INSTRUCTIONS

First, add to your Stock class from Lab11 two new methods named `average()` and `last()`.

Practice typing Javadoc comments:

```java
/**
 * Returns the average of all the prices stored in this Stock.
 * @return the average price.
 */
public double average()
{ /* You code! */ }

/**
 * @return the last price stored in this Stock.
 */
public double last()
{ /* You code! */ }
```

(Question for the final exam: Why is the name of the method, and comments like "Returns the average of all the prices stored in this Stock" redundant in Javadoc comments? Run Javadoc and figure out why by observing the documentation it generates!)

Second, make a new class named Portfolio, a container for many Stock objects. Program the Portfolio class so one Portfolio object can acquire Stocks, each with a record of its prices. For lab simplicity, make it work with only 3 Stocks.

How? To start, put in Portfolio the fields and the constructor:

```java
public class Portfolio
{
    //...
    private Stock[] arrayOfStocks;
    private final int nStocks = 3;
    public Portfolio()
    {
        arrayOfStocks = new Stock[ nStocks ];
    }
    // ...
```
Now, enable it to acquire stocks:

```java
public void acquire()
{
    for( int i = 0; i < nStocks; i++ )
    {
        arrayOfStocks[i] = new Stock();
        //Put the code to make the Stock referred to to arrayOfStocks[i]
        //read its prices into its array, by calling methods you wrote in
        //Lab 11. Finish those methods if necessary.
    }
}
```

Enable a Portfolio to print information about all its Stocks:

```java
public void printInfo()
{
    //Loop head HERE,
    //declaring int i and iterating with i from 0 to nStocks-1
    {
        //This is the loop body. Make it print info about one Stock.
        //print what’s returned by arrayofStocks[i].average()
        //etc., also for the last price.
    }
}
```

Finally, program Portfolio's `main` method so an investor or regulator (a person) can import stocks (actually, information about stocks!) into one `Portfolio`, and see the average price and last price recorded for each stock.

**HOW?**

1. Program main to construct one `Portfolio` (using `new` of course). **One line of code!**
2. Program `main` to make that `Portfolio` acquire stocks. **One line of code!**
3. Program main to make that `Portfolio` print the information about all its `Stocks`. **One line of code!**
4. TEST: Get some historical prices from say `yahoo.finance.com` Messing with the format will be necessary, or just make two new phony, different versions of our test files of Google's prices.

**PREPARATION, STUDYING AND REVIEW:**

Read about making your own classes and other stuff for this lab, and the project, in G&E chapter 11.

**Followup:** NONE for credit!! Use the practice from this lab to help you with Project05! You can discuss and work on Lab code together, but Project code must be written BY YOURSELF, WITH NO copying of code (except from the book and lecture materials.)

**PS:** You can combine work from this lab with Lab11 if you'd like and haven't finished the Lab11 followup.

(Thanks to Jeremy and Lindsay for feedback leading to improvements here.)