API design principles, Design patterns

Object-oriented software design

Dr. Balázs Simon

BME, IIT

API design principles: True or false

True or false

Is the following statement true or false?

When designing an API,
the implementation of the API precedes
the writing of the code examples for the API.

True or false

Is the following statement true or false?

Solution: False

When designing an API,
the implementation of the API precedes
the writing of the code examples for the API.

True or false

Is the following statement true or false?

In principle, a good API provides
all the functions the users need.
This is easily achievable in practice
after we have collected the users' requirements.

True or false

Is the following statement true or false?

Solution: False

In principle, a good API provides
all the functions the users need.
This is easily achievable in practice
after we have collected the users' requirements.

True or false

Is the following statement true or false?

It is recommended to publish API functions early,
so that users can benefit from them as soon as they can.

True or false

Is the following statement true or false?

Solution: False

It is recommended to publish API functions early,
so that users can benefit from them as soon as they can.

True or false

Is the following statement true or false?

An API is bad, if we need more than three
lines of program code to start it
when we want to deviate from the default behavior.

True or false

Is the following statement true or false?

Solution: False

An API is bad, if we need more than three
lines of program code to start it
when we want to deviate from the default behavior.

True or false

Is the following statement true or false?

An API should throw an exception immediately when an incoming parameter is wrong.

True or false

Is the following statement true or false?

Solution: True

An API should throw an exception immediately when an incoming parameter is wrong.

True or false

Is the following statement true or false?

We don’t have to document our API
if we implement it according to the clean code principles.

True or false

Is the following statement true or false?

Solution: False

We don’t have to document our API
if we implement it according to the clean code principles.

True or false

Is the following statement true or false?

We shouldn't add convenience methods to the API, since we should provide only a single way to do things.

True or false

Is the following statement true or false?

Solution: False

We shouldn't add convenience methods to the API, since we should provide only a single way to do things.

API design principles: Selection

Which of the following guidelines are common in the API design principles and in the Clean code principles?

A. Use the local dialect
B. Avoid long parameter lists
C. Design and document for inheritance or else prohibit it
D. Document the API
E. Use convenience methods
F. Favor unchecked exceptions
G. Avoid abbreviations
H. Avoid side-effects

Which of the following guidelines are common in the API design principles and in the Clean code principles?

A. Use the local dialect
B. Avoid long parameter lists
C. Design and document for inheritance or else prohibit it
D. Document the API
E. Use convenience methods
F. Favor unchecked exceptions
G. Avoid abbreviations
H. Avoid side-effects

Solution

Design Patterns: True or false

True or false

Is the following statement true or false?

The Singleton pattern can be used in Dependency Injection.

True or false

Is the following statement true or false?

Solution: True

The Singleton pattern can be used in Dependency Injection.

True or false

Is the following statement true or false?

We use the Factory Method pattern when we know the type of the object to be instantiated in advance.

True or false

Is the following statement true or false?

Solution: False

We use the Factory Method pattern when we know the type of the object to be instantiated in advance.

True or false

Is the following statement true or false?

There is a variant of the Abstract Factory patten in which the AbstractFactory class is concrete.

True or false

Is the following statement true or false?

Solution: True

There is a variant of the Abstract Factory patten in which the AbstractFactory class is concrete.

True or false

Is the following statement true or false?

The Abstract Factory pattern cannot be used in Dependency Injection.

True or false

Is the following statement true or false?

Solution: False

The Abstract Factory pattern cannot be used in Dependency Injection.

True or false

Is the following statement true or false?

The goal of the Builder pattern is to make sure that a family of products is always instantiated together.

True or false

Is the following statement true or false?

Solution: False

The goal of the Builder pattern is to make sure
that a family of products is
always instantiated together.

True or false

Is the following statement true or false?

The Prototype pattern always creates a deep-copy.

True or false

Is the following statement true or false?

Solution: False

The Prototype pattern always creates a deep-copy.

True or false

Is the following statement true or false?

There is a variant of the Dependency Injection patten which can inject circular dependencies.

True or false

Is the following statement true or false?

Solution: True

There is a variant of the Dependency Injection patten which can inject circular dependencies.

True or false

Is the following statement true or false?

