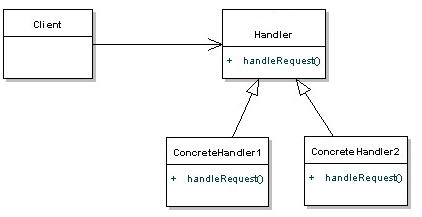
1. Are usually objects in Object pool mutable?

* **Yes**
* No
* Yes, if it is a pool of singletons
* No, if it is a pool of singletons

1. Which is **NOT** true for Facade pattern?

* Delegates activities to other portions of the sub-system
* The details of delegations by facade depend on the sub-system in question
* **It is behavioral design pattern**
* It eases interaction between a client and a sub-systems

1. What is true for the following Chain of Responsibility pattern UML class diagram?

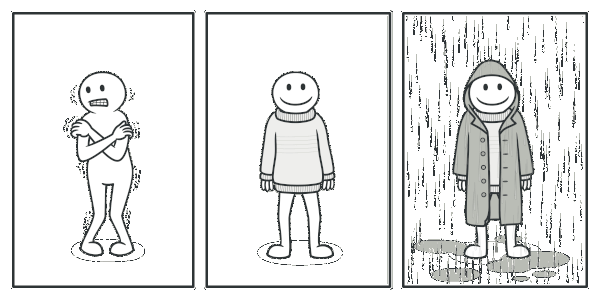


* It is a valid UML class diagram
* The *"handleRequest"* method should take the next handler as a parameter
* **There is a missing reference to the next handler**
* The postfix "*Handler"* should be renamed to "*Element"*

1. Which problem is **NOT** solved by Proxy design pattern?

* How can the access to an object be controlled
* How can additional functionality be provided when accessing an object
* How can clients treat a remote object as if it were a local object
* **How can an object be accessed that has not the needed interface**

1. Wearing clothes is a real-world analogy of ...? (All of garments “extend” your basic behavior but aren’t part of you.)

****

* Builder
* **Decorator**
* Composite
* Proxy

1. This pattern can be easily recognized by a *clone()* or *copy()* methods or a constructor accepting the same class.



* Memento
* Builder
* **Prototype**
* Factory Method