**Objectives**

* Demonstrate code modifications are done for any 2 of the SOLID principles
  + Modify code to implement Single responsibility, Interface segregation
* Demonstrate the usage of Abstract factory design pattern
  + Factory of objects created using factory design pattern
* Demonstrate the usage and implementation of Singleton design pattern
  + Realtime instance of significance of one object instance in memory, Eg: single DB connection object; how to achieve it using Private Static Constructor, thread safety

**Analyze and implement the following design patterns thru real life problems**

**Implement all these problems thru a Console application**

Singleton:

Please refer to the handson document available [here](https://cognizantonline.sharepoint.com/:w:/r/sites/GTP-Solutions/Gencsharepath/Shared%20Documents/Internship2020/FSE/DotNet/04%20-%20DesignPrinciples/References/GoF%20Hands%20on%20Exercises/GOF_SingletonHandsOn.docx?d=waf0742650af74b079783eeef077ac181&csf=1&e=jbS4tF).

Abstract factory:

Please refer to the handson document available [here](https://cognizantonline.sharepoint.com/:w:/r/sites/GTP-Solutions/Gencsharepath/Shared%20Documents/Internship2020/FSE/DotNet/04%20-%20DesignPrinciples/References/GoF%20Hands%20on%20Exercises/GOF_Abstract%20FactoryHandsOn.docx?d=w5f0eaac3ad334c1585f032c6012f06cd&csf=1&e=nFjhac).