

Add a comment...

2- What we mean by coding against interface rather than class? and if u get it so What we mean by code against abstraction not concreteness?

Coding Against an Interface vs Coding Against a Class

Coding against an interface means:

Writing your code to depend on an interface (or abstract contract) rather than a specific concrete class.

Coding Against Abstraction, Not Concreteness

Abstraction => an interface or abstract class that defines behavior but hides implementation.

Concreteness => a specific class with exact implementation.

Coding against abstraction means:

Your code depends on “**what it should do**”, not “**how it does it**”.

3- What is abstraction as a guideline and how we can implement this through what we have studied?

Abstraction is:

The practice of hiding complex implementation details and exposing only the essential functionality to the user.

It is a design guideline that tells us:

- Focus on what an object does, not how it does it.
- Expose only necessary operations.
- Hide unnecessary details and internal complexity.