**Solid Principle**

1. S-single Responsibility principle for Substitution->every java class must perform a single functionality. multiple functionalities within same class can mashup the code and reduced number of lines.
2. O-stands for Open & Closed Principle->states that according to new requirements it should be Open for extension and closed for modification
3. L-stands for Leskov’s Substitution Principle->used in inheritance basically derived classes should provide the complete substitute for Base class.

Suppose if A is the derived class of B then we should be able to replace A with B without interrupting the behaviour of class.

1. I-stands for Interface Segregation Principle->principle states that larger interface should break into smaller interface because implementation classes use only the methods that are required and the user should not force to implement all the methods even that they don’t want.
2. D-stands for Dependency Inversion principle->High level module should not depend on low level module but both should depend on abstraction or principles state that we must use abstraction(abstract classes and interafce)