The solid principles:

S O L I D

1.For the letter (S) it refers to "Single reusability principle"

The class should solve only one problem and it should has a single reason to change

2.(O) open/ closed pirnciple

The class should be opened extenssion closed for modification

3.(L) liskov sustitution principle

If you substitue any type with one of its subtypes the behavior should not change

4.(I) interface segration principle

Avoid making general interface contains all methods

5.(D) dependency inversion principle

Higher level classes should not know the implementation of low level classes but depends on abstruction.