The Object Pool pattern is especially efficient
if must manage external resources,
and not merely memory.

True or false

Is the following statement true or false?

Solution: True

The Object Pool pattern is especially efficient
if must manage external resources,
and not merely memory.

True or false

Is the following statement true or false?

In the Flyweight pattern the state of the Flyweight object can be changed by the client.

True or false

Is the following statement true or false?

Solution: False

In the Flyweight pattern the state of the Flyweight object can be changed by the client.

True or false

Is the following statement true or false?

In the Decorator pattern the chain of calls is usually terminated by one of the Decorators.

True or false

Is the following statement true or false?

Solution: False

In the Decorator pattern the chain of calls is usually terminated by one of the Decorators.

True or false

Is the following statement true or false?

Unlike Adapter, Decorator enhances another class without changing its interface.

True or false

Is the following statement true or false?

Solution: True

Unlike Adapter, Decorator enhances another class without changing its interface.

True or false

Is the following statement true or false?

There is a variant of the Composite pattern that violates LSP.

True or false

Is the following statement true or false?

Solution: True

There is a variant of the Composite pattern that violates LSP.

True or false

Is the following statement true or false?

In the Proxy pattern the Proxy object
may not have a direct reference to the Service object.

True or false

Is the following statement true or false?

Solution: True

In the Proxy pattern the Proxy object
may not have a direct reference to the Service object.

Design Patterns: Applications

True or false

Which design pattern solves the following task?

Define an interface for creating an object in a superclass, but let subclasses decide which class to instantiate.

True or false

Which design pattern solves the following task?

Solution: Factory Method

Define an interface for creating an object in a superclass, but let subclasses decide which class to instantiate.

True or false

Which design pattern solves the following task?

Dynamically attach additional responsibilities
to an object by wrapping it,
without changing its interface.

True or false

Which design pattern solves the following task?

Solution: Decorator

Dynamically attach additional responsibilities
to an object by wrapping it,
without changing its interface.

True or false

Which design pattern solves the following task?

Share common state to support large numbers of objects efficiently.

True or false

Which design pattern solves the following task?

Solution: Flyweight

Share common state to support large numbers of objects efficiently.

True or false

Which design pattern solves the following task?

Provide an interface for creating families of related or dependent objects without specifying their concrete classes.

True or false

Which design pattern solves the following task?

Solution: Abstract Factory

Provide an interface for creating families of related or dependent objects without specifying their concrete classes.

True or false

Which design pattern solves the following task?

Keep a set of initialized objects ready to use rather than allocating and destroying them on demand.

True or false

Which design pattern solves the following task?

Solution: Object Pool

Keep a set of initialized objects ready to use rather than allocating and destroying them on demand.

Design Patterns: matching

Match the following examples with the most appropriate design pattern!

1. Object.clone() in Java
2. Request processing threads of a web server
3. One common global database connection
4. Configuring a logger using a fluent API
5. Initiation of UI support for Windows or Linux platform

A. Singleton
B. Abstract Factory
C. Builder
D. Prototype
E. Object Pool

1	2	3	4	5

Match the following examples with the most appropriate design pattern!

1. Object.clone() in Java
2. Request processing threads of a web server
3. One common global database connection
4. Configuring a logger using a fluent API
5. Initiation of UI support for Windows or Linux platform

A. Singleton
B. Abstract Factory
C. Builder
D. Prototype
E. Object Pool

1	2	3	4	5
D	E	A	C	B

Match the following examples with the most appropriate design pattern!

1. Tree to represent mathematical expressions
2. Lazy initialization
3. Reader wrapping an InputStream in Java
4. Public API of a PDF generator library
5. Encrypter OutputStream wrapping a network OutputStream in Java

A. Proxy
B. Decorator
C. Composite
D. Adapter
E. Facade

1	2	3	4	5

Match the following examples with the most appropriate design pattern!

1. Tree to represent mathematical expressions
2. Lazy initialization
3. Reader wrapping an InputStream in Java
4. Public API of a PDF generator library
5. Encrypter OutputStream wrapping a network OutputStream in Java

A. Proxy
B. Decorator
C. Composite
D. Adapter
E. Facade

1	2	3	4	5
C	A	D	E	B