Open Close principle,

*Software entities should be open for extension, but closed for modification.*

*—Bertrand Meyer*

This was the initial definition given by xyz in 1988. But there was another definition given by Robert B Martin (uncle Bob) which is more verbose version of original.

*” Open for extension*.*” This means that the behavior of the module can be extended*. *As the requirements of the application change, we are able to extend the module with new behaviors that satisfy those changes*. *In other words, we are able to change what the module does.*

*“Closed for modification.” Extending the behavior of a module does not result in changes to the source or binary code of the module*. *The binary executable version of the module, whether in a linkable library, a DLL, or a Java .jar, remains untouched.*

*—Robert C. Martin, Agile Software Development: Principles, Patterns, and Practices (Prentice Hall, 2003)*

For both sides of the open/closed principle, Martin explains in further detail what is meant by the key terms from the Meyer definition. To be open for extension, Martin explains, developers must be able to respond to changing requirements and support new features. This must be achieved despite modules being closed to modification. Developers must support new functionality without editing the source code—or compiled assembly—of the existing modules.