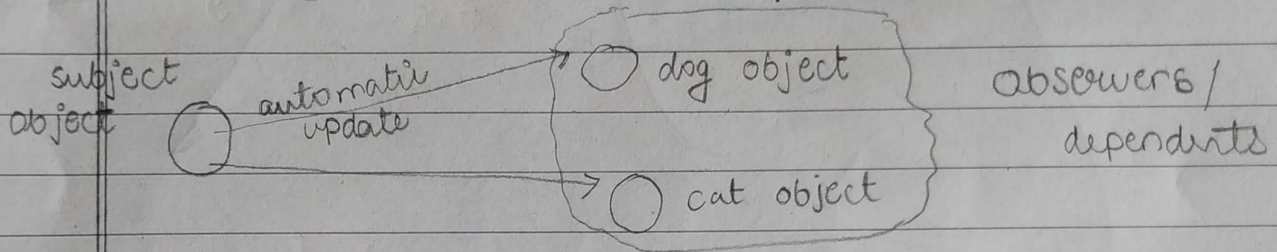


The Observer Pattern

Observer pattern notifies observers when something significant happens.

Observer pattern defines a one-to-many dependency between objects so that when object changes state, all its dependents are notified and updated automatically.

E.g. - In a newspaper agency, all of its subscribers get new edition as soon as it is printed.



When two objects are loosely coupled they have little knowledge of each other during interaction.

In our case the 'Subject' knows very little about the Observers. They implement the 'Observer' interface.

Design

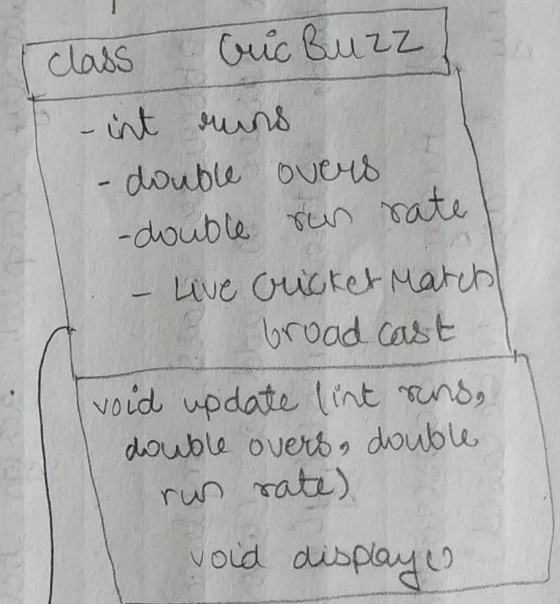
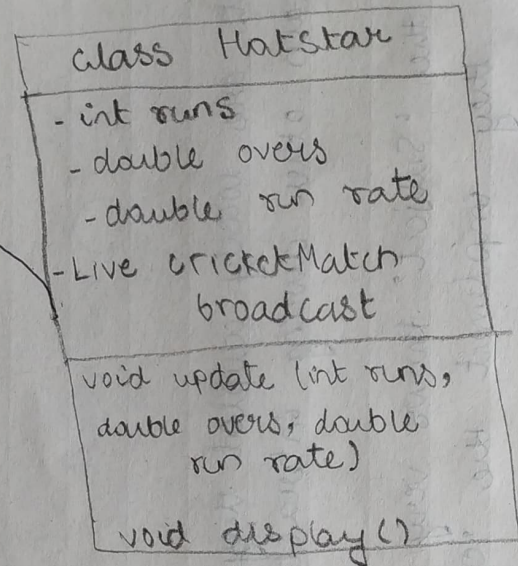
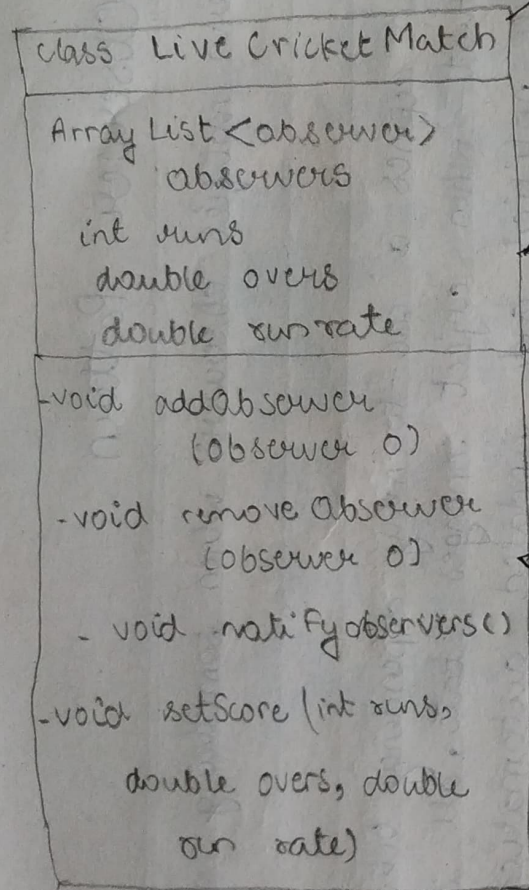
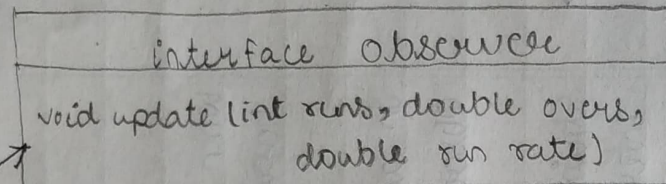
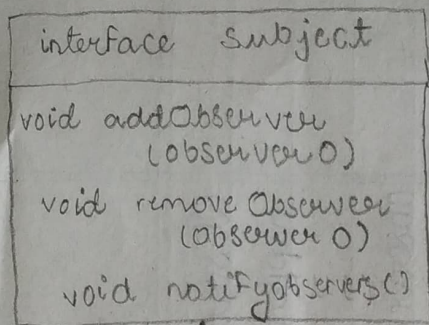
Principle →

Strive for loosely coupled design between objects that interact.

Answer →

Why every Observer class implements Observer interface? It becomes easier to add or remove Observers in Subject class, without modifying it.

with help of ArrayList <Observer> data type.
(flexibility & ease increased)



observers
reference

subject

subject

The Observer Pattern.