1. With this reference objects are reclaimed by GC only when JVM runs out of memory.

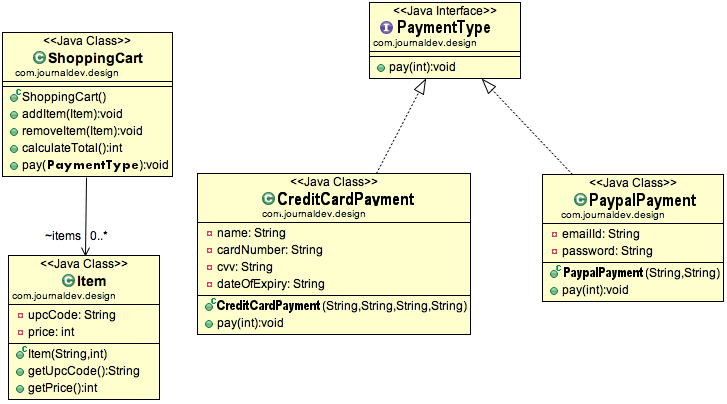
* Strong Reference
* **Soft Reference**
* Weak Reference
* Phantom Reference

1. Which Design Pattern should you use?

*When an object should be able to notify other objects without making assumptions about who these objects are. In other words, you don't want these objects tightly coupled.*

* Command
* Visitor
* **Observer**
* Strategy

1. What Design Pattern has been used in the following UML Diagram?

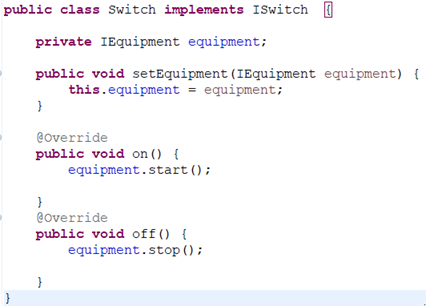


* **Strategy**
* Abstract Factory
* Composite
* Visitor

1. Bridge is a \_\_\_\_\_\_\_\_\_\_ Design pattern

* Creational
* **Structural**
* Enterprise
* Behavioral

1. [Bridge] Following class is an example of:



* Abstraction
* Implementor
* Concrete Implementor
* **Refined Abstraction**

1. Which of following statements is **NOT** true for Bridge design pattern ?

* **Makes things work after they're designed**
* Used mainly for implementing platform independence feature
* Puts the implementation in its own inheritance hierarchy
* Abstraction/implementor is an interface or abstract class