

Professional Skills
Agile Fundamentals
Jira
Git
Databases Introduction
Java Beginner
Maven
Testing (Foundation)
Java Intermediate
<div><div></div>Optionals</div>
<div><div></div>JDBC CRUD</div>
<div><div></div>Exceptions</div>
<div><div></div>SOLID Principles</div>
<div><div></div>Single Responsibility</div>
<div><div></div>Open/Closed</div>
<div><div></div>Liskov Substituiton</div>
<div><div></div>Interface Segregation</div>
<div><div></div>Dependency Inversion</div>
<div><div></div>Best Practice</div>
<div><div></div>Design Patterns</div>
<div><div></div>Creational Design Patterns</div>
<div><div></div>Structural Design Patterns</div>
<div><div></div>Behavioural Design Patterns</div>
<div><div></div>Collection &amp; Map</div>
<div><div></div>HashSets</div>
<div><div></div>HashMaps</div>
<div><div></div>Enums</div>
<div><div></div>Logging</div>
<div><div></div>Generics</div>
<div><div></div>Lambda Expressions</div>
<div><div></div>Streams</div>
<div><div></div>Complexity</div>
<div><div></div>Input and Output</div>
<div><div></div>Local Type Inference</div>
HTML
CSS

# SOLID Principles

## Contents

- [Overview](#)
- [Design Principles](#)
- [SOLID](#)
- [Tutorial](#)
- [Exercises](#)

## Overview

In object-oriented programming, the **SOLID Principles** focus on good design and coding practice.

They are designed to encourage developers to create easily-maintainable, more understandable, and flexible software.

As such, they fit very well into **Agile** software development.

## Design Principles

There are several other design principles which are commonly used ion software development. The most common are:

- GRASP** - General Responsibility Assignment Software Principles
- DRY** - Don't Repeat Yourself
- KISS** - Keep It Simple, Stupid

Generally, these principles aim to reduce unnecessary complexity, improve readability, and help to solve common issues encountered in the development cycle.

## SOLID

**SOLID** is an acronym for five distinct principles, for each of which a module has been created:

- [S - Single Responsibility](#)
- [O - Open/Closed](#)
- [L - Liskov Substitution](#)
- [I - Interface Segregation](#)
- [D - Dependency Inversion](#)

## Tutorial

Not applicable.

## Exercises

Not applicable.

Javascript
Spring Boot
Selenium
Sonarqube
Advanced Testing (Theory)
Cucumber
MongoDB
Express
NodeJS
React
Express-Testing
Networking
Security
Cloud Fundamentals
AWS Foundations
AWS Intermediate
Linux
DevOps
Jenkins Introduction
Jenkins Pipeline
Markdown
IDE Cheatsheet