Exercise 1: Class Currency Converter

You just returned from a trip to Europe, and you want to convert your leftover Euros back to US dollars. In this exercise, we will write a currency converter.

Part a: Go to a computer and look up the current conversion rate. Try http://www.x-rates.com/calculator.html

Part b: In groups of 2 or 3, generate (by hand) some test data. Start with the following

I. 100 Euro = __________ dollars
II. 37 Euro = __________ dollars
III. $100 = __________ Euro
IV. $500 = __________ Euro

Part c: Write a Java program to do the conversion. The main method should

I. Ask the user to enter the amount of Euro they have
II. Display: Amount user has in both currency types
   Example output: Your 120 dollars is equal to 95.27 Euro.

Part d: Test your program using the results from Part b.

Part e: Currency converter objects. Being a world traveler, Euro is not the only currency of interest. Generalize your program able to create currency-converter objects that can convert between two currencies. To generalize the program from Part c, what attributes and methods does the currency-convert object need?

Part f: To test your class, create three currency-converter objects (for Euro to Dollars, Yen to Dollars, and Dollars to Euro), then write a test driver (main method) that allows a user to select which of the three objects is used before prompting for an amount and doing the conversion (using methods of the converter class).
Exercise 2: PigLatin

The rules for translating an English word to Pig-Latin are as follows:

- If the word starts with a vowel, it is unchanged.
- Otherwise, all of the consonants at the beginning of the word (up to the first vowel) are moved to the end of the word, preceded by a dash (for readability), and followed by "ay".

For example, the sentence "I love Baltimore in the springtime" would be translated to "I ove-lay altimore-Bay in e-thay ingtime-spray".

For this exercise, you will complete the PigLatin class from the handout, which will allow users of the class to translate an English word to Pig-Latin. First, make sure you understand the translation rules by translating the following phrases into Pig Latin:

Hello world
I love computers
the car says ooga ooga

Part a: Take a look at the incomplete PigLatin class. Make sure you understand what each method is supposed to do, and discuss the reasons for making some of the methods private and others public.

Part b: Write the isVowel method, which returns true when its character argument is a vowel and then the isVowelAt method, which returns true if the letter at the given position of the englishWord data member is a vowel, and otherwise returns false. To get the letter at the given position, use the charAt method of the String class:

charAt( int index )     returns the character at the specified index

Try writing the isVowel method using an if statement and also using a switch statement.

To test your code, add some statements to the PigLatin class's main method that create a Word and then call its isVowelAt method (passing different values for the position) and print the returned value. Then compile and run the program.

Part c: Now write the firstVowelPos method, which returns the position of the first vowel in the englishWord data member of the PigLatin class, or returns -1 if there is no vowel in the word. Here are some examples:

<table>
<thead>
<tr>
<th>englishWord</th>
<th>Result of calling firstVowelPos</th>
</tr>
</thead>
<tbody>
<tr>
<td>hello</td>
<td>1</td>
</tr>
<tr>
<td>ice</td>
<td>0</td>
</tr>
<tr>
<td>spring</td>
<td>3</td>
</tr>
<tr>
<td>qyzzx</td>
<td>-1</td>
</tr>
</tbody>
</table>

Test your firstVowelPos method by replacing the calls to isVowelAt in the main method of the PigLatin class with a call to firstVowelPos.

Part d: Now describe in English how the translate method of the PigLatin class should work (using the rules for translating from English to Pig-Latin given above). Then write the actual code, remove the code in the main method that prints the first vowel position (leaving the call to the translate method), compile your code and run it!

Part e: A word object is a String. Modify class Word to be a subclass of class String.