EXERCISE 2

I. Pig Latin

Pig Latin is childish encoding of English that adheres to the following rules:

1. The vowels are 'a', 'e', 'i', 'o', 'u', as well as any 'y' that is not the first letter of a word. All other letters are consonants.
For example, 'yearly' has three vowels ('e', 'a', and the last 'y') and three consonants (the first 'y', 'r', and 'l').

2. If the English word begins with a vowel, append 'hay' to the end of the word to get the Pig Latin equivalent. For example, 'ask' becomes 'askhay', 'use' becomes 'usehay'.

3. If the English word starts with 'q', assume it is followed by 'u'; move 'qu' to the end of the word, and append 'ay'. Hence 'quiet' becomes 'ietquay', 'quay' becomes 'ayquay'.

4. If the English word begins with a consonant, move all the consonants up to the first vowel (if any) to the end and add 'ay'. For example, 'tomato' becomes 'omatotay', 'school' becomes 'oolschay', 'you' becomes 'ouyay', and 'ssssh' becomes 'sssshay'.

Your goal is to write a function pigify that take a single English word (e.g. a string with only letters and no spaces), and converts it into Pig Latin.

To aid with our Pig Latin conversion, we have provided a helper function first_vowel(w):

```
def first_vowel(w):
    """
    Returns: position of the first vowel; -1 if no vowels.

    Parameter w: the word to check
    Precondition: w is a nonempty string with only lowercase letters
    """
```

We hope that this helper function is correct. To verify this, write down at least 8 key test cases. We do not want you to write a test script or try to fix any bugs.

Input	Expected Output

II. The Function pigify

The function pigify(w) has a short-and-simple specification:

```
def pigify(w):
    """
    Returns: copy of w converted to Pig Latin

    See the lab instructions for the complete rules on Pig Latin.

    Parameter w: the word to change to Pig Latin
    Precondition: w is a nonempty string with only lowercase letters
    """
```

This specification assumes that you have read the definition of Pig Latin on the previous page. Implement this function in lab02.py.

When you are done, you will want to test your answer. Instead of creating a unit test, we only want you to write down a list of test cases to verify that your implementation is correct. Provide at least four tests.

Input	Expected Output