

[Dashboard](#) / [My courses](#) / [Konstruksi Perangkat Lunak \(..\)](#) / [SOLID Principles](#) / [\[Post Test\] SOLID principles](#)

Started on Wednesday, 19 October 2022, 7:42 AM

State Finished

Completed on Wednesday, 19 October 2022, 7:52 AM

Time taken 9 mins 40 secs

Grade 8.00 out of 13.00 (62%)

Question **1**

Complete

Mark 1.00 out of 1.00

The Dependency Inversion Principle states that:

Select one:

- ☐ a. public classes' dependencies should be declared as internal.
- ☐ b. abstractions should depend on high-level modules, which in turn should depend on low-level modules.
- ☒ c. high-level modules should not depend on low-level modules. Both should depend on abstractions.
- ☐ d. classes' dependencies should be packaged in separate packages.

Question **2**

Complete

Mark 0.00 out of 1.00

Following SOLID principles often results in having more files involved in the solution. What's the simplest way to manage these files?

Select one:

- ☐ a. Projects
- ☐ b. Areas
- ☐ c. A database
- ☐ d. Folders
- ☒ e. Namespaces

Question **3**

Complete

Mark 1.00 out of 1.00

Which SOLID principle is being violated if you keep having to add more to the same function again and again?

Select one:

- ☐ a. Dependency Inversion Principle
- ☐ b. Liskov Substitution Principle
- ☒ c. Open / Closed Principle
- ☐ d. Interface Segregation Principle

Question **4**

Complete

Mark 0.00 out of 1.00

When a class's elements all belong together, the class is said to be

Select one:

- ☐ a. Abstract
- ☐ b. Elegant
- ☐ c. Polymorphic
- ☒ d. Loosely coupled
- ☐ e. Cohesive

Question **5**

Complete

Mark 1.00 out of 1.00

The Open/Closed Principle states that software entities should be...

Select one:

- ☒ a. open to extension, but closed for modification.
- ☐ b. open to modification, but closed for extension.
- ☐ c. open to abstraction, but close to implementation.
- ☐ d. open to composition, but closed for inheritance.

Question 6

Complete

Mark 1.00 out of 1.00

The Liskov Substitution Principle states that...

Select one:

- ☒ a.subtypes must be substitutable for their base types.
- ☐ b.base types must be substitutable for their child types.
- ☐ c.generic types must be substitutable for non-generic types.

Question 7

Complete

Mark 0.00 out of 1.00

Which of the following in your code usually indicates a Liskov Substitution Principle violation?

Select one:

- ☐ a.NotImplementedException
- ☐ b.ArgumentNullException
- ☐ c.NullReferenceException
- ☒ d.InvalidCastException

Question 8

Complete

Mark 1.00 out of 1.00

The Interface Segregation Principle states that...

Select one:

- ☒ a.clients should not be forced to depend on methods they do not use.
- ☐ b.classes' interfaces should be defined in separate interface types/files.
- ☐ c.interfaces (abstractions) should be segregated into separate projects from the classes that implement them.
- ☐ d.read methods and write methods should belong to separate interfaces.

Question **9**

Complete

Mark 0.00 out of 1.00

An effective way to follow the Interface Segregation Principle when you already have a large interface is to...

Select one:

- ☒ a.create a single facade interface that encompasses every legal operation in the application layer.
- ☐ b.extract and call private methods from public methods.
- ☐ c.make every method belong to its own interface.
- ☐ d.define smaller interfaces and compose the large interface from the new, smaller interfaces.

Question **10**

Complete

Mark 1.00 out of 1.00

The use of 'as' or 'is' keywords to check the type of a parameter or variable before performing some operation on it is a symptom of which SOLID principle violation?

Select one:

- ☐ a.Single Responsibility Principle
- ☐ b.Open / Closed Principle
- ☐ c.Dependency Inversion Principle
- ☒ d.Liskov Substitution Principle

Question **11**

Complete

Mark 1.00 out of 1.00

How is the Single Responsibility Principle best summarized?

Select one:

- ☐ a.Segregate responsibilities using small interfaces
- ☐ b.Simple applications should do one thing, and do it well
- ☒ c.Each software module should have one and only one reason to change
- ☐ d.Static methods must do have a single responsibility

Question **12**

Complete

Mark 0.00 out of 1.00

Which design pattern is effective for dealing with large interfaces your application depends on, but can't change?

Select one:

- ☒ a.Factory
- ☐ b.Adapter
- ☐ c.Repository
- ☐ d.Builder
- ☐ e.Flyweight

Question **13**

Complete

Mark 1.00 out of 1.00

What does the phrase, "New is glue," mean in relation to the Dependency Inversion Principle?

Select one:

- ☐ a.Prefer direct static calls to use of the 'new' keyword
- ☐ b.To reduce dependencies, class instances should only be created using reflection
- ☐ c.The 'new' keyword is the glue that holds your application together
- ☒ d.Using 'new' to create dependencies glues your code to that dependency

[Previous Activity](#)[Jump to...](#)

Next Activity