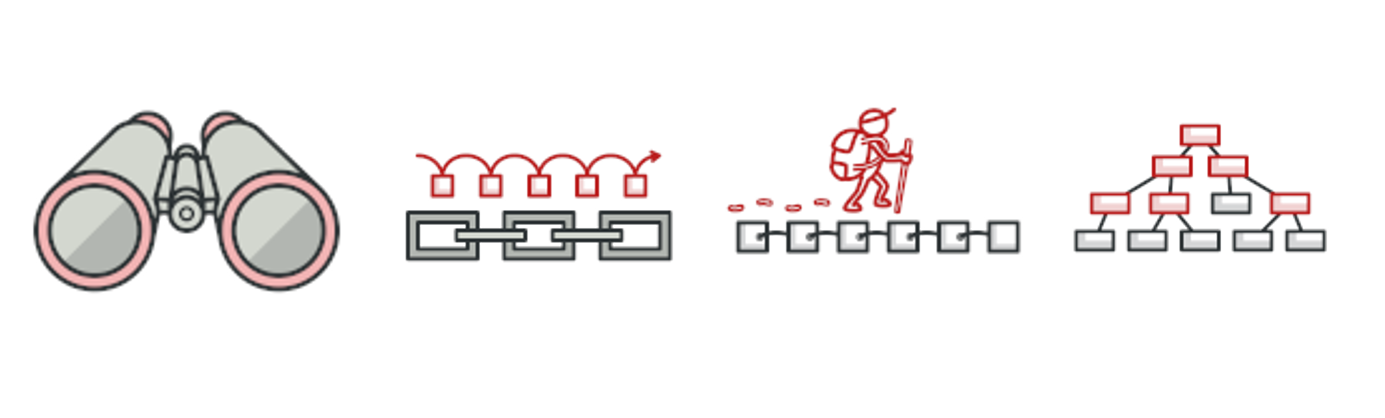
1. Are you able to recognize design patterns based on their logos? (The list is unsorted.)



* Chain of Responsibility, Composite, Observer, Bridge
* Visitor, Composite, Observer, Iterator
* **Chain of Responsibility, Composite, Observer, Iterator**
* Chain of Responsibility, Template Method, Observer, Iterator

1. Suggest the most appropriate design pattern for the following real world scenario.

*Let’s say you have some UI that edits some data, and you want the UI to react when the data is updated. Which pattern is most appropriate?*

* Command
* **Observer**
* Strategy
* Iterator

1. The following high-level GoF description defines the \_\_\_\_\_\_\_\_\_\_\_\_ design pattern

The intention of the pattern is to avoid coupling the sender of a request to its receiver by giving more than one object a chance to handle the request. Client initiates a request and passes it to a handler. Each handler performs its processing logic, and then potentially passes the processing request onto the next link (i.e. handler).

* **Chain of Responsibility**
* Observer
* Visitor
* Command

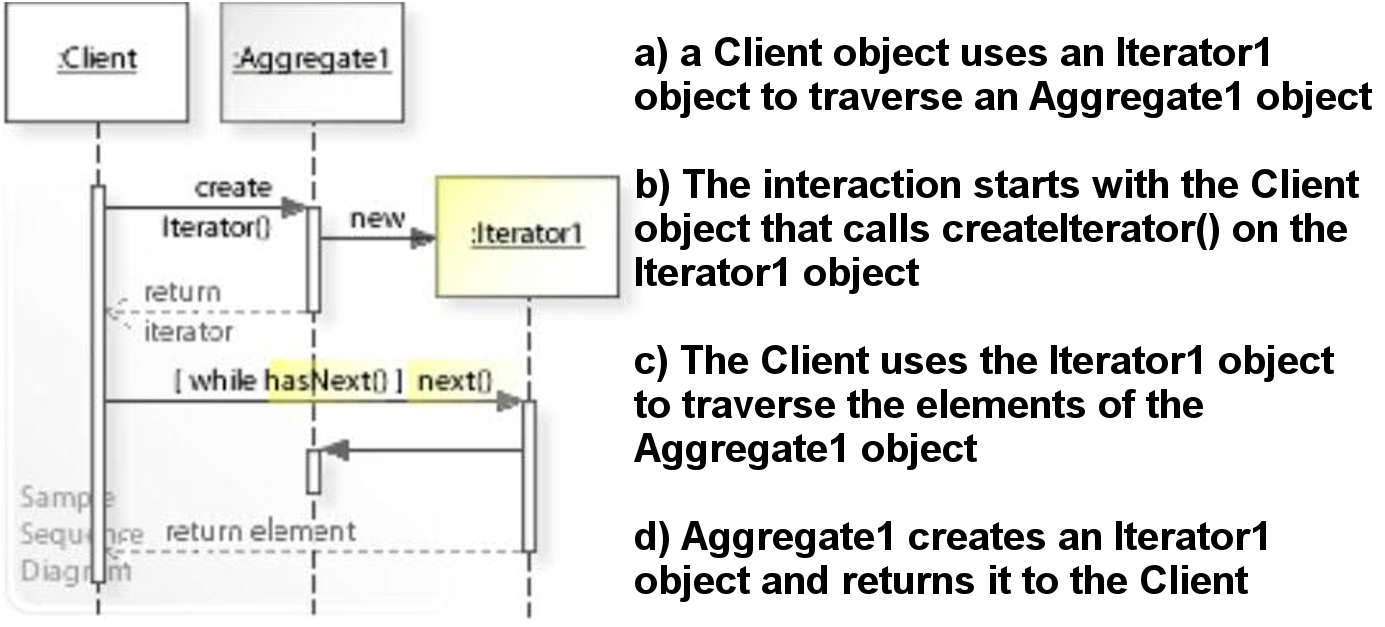
1. Client needs to access and traverse aggregate without knowing it's underlying data structure ==

* Visitor
* Composition
* Strategy
* **Iterator**

1. In which group of design patterns you could find Iterator ?

* Creational
* **Behavioral**
* Structural
* Concurency

1. Considering the depicted UML sequence diagram, which of the following statements are NOT true?



* A
* **B**
* C
* D