

JAVA means DURGA SOFT

Design Patterns

Design Patterns introduction



India's No.1 Software Training Institute

DURGASOFT

www.durgasoft.com Ph: 9246212143 ,8096969696

DESIGN PATTERNS

Introduction

While using regular technologies (like Java ,C++,C etc) in product or application development there may be a chance of getting some problems repeatedly then a solution to that problem has been used to resolve those problems for getting better results. That solution is described as a pattern. Simply we can define a **pattern** as "A solution to a problem in a context". Each pattern is a three-part rule, which expresses a relation between a certain context, a problem and a solution". Patterns can be applied to many different areas of human endeavor, including software development.

If we use software technologies directly there is a chance of getting some side effects and problems in project development. We generally write some helper code or some helper resources to solve the problem in a best manner. This helper code or resources are called as **design patterns**.

In software engineering, a **design pattern** is a general reusable solution to a commonly occurring problem within a given context in software design. A design pattern is not a finished design that can be transformed directly into code. It is a description or template for how to solve a problem that can be used in many different situations.

The design patterns are language-independent strategies for solving common object-oriented design problems. When you make a design, you should know the names of some common solutions. Learning design patterns is good for people to communicate each other effectively.

LEARN FROM EXPERT & DIAMOND FACULTIES OF AMEERPET...

JAVA MEANS DURGASOFT

INDIA'S NO. 1 SOFTWARE TRAINING INSTITUTE

AN ISO 9001:2008 CERTIFIED
DURGA
SOFTWARE SOLUTIONS

#202 2nd FLOOR
www.durgasoft.com

040-64512786
+91 9246212143
+91 8096969696

Defining Design Patterns?

- ✓ Design patterns are solutions to recurring design problems you see over and over.
- ✓ A pattern addresses a recurring design problem that arises in specific design situations and presents a solution to it.
- ✓ Design patterns are the set of rules which comes as best solutions for recurring problems of project/application development.
- ✓ Design Patterns are proven solutions approaches to specific problems.

The Gang of Four (Gamma, Helm, Johnson and Vlissides) describes design patterns are **"descriptions of communicating objects and classes that are customized to solve a general design problem in a particular context."**

The design pattern identifies the participating classes and instances, their roles and collaborations, and the distribution of responsibilities. Each design pattern focuses on a particular object-oriented design problem or issue. It describes when it applies, whether it can be applied in view of other design constraints, and the consequences and trade-offs of its use.

- Designing object-oriented software is hard and designing reusable object-oriented software is even harder - Erich Gamma.
- Learning design patterns speeds up your experience accumulation in OOA/OOD.
- Design patterns are the best practices to use software technologies more effectively in project development.
- A design pattern names, abstracts, and identifies key aspects of a common design structure that makes it useful for creating a reusable object-oriented design.
- Well-structured object-oriented systems have recurring patterns of classes and objects.

Difference between Design patterns and Anti Patterns

- The best solution for a recurring problems is **Design patterns** where as the worst solution for recurring problems is **Anti patterns**.
- The organization ISO (International Standard Organization) maintains, both design patterns and anti patterns.

Note that the design patterns are not idioms or algorithms or components, just they are giving some hint to solve a problem effectively. Design patterns are no way related to with designing phase of the project they will be purely implemented in the development phase of the project.

Designing patterns can be implemented by using any programming language. Since Java is popular for large scale projects, we can see more utilization of design patterns in java.



Pattern Elements

In general, a pattern has four essential elements:

Pattern Name

JAVA Means DURGA SOFT

- ✓ Having a concise, meaningful name for a pattern improves communication among developers

Problem

- ✓ What is the problem and context where we would use this pattern?
- ✓ What are the conditions that must be met before this pattern should be used?

Solution

- ✓ A description of the elements that make up the design pattern
- ✓ Emphasizes their relationships, responsibilities and collaborations
- ✓ Not a concrete design or implementation; rather an abstract description

Consequences

- ✓ It normally examines the trade-offs, benefits and liabilities associated with applying the pattern.
- ✓ Includes impacts on reusability, portability, extensibility



Classification of Design Patterns

Mainly there are 3 basic classifications of patterns: Creational, Structural, and Behavioral patterns.

