1. Builder Pattern :   
   (<https://www.journaldev.com/1425/builder-design-pattern-in-java>)
2. Factory, Abstract Factory  
   (<https://www.javatpoint.com/prototype-design-pattern>)
3. Strategy  
   (<https://www.javatpoint.com/prototype-design-pattern>)
4. Adapter  
   (<https://www.javatpoint.com/adapter-pattern>)
5. SOLID principle  
   (<https://medium.com/@mari_azevedo/s-o-l-i-d-principles-what-are-they-and-why-projects-should-use-them-50b85e4aa8b6>)
6. Decorator  
   (<https://www.journaldev.com/1540/decorator-design-pattern-in-java-example>)
7. Facade  
   (<https://www.javatpoint.com/facade-pattern>)
8. Initilization-On-Demand lazy loading singleton design pattern implementation  
   (<https://en.wikipedia.org/wiki/Initialization-on-demand_holder_idiom>)
9. Injecting prototype bean in singleton spring  
   (<https://www.netjstech.com/2016/02/injecting-prototype-bean-in-singleton-spring.html>)