Data Structure & Design Pattern

Design Principle, Algorithm

Travel from City A to City B

1. By Car/Bus/Two Wheeler
2. By Train
3. By Flight
4. By Ship/Boat

Factors deciding the solution

1. Time of travel
2. Comfortability
3. Cost & Time involved
4. Safety
5. Need (For what purpose)

Design pattern is a proven solution to resolve a particular challenge.

In Java

1. Core Java Design Patterns
2. Adv Java Design Patterns

Creational Design pattern (How we are creating the objects) Singleton, Factory, AbstractFactory, Builder, ObjectPool, ProtoType,

Behavioural Design Pattern (Iterator, Interpretor, ChainOfResposibility)

Structural Design Patern (Decorator, Façade, Flyweight)

Design Patterns are not programming lang specific.

Adv JAVA Design Patterns – MVC – web Application

Model, View , Controller

Model – Model (Data – Database – JPA, DAO, SQL related Codes)

View – It represents the UI (User Interface) – HTML (JSP/ASP/SpringBoot/Angular/React)

Controller – Represent the backend business logic, platform (Servlet, JSP, ASP/PHP)

DI – Dependency Injection – Injecting the required objects at the runtime by the container.

IoC – Inversion Of Control (Control is Inverted from Developer/programmer to the Framework)

Data Structure – Storing the data in a structured format.

In order to simplify the process of sorting, searching the data is stored in a proper format.

Arrays [Single Dimensional & Multi-Dimensional] [Homo & Hetero]

Structure , Class, Enum, Stack, Queue, List, Map, Set, Enumerations, Iterators

Stack – LIFO (Last In First Out) – Stacking 10 coins - push(), pop()

Push() – adding data to the stack (top)

Pop() – removing data from the stack (top)

List, LinkedList,

Queue – FIFO (First In First Out)

Sorting, Search – Storing the data in a structured format – Flat files only.