| Name | Categories | Intent |
| --- | --- | --- |
| **Abstract Factory** | Creational | Allows the creation of objects without specifying their concrete type. |
| **Builder** | Creational | Uses to create complex objects. |
| Factory Method | Creational | Creates objects without specifying the exact class to create. |
| **Prototype** | Creational | Creates a new object from an existing object. |
| **Singleton** | Creational | Ensures only one instance of an object is created. |
| **Adapter** | Structural | Allows for two incompatible classes to work together by wrapping an interface around one of the existing classes. |
| Bridge | Structural | Decouples an abstraction so two classes can vary independently. |
| Composite | Structural | Takes a group of objects into a single object. |
| Decorator | Structural | Allows for an object’s behavior to be extended dynamically at run time. |
| Facade | Structural | Provides a simple interface to a more complex underlying object. |
| FlyWight | Structural | Reduces the cost of complex object models. |
| Proxy | Structural | Provides a placeholder interface to an underlying object to control access, reduce cost, or reduce complexity. |
| **Chain of Responsibility** | Behavioral | Delegates commands to a chain of processing objects. |
| **Command** | Behavioral | Creates objects which encapsulate actions and parameters. |
| **Interpreter** | Behavioral | Creates objects which encapsulate actions and parameters. |
| **Iterator** | Behavioral | Implements a specialized language. |
| **Mediator** | Behavioral | Allows loose coupling between classes by being the only class that has detailed knowledge of their methods. |
| **Memento** | Behavioral | Provides the ability to restore an object to its previous state. |
| **Observer** | Behavioral | Is a publish/subscribe pattern which allows a number of observer objects to see an event. |
| **State** | Behavioral | Allows an object to alter its behavior when its internal state changes. |
| **Strategy** | Behavioral | Allows one of a family of algorithms to be selected on-the-fly at run-time. |
| **Template Method** | Behavioral | Defines the skeleton of an algorithm as an abstract class, allowing its sub-classes to provide concrete behavior. |
| **Visitor** | Behavioral | Separates an algorithm from an object structure by moving the hierarchy of methods into one object. |