

Instructions



How is it???

0 responses



Do you have problems with practice?

Agenda

- Recall what was learned
- Look at current topic practice
- Check current topic knowledge

Total Recall



Is shallow copy allowed for immutable object?

0 

yes

0 

sometimes

0 

no

Pros of Facade pattern

0 ✓

You can isolate your code from the complexity of a subsystem.

0 ✗

A facade can become a god object coupled to all classes of an app.

0 ✗

You can create platform-independent classes and apps.

Which of the following is true for facade pattern?

- a) It is a broker pattern that eases interaction between a client and a sub-system of suppliers by providing a simpler interface to the sub-system ✖
- b) Systems may have many classes with complex interfaces, but often clients need basic services that can be supplied through simple interfaces ✖
- c) Broker class, called a façade, provides basic, simplified services to clients by taking upon itself the job of dealing with a complex sub-system ✖
- d) All of the mentioned ✓

Benefits of Command pattern



Which of the following are the consequences for Command pattern?

- a) Event-driven design is facilitated
- b) Making event-handling code easy to find, change, and reuse
- c) The invoker is loosely coupled to both the command and client classes, making them easy to change
- d) All of the mentioned

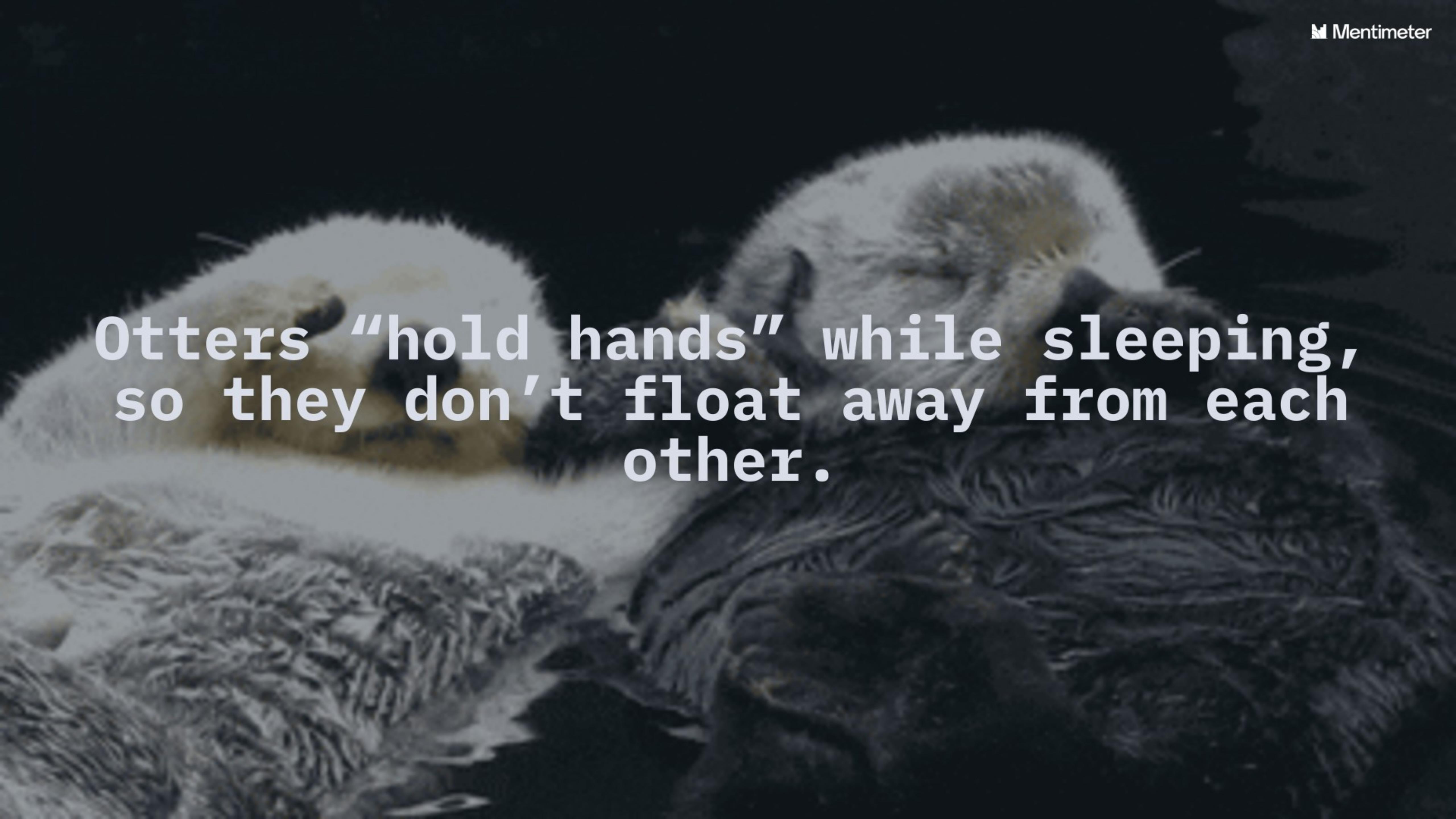




Discuss current topic

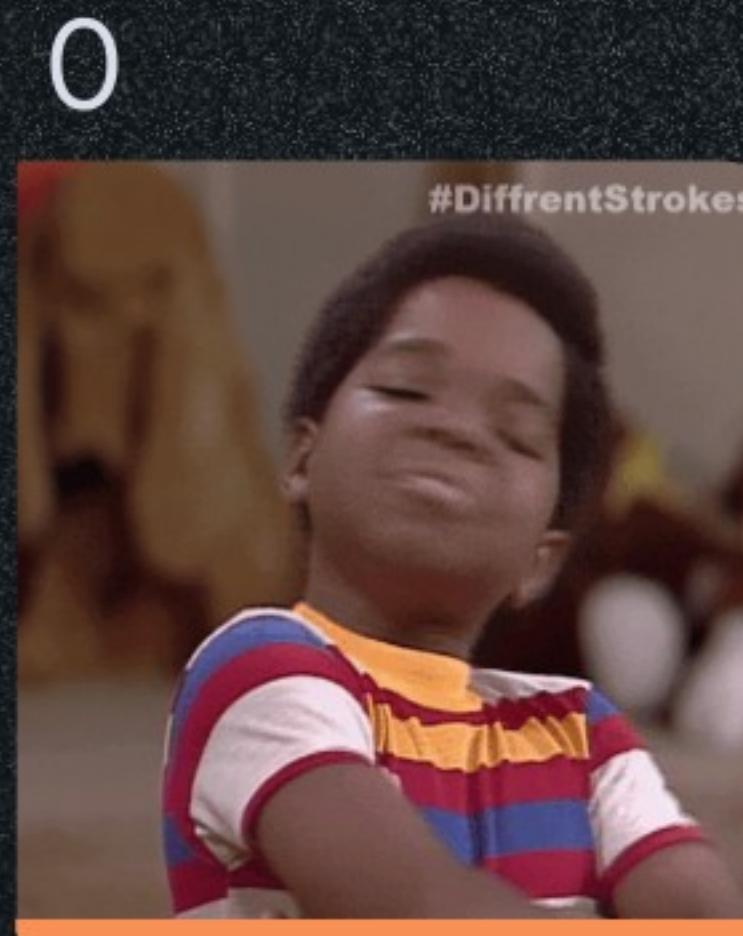


Monterey Bay
Aquarium

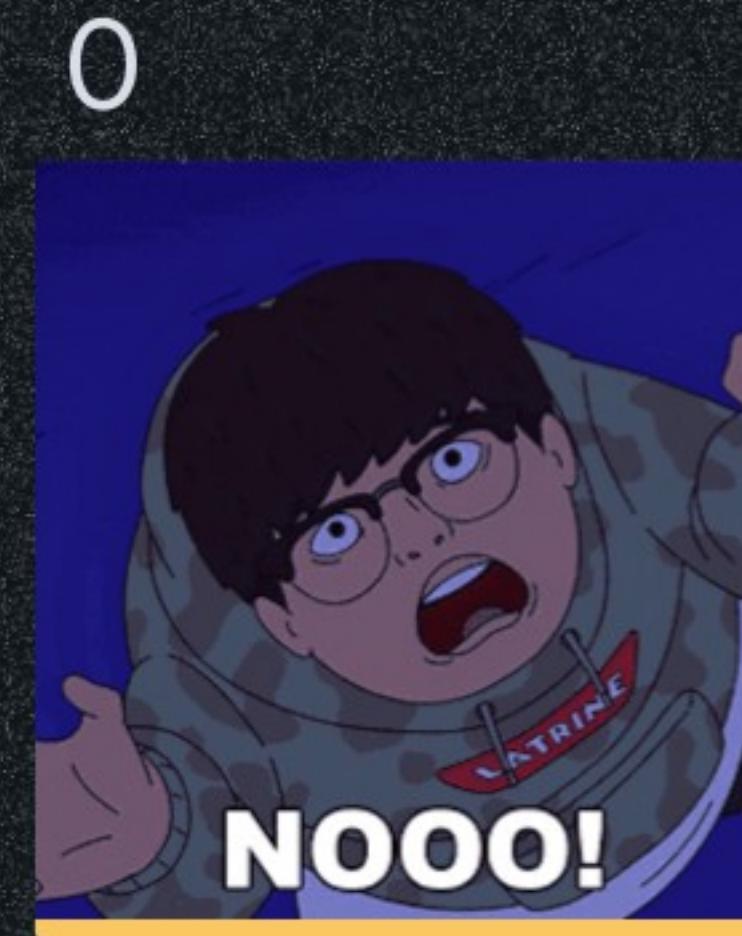


Otters “hold hands” while sleeping,
so they don’t float away from each
other.

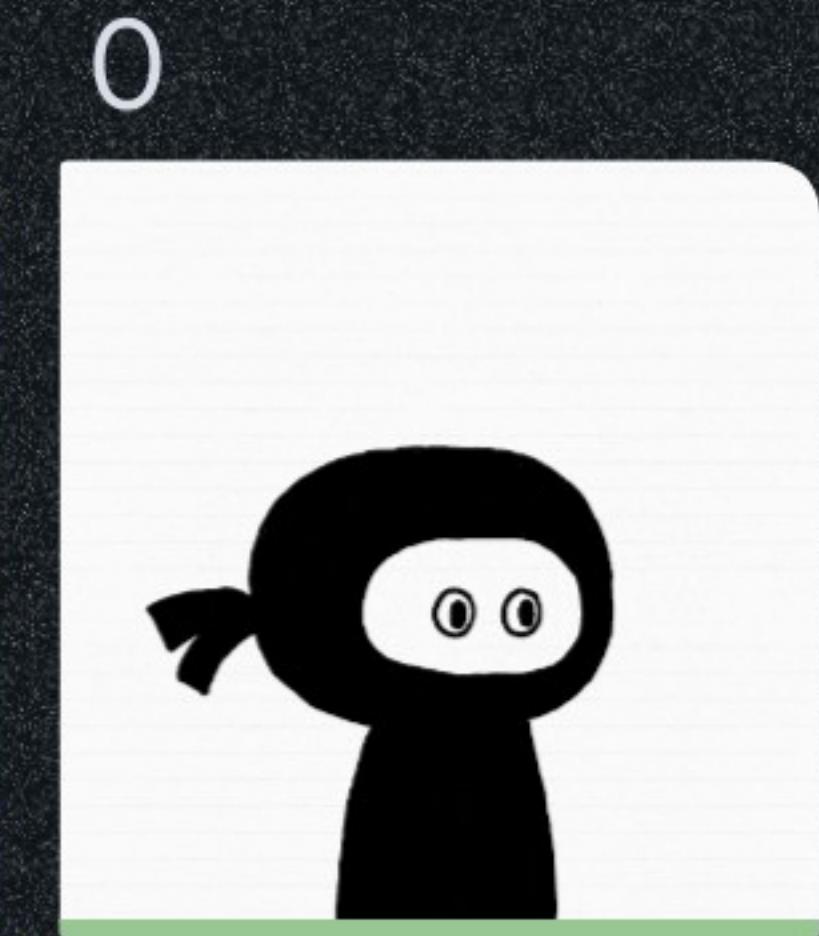
Have you looked at todays' materials?



