

Behavioral Design pattern

Behavioral design patterns focus on communication between objects, defining the interaction, responsibilities, and control flow among them. These patterns help manage complex control flows and allow objects to collaborate efficiently.

[Chain of Responsibility]

* Express Middleware pattern is example of this.

The **Chain of Responsibility** design pattern allows a request to pass through a chain of potential handlers. Each handler in the chain decides whether to handle the request or pass it to the next handler. This decouples the sender and receiver of a request, providing flexibility and scalability.

Key Concepts:

1. Handler Interface:

- Defines a method to handle requests and set the next handler.

2. Concrete Handlers:

- Implement the handler interface.
- Decide whether to process the request or forward it to the next handler.

3. Client:

- Sends the request to the chain.