1. Creational Patterns - Concern the process of object creation

2. Structural Patterns- Deal with the composition of classes and objects
3. Behavioral Patterns- Deal with the interaction of classes and objects

Creational Patterns: how an object can be created i.e. creational design patterns are the design patterns that deal with object creation mechanisms, trying to create objects in a manner suitable to the situation. The basic form of object creation (using **new** keyword) could result in design problems or added complexity to the design. Creational design patterns solve this problem by somehow controlling this object creation.

All the creational patterns define the best possible way in which an object can be instantiated. These describes the best way to CREATE object instances. There are five types of creational patterns.

1. Abstract Factory Pattern
2. Builder Pattern
3. Factory Pattern
4. Prototype Pattern
5. Single Pattern

Structural Patterns: Structural design patterns are design patterns that ease the design by identifying a simple way to realize relationships between entities. Structural Patterns describe how objects and classes can be combined to form larger structures. There are seven structural patterns described. They are as follows:

1. Adapter Pattern
2. Bridge Pattern
3. Composite Pattern
4. Decorator Pattern
5. Facade Pattern
6. Flyweight Pattern
7. Proxy Pattern

Behavioral Patterns: Behavioral design patterns are design patterns that identify common communication patterns between objects and realize these patterns. By doing so, these patterns increase flexibility in carrying out this communication i.e. prescribes the way objects interact with each other. They help make complex behavior manageable by specifying the responsibilities of objects and the ways they communicate with each other. The 11 behavioral patterns are:

1. Chain of Responsibility Pattern
2. Command Pattern
3. Interpreter Pattern
4. Iterator Pattern
5. Mediator Pattern
6. Memento Pattern
7. Observer Pattern
8. State Pattern
9. Strategy Pattern
10. Template Pattern
11. Visitor Pattern

Now we can see the regularly used design patterns in real time java projects as they are classified based on the type of applications which we are developing.



Java Real Time Tools

JAVA TOOLS Means DURGASOFT

Multiple Faculty Members only For JAVA TOOLS

Online Training Class Room Training

DURGA SOFTWARE SOLUTIONS

www.durgasoftonlinetraining.com durgasoftonlinetraining@gmail.com Ph: +91- 8885252627 +91- 7207212428

DURGA SOFTWARE SOLUTIONS ,202 HUDA Maitrivanam, Ameerpet , Hyd. Ph: 040-64512786

Page 7

➤ **Core Java Level or Standalone Application Level Design Patterns**

- Factory Method Design Pattern
- Singleton Design Pattern
- Synchronized Singleton Design Pattern
- Factory Pattern Design Pattern
- Abstract Factory Design Pattern
- Builder Design Pattern
- Prototype Design Pattern
- Template Method Design Pattern
- Adapter Design Pattern
- Fast Line Reader Design Pattern
- DTO/VO Design Pattern
- IOC or Dependency Injection Design Pattern

➤ **Web Level Design Patterns**

- MVC1 & MVC2 Design Pattern
- View Helper Design Pattern
- Composite View Design Pattern
- Front Controller Design Pattern
- Abstract Controller Design Pattern
- Intercepting Filter Design Pattern

➤ **Enterprise Application Level Design Patterns**

- DAO Design Pattern
- DAO Factory Design Pattern
- Business Delegate Design Pattern
- Service Locator Design Pattern
- Session Façade Design Pattern
- Message Façade Design Pattern

LEARN FROM EXPERTS ...

COMPLETE JAVA

CORE JAVA, ADV. JAVA, ORACLE, STRUTS, HIBERNATE, SPRING, WEB SERVICES,...

COMPLETE .NET

C#.NET, ASP.NET, SQL SERVER, MVC 5 & WCF

TESTING TOOLS

MANUAL + SELENIUM

ORACLE | D2K

MSBI | SHARE POINT

HADOOP | ANDROID

C, C++, DS, UNIX

CRT & APTITUDE TRAINING

AN ISO 9001:2008 CERTIFIED

DURGA

Software Solutions®

202, 2nd Floor, HUDA Maitrivanam,
Ameerpet, Hyd. Ph: 040-64512786,

9246212143, 8096969696

www.durgasoft.com