Inner Classes Lab 1:

Objectives

- After this lab students should be able to implement interfaces in inner classes.

Unit Tests

You should write unit tests as you develop your program. Remember to use the tests to guide your development process, and be sure to test both that your classes do what they should do, and that they don't do what they should not do. Some examples of what you should test include:

- Getting connections from the ConnectionManager
- Requesting too many connections
- Getting field values from a requested connection
- Getting (invalid) field values from a closed connection

Instructions

Part I: Return of the Connection Manager

Define a `Connection` interface with accessors for IP, Port, and Protocol, as well as a `connect()` method that returns a String.

Create a `ConnectionManager` class with a private nested `ManagedConnection` class that implements `Connection`.

`ConnectionManager` should have methods to get connection objects based on IP and protocol (You must support at least three protocols from this list) or custom-configured connections based on IP and port number with optional protocol (the default protocol should be HTTP)

The ConnectionManager can only handle so many connections (You can set this limit in the constructor). ConnectionManager should keep track of how many connections it has initialized; if it has reached its limit and another request for a connection comes in, ConnectionManager should return `null` instead of a Connection object.

Part II: Closing Connections

When you're done with a connection, you really should clean up after yourself. Add a `close()` method to the `Connection` interface and to your nested `ManagedConnection` class. A `ManagedConnection` object should tell its ConnectionManager that it has been closed, so that the ConnectionManager knows that there is one more connection available on the system.

A Connection should not provide its IP or Port numbers once it has been closed. Add logic to your implementation so that the accessors for these values return invalid values (i.e.: error codes) if the connection has been closed. The `connect()` method should also return an error message if it is called after a connection has been closed.

Additional Practice

If you are having trouble with this lab:

- try Exercise 1 from the Inner Classes chapter in Thinking In Java.
- Read through the Oracle Nested Class Tutorial