Summary of Structural design Patterns

1. Adapter - Allows objects with incompatible interfaces to collaborate.

2. Bridge – Lets you split a large class or a set of closely related classes into two separate hierarchies – abstraction and implementation – which can be developed independently of each other.

3. Composite – Lets you compose objects into tree structures and then work with these structures as if they were individual objects.

4. Decorator – Lets you attach new behaviors to objects by placing these objects inside special wrapper objects that contain the behaviors.

5. Façade – Provides a simplified interface to a library, a framework, or any other complex set of classes.

6. Flyweight – Lets you fit more objects into the available amount of RAM by sharing common parts of state between multiple objects instead of keeping all of the data in each object.

7. Proxy – Lets you provide a substitute or placeholder for another object. A proxy controls access to the original object, allowing you to perform something either before or after the request gets through to the original object.