Decorator Pattern

In this lab, you will use the Decorator Pattern to wrap another object.

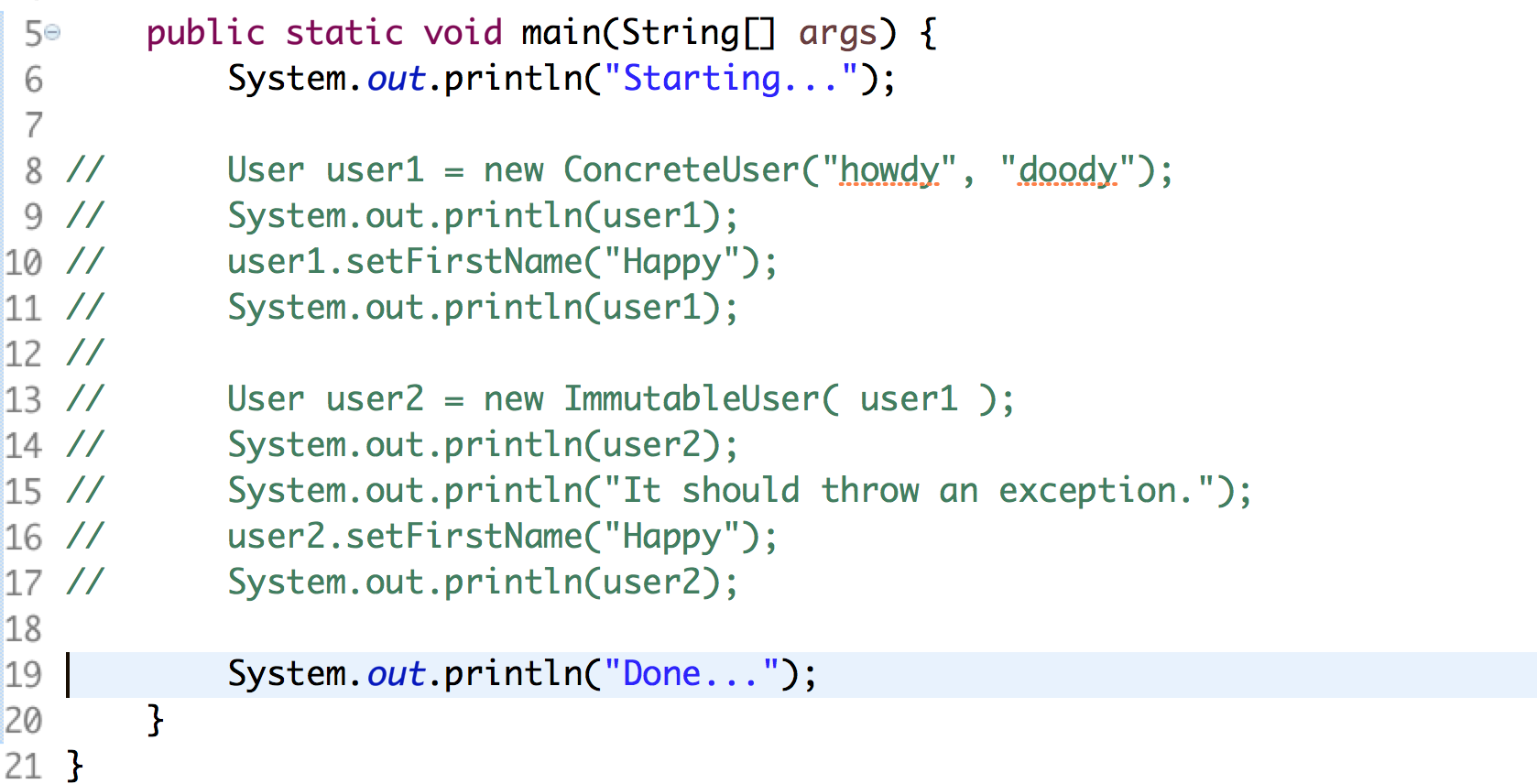
# Objectives

In this lab, you will

* Create an ImmutableUser which decorates User objects.
* The Decorator’s constructor has one parameter, a User object
* The Decorator delegates all accessors (getters) to the User object
* The Decorator intercepts all modifiers (setters) and throws an Exception
* Run the app