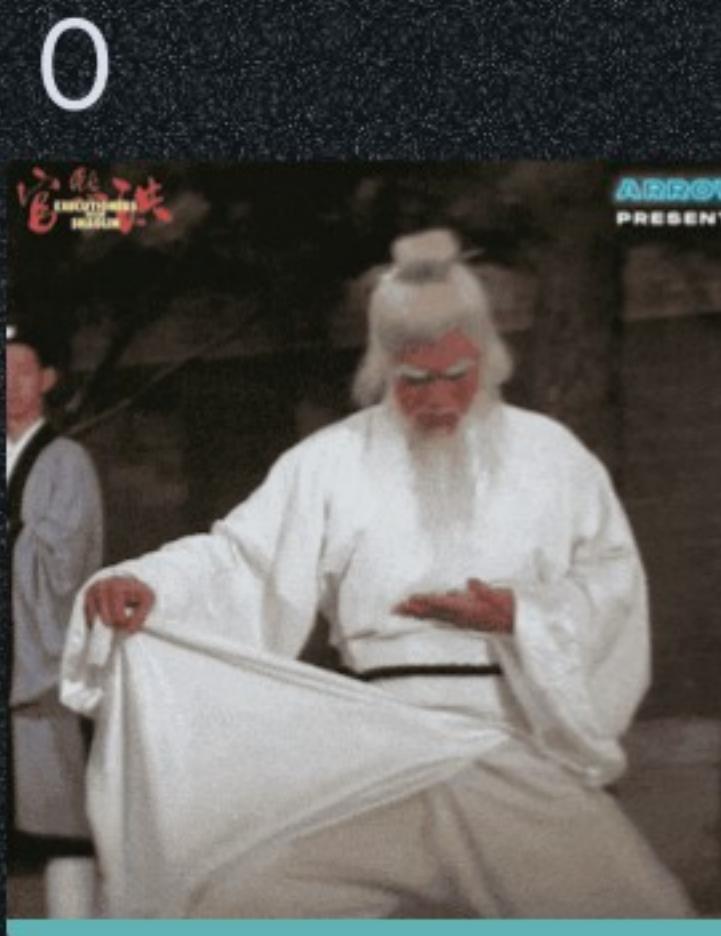
yes



no



partially



i'm the author

Convinient Patterns: Observer

What is the purpose of the Observer pattern in software engineering?

- Enforces strict coupling between objects ✖
- Limits the number of objects in a system ✖
- Specifies a dependency relationship between objects ✓
- Controls the flow of program execution ✖

Which objects are involved in the Observer pattern?

- Controller and View ✗
- Subject and Observer ✓
- Client and Server ✗
- Model and View ✗

What does the subject do in the Observer pattern?

- Controls the behavior of the observer ✖
- Updates its state based on observers ✖
- Maintains a list of observers and notifies them ✓
- Executes actions independently of observers ✖

Which method is called by the subject to notify observers of a state change?

- attach(observer) ✗
- detach(observer) ✗
- update() ✗
- notify() ✓

In the Observer pattern, what does the observer implement to receive state changes?

- attach(observer) ✗
- detach(observer) ✗
- notify() ✗
- update() ✓

What could be an example scenario where the Observer pattern can be applied?

- Weather station displaying weather conditions
- Network routing algorithm
- Database management system
- User authentication system



How does the WeatherData class notify observers of a state change in the Observer pattern?

- By triggering an event in the observer objects ✖
- By directly updating observer objects ✖
- By calling `notifyObservers` method and iterating through observers ✓
- By sending notifications via a separate channel ✖



10 mins break

Convenient Patterns: Builder

What is the purpose of the Builder Pattern?

- To convert the interface of a class into another interface ✗
- To deposit a specific amount into an account ✗
- To allow the construction of complex objects step by step and produce different types and representations of an object using same construction code ✓
- To execute and undo a deposit operation ✗

Which of the following is NOT a benefit of using the Builder design pattern?

- a) Encapsulation of object construction ✗
- b) Immutable objects creation ✗
- c) Reduced parameters in constructor ✗
- d) Increased complexity of code ✓



In the Builder design pattern, which component defines the process for creating the parts of a complex object?

0

a) Concrete Builder

0

b) Director

0

c) Client

0

d) Product

What role does the "Director" play in the Builder design pattern?

- a) It constructs an object using the Builder interface
- b) It is responsible for defining the individual components of the object
- c) It is the actual object being constructed
- d) It provides an interface for creating parts of a product



How does the Builder design pattern achieve better maintainability compared to having numerous constructors with varying parameters?

- a) By allowing the step-by-step construction of complex objects
- b) By providing a common interface for object construction
- c) By consolidating all variations of object construction into a single builder
- d) By incorporating all parameters into a single constructor call

Q&A part

0 questions
0 upvotes