SOLID

* S _ Single Responsibility
  every class that is created should handle a single
  responsibility.

* O — Open/Closed
  Software entities like classes, modules, functions should
  be open for extensions but closed for modifications. To add
  new features, classes should be extended.

* L _ Liskov Substitution
  whenever a parent - child (super class - sub class) relation
  is created, both the classes should be interchangeable and
  substitutable.

* I — Interface Segregation
  no clients should be forced to depend upon methods which
  it does not use.

* D — Dependency Inversion
  high level modules should not depend on low level modules.
  Both should depend on abstractions. Abstraction should not
  depend on details. Details should not depend upon abstractions.

SRP — One reason to change
  OCP — Extend, dont modify
  LSP — Subclass property
  ISP — Split big interfaces
  DIP — Depend on abstractions