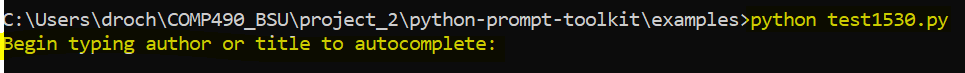
TEST 1: When the prompt library is imported and a prompt instance is created, the user can choose to set move\_back\_one\_space to True. If they do, then the cursor will move back one space after each and every autocomplete when the user presses tab (and if there is more than one autocomplete suggestion, enter). NOTE: An editable version of the prompt\_toolkit should be installed using the local version pulled from our github to be able to see this change. To install an editable version, clone the repository and run the command **pip install -e <DIRECTORY FILE PATH>**. If the editable version is installed and a sample program importing the prompt\_toolkit is run correctly from the command line, it should look like the following:

Sample program: test1530.py, move\_back\_one\_space = True

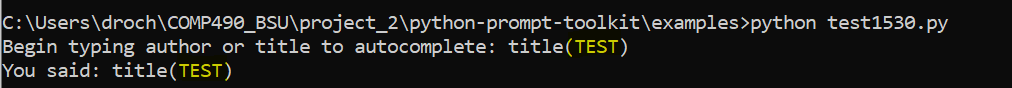
Text

Description automatically generated

Run the sample program from the command line to get the following:



Type “ti” and press tab to autocomplete. The cursor should now be inside of the parentheses. If you type “TEST” and press enter, the word “TEST” should be inside the parentheses. It should look something like this:

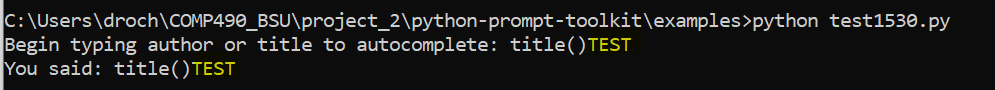


NOTE: If the program does not set move\_back\_one\_space to True, it is set to False by default in the constructor for the class in word\_completer.py. An example program would look like the following:

Text

Description automatically generated

If you then run the program and type “ti” and press tab, the cursor should be outside the parentheses at the end of the autocomplete. If you then type “TEST” and press enter, the word “TEST” will be outside of the parentheses. It should look like the following:



NOTE: There are instances in which you may want to move the cursor back for some words and not for others. In this current implementation, the cursor either never moves back after autocompleting or always moves back. This could be a problem if you are typing and don’t realize the cursor has moved back. So while this fix does solve this issue, I personally would add another feature request to make the cursor movement more customizable. But this edge case does exist.