➢ Behavioral Pattern

1. Behavioral patterns concerned with object interaction and their responsibilities
2. Behavioral pattern provides objects can easily talk with each other and still should be loose coupling

Command:
The command pattern is used to express a request, including the call to be made and all of its required parameters, in a command object. The command may then be executed immediately or held for later use.

Observer:
The observer pattern is used to allow an object to publish changes to its state. Other objects subscribe to be immediately notified of any changes.

Strategy:
The strategy pattern is used to create an interchangeable family of algorithms from which the required process is chosen at run-time.

State:
The state pattern is used to alter the behavior of an object as its internal state changes. The pattern allows the class for an object to apparently change at run-time.

Visitor:
The visitor pattern is used to separate a relatively complex set of structured data classes from the functionality that may be performed upon the data that they hold.

Mediator:
The mediator design pattern is used to provide a centralized communication medium between different objects in a system. This pattern is very helpful in an enterprise application where multiple objects are interacting with each other.

Memento:
The memento pattern is a software design pattern that provides the ability to restore an object to its previous state (undo via rollback).

Chain of Responsibility:
The chain of responsibility pattern is used to process varied requests, each of which may be dealt with by a different handler.