[State Design Pattern]

The **State Pattern** is a behavioral design pattern that allows an object to change its behavior when its internal state changes. It encapsulates state-specific behavior into separate classes, making it easy to add or modify states without affecting other parts of the system.

Key Concepts:

1. **Context**:
   - The main object whose behavior changes based on its state.
   - Maintains a reference to a State object.

2. **State Interface**:
   - Defines a common interface for all concrete states.
   - The context delegates state-specific behavior to the current state object.

3. **Concrete States**:
   - Implement the state-specific behavior.
   - These are the different states the Context can transition between.