**؟What’s a Design Pattern**

1-The first stage in this explanation is knowing what Pattern Design means, and how it differs from algorithms.

2-And it can Saying that the Design patterns are the standard solution to most of the problems that are usually repeated when designing any software

3-This design can be customized to suit the design software until the problems related to the code are resolved!

4-And you should know that the pattern design cannot be searched for and then copied to the project as if it were a piece of code!, because It represents the concept of how to design the software code and solve specific problems with the design of the software! What he does is to understand the details of the pattern design and then try to apply the concepts related to it within the program .your problem to be solved by applying this understanding ,and This means that the implementation of this concept may differ from code.to others in proportion to the method of implementing the program and without deviating from its own concept!

**?Why Should I Learn Patterns**

There are many programmers who have not heard of or are aware of any pattern design and yet they practice their work in programming Ordinarily, as there are a lot of programmers who use the pattern design and don't realize it, this is Although the groups of programmers continued their work and succeeded in that, they lost an important part in the ability to develop skills And solve problems in the best way using the concepts of OOP, which we usually find these errors repeated frequently...From here, it can be said that we learn the pattern design for:

1-Because the pattern design represents a tool that contains a set of solutions that have been used to ensure their effective ness more times in solving the most common or recurring problems, and it teaches you new ways to solve problems based on OOP concepts, so you will be able to solve problems correctly, smarter, and better, and follow the rules more accurately better.

2-By knowing the pattern design, the team or programmers can communicate with each other to solve Problems only by suggesting the appropriate pattern design without having to explain this design pattern, if we assume that we are in a work team within one company that uses more than one programming language, and one of the team has a problem, I can simply refer to it using the Singleton or Decorator or Observer and without the need to explain this pattern design, and thus the work team will be able to propose solutions And processing it through concepts with ease...

3-With your understanding of the pattern design, you can plan to build your system before starting it and choosing The appropriate pattern for you, and this will make the process of designing any system better, more accurate, and the code Better and follow OOP standards correctly.

**What is the difference between Algorithms and Pattern Design?**

\*Many people get confused between Pattern and Algorithm, and the reason for that is because the two concepts refer to the best way to solve the problems that we may encounter, but the main difference is that the Algorithm defines A group of actions of sets through which the desired goal can be achieved, while the Pattern represents the degree of Above is the description of the solution that can be implemented in two different ways in two different programs.

\*Simply put, it can be said that the algorithms represent the necessary steps to reach the goal, while the Pattern represents what is required You will see results and features, but the exact order of implementation is up to you

**-What does the pattern consist of?**

\*Creational Pattern :-

“It is a group of patterns that are concerned with the mechanism or the method through which it is done Create an object in a way that makes it easy and more flexible to reuse it ”.

\*Structural Pattern :-

“The collection of patterns that takes care of how classes and objects are collected Within larger structures (while maintaining the efficiency and flexibility of structure ”

\*Behavioral Pattern :-

“It is a set of patterns that are concerned with the way objects communicate Effectively and how to distribute responsibilities among them”

**( Introduction to SOLID Principles**  **)**

**What Is The Solid Principle ?**

-These principles establish practices that lend to developing software with considerations for maintaining and extending as the project grows.

-They are five techniques for writing the code in a way that is easy to modify if needed at any time, and they can be applied before applying the concepts of the design pattern

S =>Single Responsibility Principle

O=>Open / Closed Principle

L=>Liskov Substitution

I=>interface Segregation Principle

D=>Dependency inversion Principle