# Exercise

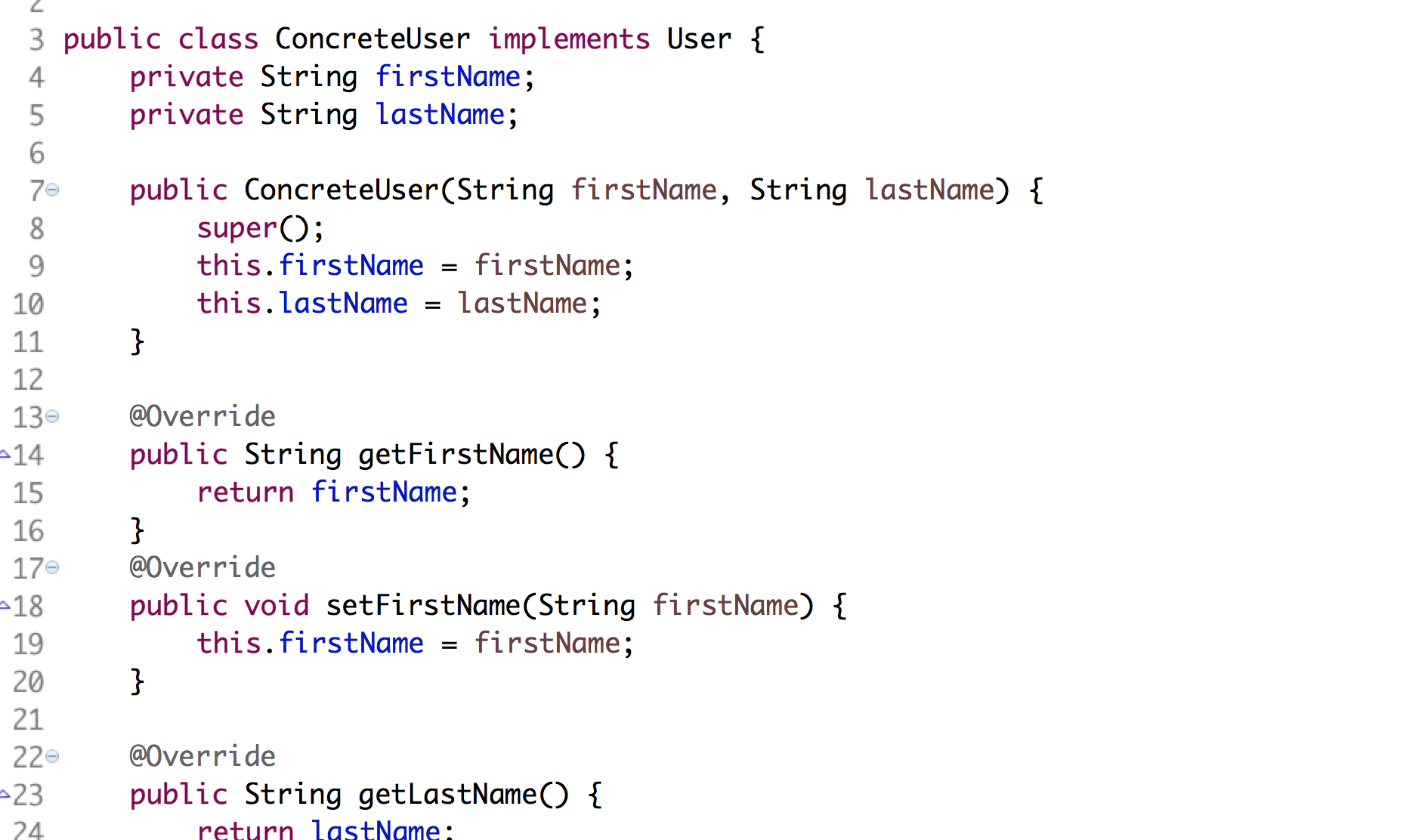
1. The Decorator pattern represents yet another wrapper around an object. Like a Proxy, the Decorator provides the same interface. However, the Decorator’s constructor has a single parameter, the object it is wrapping. The Decorator intercepts all the methods of the original object. It either delegates the methods to the original or modifies the behavior of the original in specific methods.
2. In this example, we provide an ImmutableUser that disallows all the mutator methods of the wrapped User object.
3. In Eclipse, in the exercises project, open the package com.paypal.patterns.Decorator to view the project files.
4. Examine Tester.java shown below:



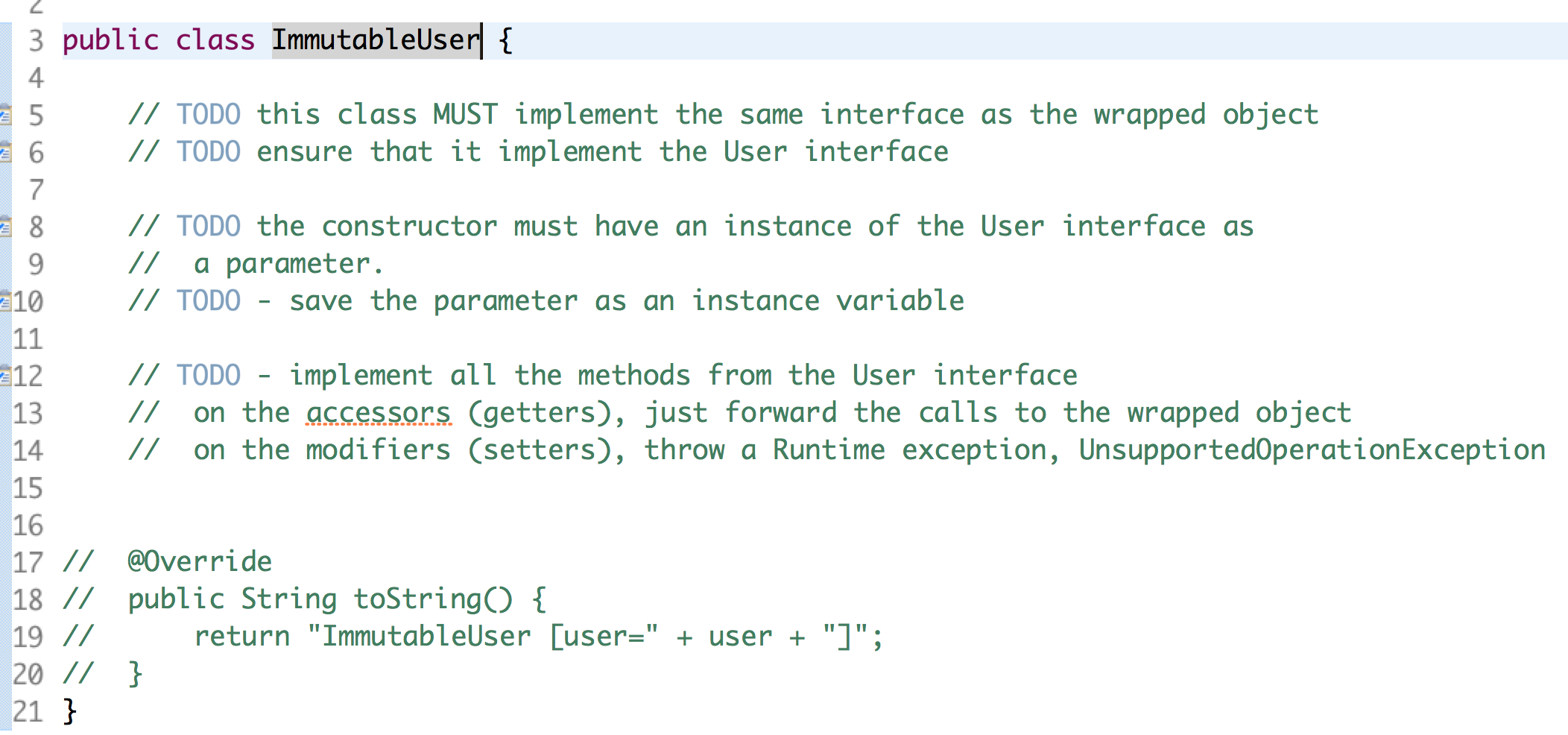
1. In line 8, we create a ConcreteUser. Then, in line 13, we create an ImmutableUser using the ConcreteUser we created above.
2. In this example, both the ConcreteUser and the ImmutableUser implement the User interface shown below:



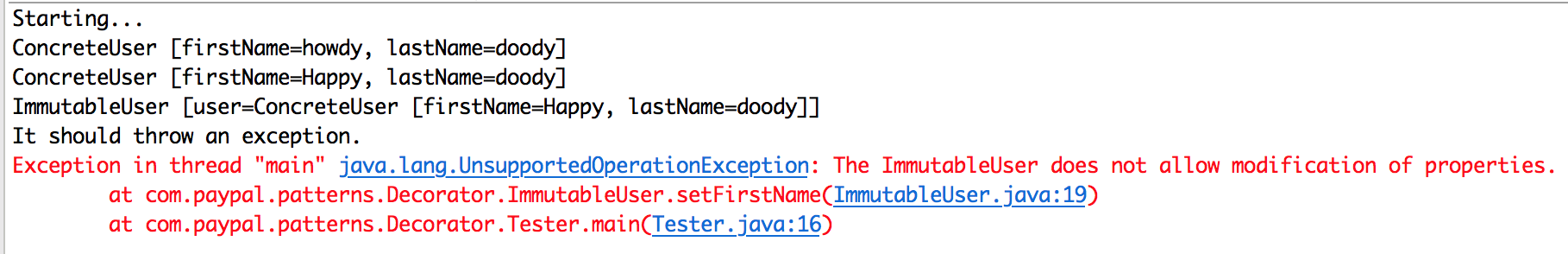
1. The ConcreteUser is a simple Java bean around the firstName and lastName strings as shown below:



1. On line 3 above, notice the ConcreteUser implements the User interface. The Decorator, ImmutableUser, must also implement the same interface.
2. Edit ImmutableUser.java shown below:



1. Note: the Decorator MUST implement the same interface as the wrapped object and the constructor MUST have the wrapped object as its parameter.
2. After fixing all the TODOs, uncomment the toString() method.
3. Uncomment the lines in Tester.java and execute the file as a Java application. You should see:



1. In the above, the Exception occurs because the client attempted to modify the immutable user.

Congratulations. You have completed this lab.