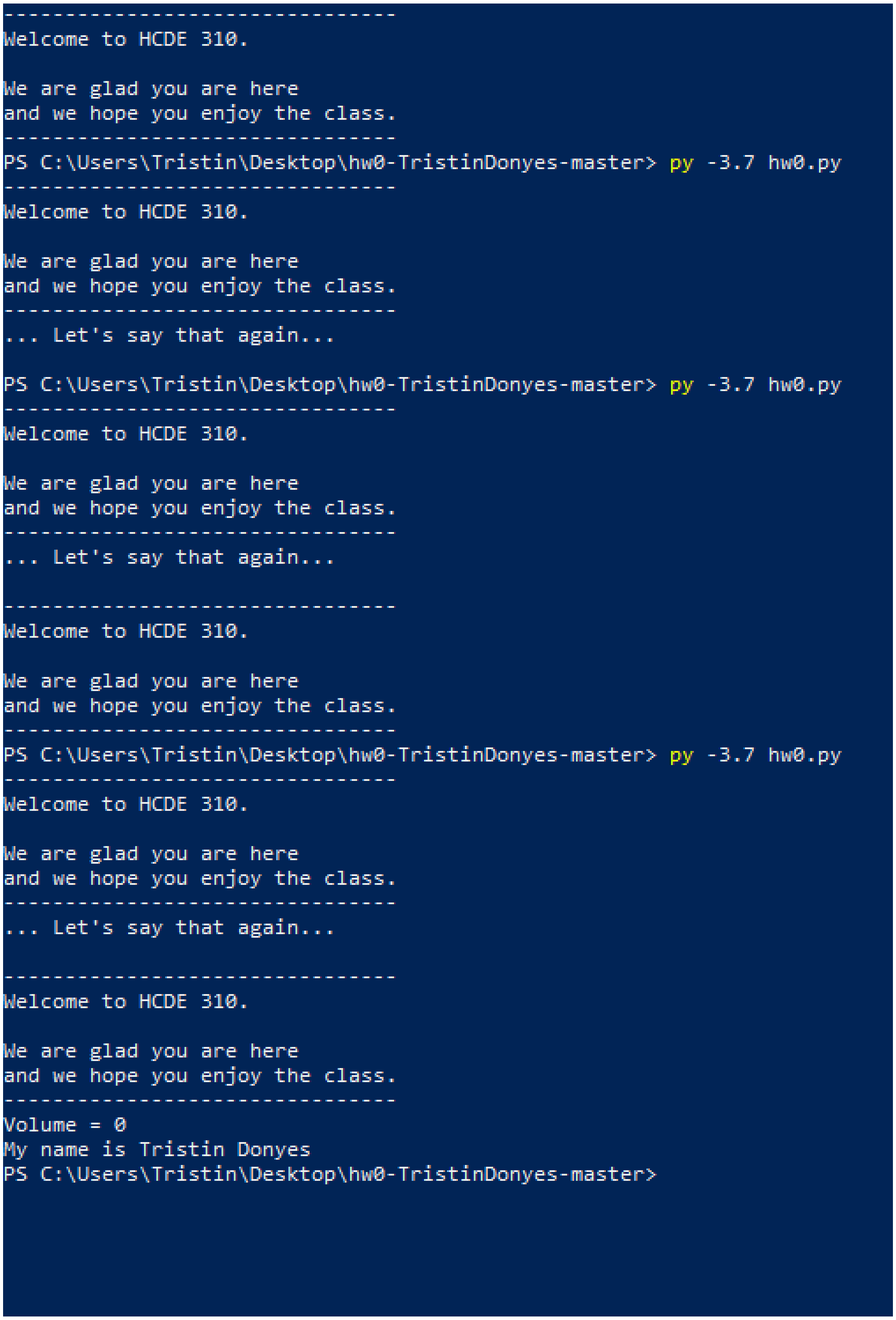

```
--------------------------------
Welcome to HCDE 310.

We are glad you are here
and we hope you enjoy the class.
--------------------------------
PS C:\Users\Tristin\Desktop\hw0-TristinDonyes-master> py -3.7 hw0.py
--------------------------------
Welcome to HCDE 310.

We are glad you are here
and we hope you enjoy the class.
--------------------------------
... Let's say that again...

PS C:\Users\Tristin\Desktop\hw0-TristinDonyes-master> py -3.7 hw0.py
--------------------------------
Welcome to HCDE 310.

We are glad you are here
and we hope you enjoy the class.
--------------------------------
... Let's say that again...

--------------------------------
Welcome to HCDE 310.

We are glad you are here
and we hope you enjoy the class.
--------------------------------
PS C:\Users\Tristin\Desktop\hw0-TristinDonyes-master> py -3.7 hw0.py
--------------------------------
Welcome to HCDE 310.

We are glad you are here
and we hope you enjoy the class.
--------------------------------
... Let's say that again...

--------------------------------
Welcome to HCDE 310.

We are glad you are here
and we hope you enjoy the class.
--------------------------------
Volume = 0
My name is Tristin Donyes
PS C:\Users\Tristin\Desktop\hw0-TristinDonyes-master>
```

For part 6 I forgot to change the volume components (length, height, width), hopefully that's okay

Part 7:

I will answer this question to the best of my ability because I do not quite understand if we are counting the total number of words in the document or if we are counting the different occurrences of different words throughout the document, but anyways here we go.

I would tell someone/a computer to scan the document for a character that isn't a space, and to then increase the count (starting at 0) by 1. While this is happening, there would be another value somewhere, probably oscillating between 1 and 0, that articulated whether the scanned character was part of a word or was a 'space' character. Thus, when the document was being scanned, and a character was detected (increasing the count by 1), this other value would increase from 0 to 1, indicating that we had started a word. While this second value is not 0, the other characters wouldn't increase the count anymore, since it is part of the same word. Then when the scan encounters a 'space' character, indicating the end of a word, the second value would be changed back to 0, indicating that the first method can commence again, and any character other than a space would indicate the start of another word.

This doesn't account for special cases yet, like periods, semicolons, or other weird symbols that might be counted as a word when they are not, but conceptually as a class we haven't defined what we are counting as a word or not, and therefore I cannot address special cases without proper instruction.

Part 8:

The set up of the software was relatively straight forward, the hw0 structure was easy to follow and the tasks were very simple. The only thing that I found annoying was pushing files from PowerShell to GitHub, which seems largely unnecessary. While I understand that conceptually this makes sense for group coding in a professional environment, this is a skill that will likely not be used by people in the hcde field, as well as there are plenty of other ways to accomplish said task, without using such an overly basic platform such as PowerShell/console. Again I do understand that this is being taught for conceptual understanding based on the goals of the curriculum, however, as someone who has programmed for several years in many different languages, I also understand that many people who are newer to this, some of my friends included, are asking tons of questions related to PowerShell specifically, when the core understanding that they are attempting to get out of this class is purely Python related, in the context of a Python IDE such as PyCharm. I'm just not convinced that this is the best way to further people's learning, as I see it as an additional stressor that sort of distracts from the basic ideas of Python and OO programing. Just some food for thought