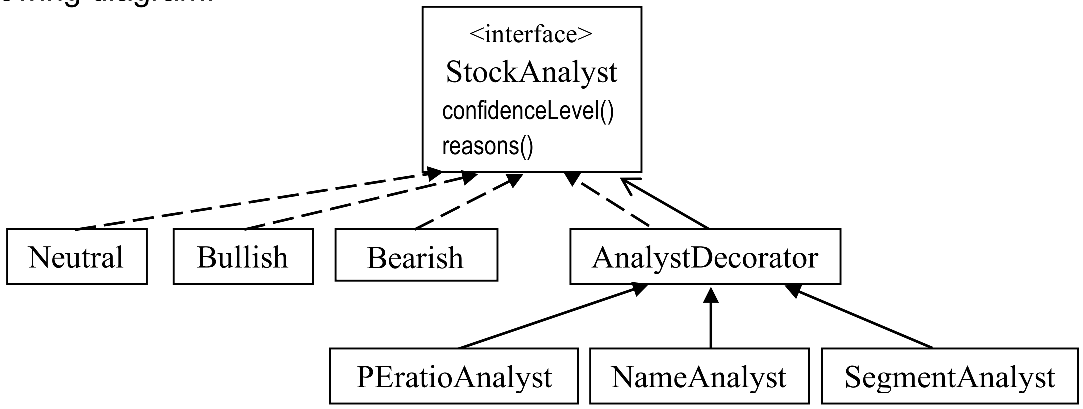
CSCI 3353 Object Oriented Design
Homework Assignment 8
Due Wednesday, April 11

Your task is to write a flexible system for analyzing stocks. The system contains various classes, called *stock analysts*, whose job is to make predications based on various aspects of a company. For example, there might be an analyst that makes predictions based on the company's earnings history, one that makes predications based on the price-earnings ratio, one that understands the broad trends in different market segments (e.g. "auto stocks are bad; technology stocks are good"), and so on.

Each analyst will implement the *StockAnalyst* interface. This interface has two methods:

- the method *confidenceLevel*, which returns a value between 0.0 and 1.0 that denotes the predicted probability that the company's stock will rise in value;
- the method *reasons*, which returns a string explaining the reasons for this value.

The stock analyst classes are organized according to the decorator pattern, as in the following diagram:



The base classes are *Neutral*, *Bullish*, and *Bearish*. Their confidence levels are based on the market as a whole, and apply equally to all companies. Assume that *Neutral* always gives a confidence level of 50%, *Bullish* 70%, and *Bearish* 30%.

In this assignment, you will also write three decorator classes. The *PERatioAnalyst* class calculates the company's Price-Earnings ratio by dividing the current price by current earnings. A PE ratio of 12 is considered to be average. A smaller ratio indicates an underpriced stock, and therefore a higher confidence that it will increase in price; and a higher ratio indicates an overpriced stock and a consequent lower level of confidence. Let's say that the analyst uses the following formula to generate its confidence level:

```
confidence = 1 - (PEratio / 24)
```

The *SegmentAnalyst* class has a preset confidence level for certain market segments. Let's assume that auto stocks have a confidence of 20%, and technology stocks have a confidence of 80%. For other stocks, the analyst has no opinion.

The *NameAnalyst* class bases its confidence level on the company's name. Let's assume that the analyst has 100% confidence in companies that begin with the letter "A", and has no opinion about other companies.

If a decorator analyst needs more information about a company than was given, then the analyst will have no opinion about the company.

The individual confidence levels of each analyst must be combined into a single overall confidence level. The *confidenceLevel* method for an analyst denotes this overall level. In particular, the overall confidence level of a decorator analyst is the average of its own confidence and the overall confidence of the analyst that it decorates. If the analyst has no opinion, then its overall confidence level is the same as the analyst it decorates.

Note that the overall confidence of a chain of decorators will differ depending on the order that the decorators appear in the chain. In particular, the opinions of the outer decorators affect the overall confidence more than the inner ones. Consider this a feature, not a bug.

The *reasons* method operates similarly to *confidenceLevel*. If an analyst has an opinion about a company, then it also has a reason for that confidence. For example, the reason given by the segment analyst for its low confidence in auto stocks might have the corresponding reason "Auto stocks are a bad risk.". The overall reasons for a chain of analysts is the concatenation of their individual reasons.

The information about each company is stored in a text file. The file can contain an arbitrary number of lines, each of which denotes information about the company. The first word on each line is a keyword, such as "name", "marketsegment", "earnings", "shareprice", etc. The remainder of the line contains data about that keyword. An example file is *apple.txt*, which you can download.

You should write a class *StockInfo*, whose constructor reads a specified file, extracts its information, and saves it in a data structure of your choice. The class should have methods that the analysts can call to retrieve the data they need.

I have written a test file named *HW8Test.java*, which you should download. The code asks the user to interactively specify different decorator chains, and then prints the confidence level and reasoning corresponding to that chain. Here is some example output from my program:

```
File name: apple.txt
General outlook (0=neutral, 1=bearish, 2=bullish): 2
Analyst: (0=done, 1=segment, 2=PEratio, 3=name): 3
Analyst: (0=done, 1=segment, 2=PEratio, 3=name): 2
Analyst: (0=done, 1=segment, 2=PEratio, 3=name): 1
Analyst: (0=done, 1=segment, 2=PEratio, 3=name): 0

Confidence level is 0.7138600113927656
Reasoning: I think the market is going up.
     I love companies that begin with 'A'.
     The PE ratio is 14.269438906294504, which is bad.
     Tech stocks are a good risk.
```

Here is another:

```
File name: apple.txt
General outlook (0=neutral, 1=bearish, 2=bullish): 0
Analyst: (0=done, 1=segment, 2=PEratio, 3=name): 0

Confidence level is 0.5
Reasoning: The market's future is unclear.
```

When you are done, create a zip file containing the ten files *StockAnalyst.java*, *Bearish.java*, *Bullish.java*, *Neutral.java*, *AnalystDecorator.java*, *PEratioAnalyst.java*, *SegmentAnalyst.java*, *NameAnalyst.java*, *StockInfo.java*, and *HW8Test.java*, and submit it to Canvas. Make sure that all of your java files are in the package *hw8*.