- BMH1077754
@q$v+]4z}Jn

C++ Training - Design Pattern

Design Pattern is used for Scalability

~~Can python generate binary?~~

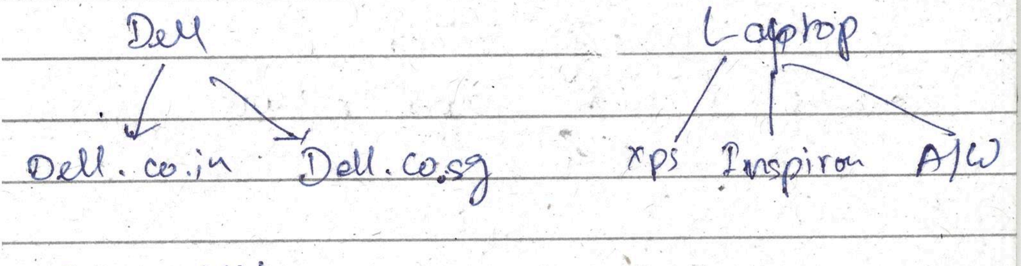
Structural
Creational
Behaviorial

· Composite Design Pattern → structural
↳ Tree view ~~based~~
Composite - Leaf

This design pattern is searchable
Helps to add/remove Leaf's or composite's easily.

2. Factory Method → Creational

Creator Product



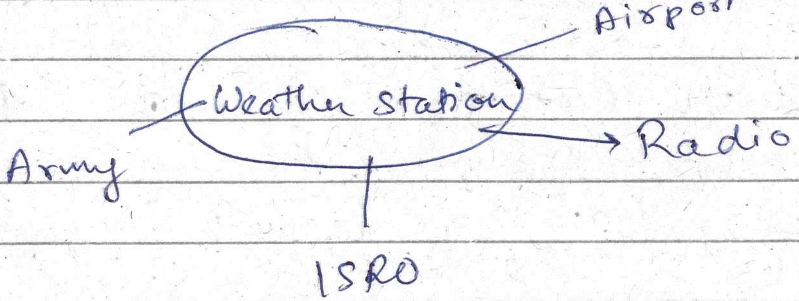
Scalability.

Create simple customized objects.

3. Template Method

Use this to implent a workflow

4. Observer → Behavioral
or notification pattern



One to many relation.
Consumer- Provider architecture

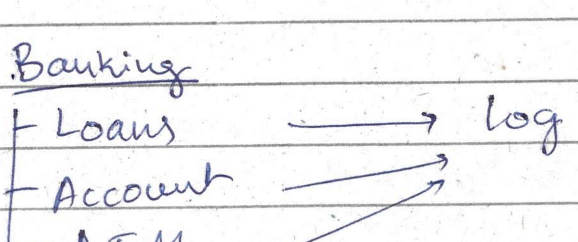
5. Horse Shoe

6. FACADE

Combine multiple interface into smaller set of interface to simplify.

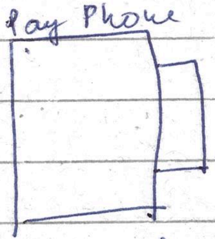
Ref: sourcemaking.com

7. Singleton



Only one instance of Log
constructor is private

8. State



Conditional
↳ you can dial only if you pay.

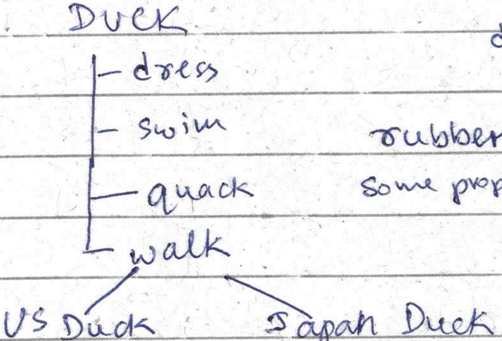
Number of states are scalable.
New State should be possible add.

9. Strategy 3 Actors → context, strategy, → behavior ~~quack~~ concrete strategy
↓ Duck ↓ dress

Action Game

Indian Duck

Yoga
↳ others cant



rubber duck
Some properties dont apply

Dress-Cowboy Samurai

Behavior mismatch during inheritence.
Take out mismatching properties and create new class.

10. Abstract Factory

Provide user flexibility to chose features

~~Creator~~ ~~Product~~

Deep customization

Use when to create family of objects

Trainer: Mahavir
Email: gurumahaveer@gmail.com