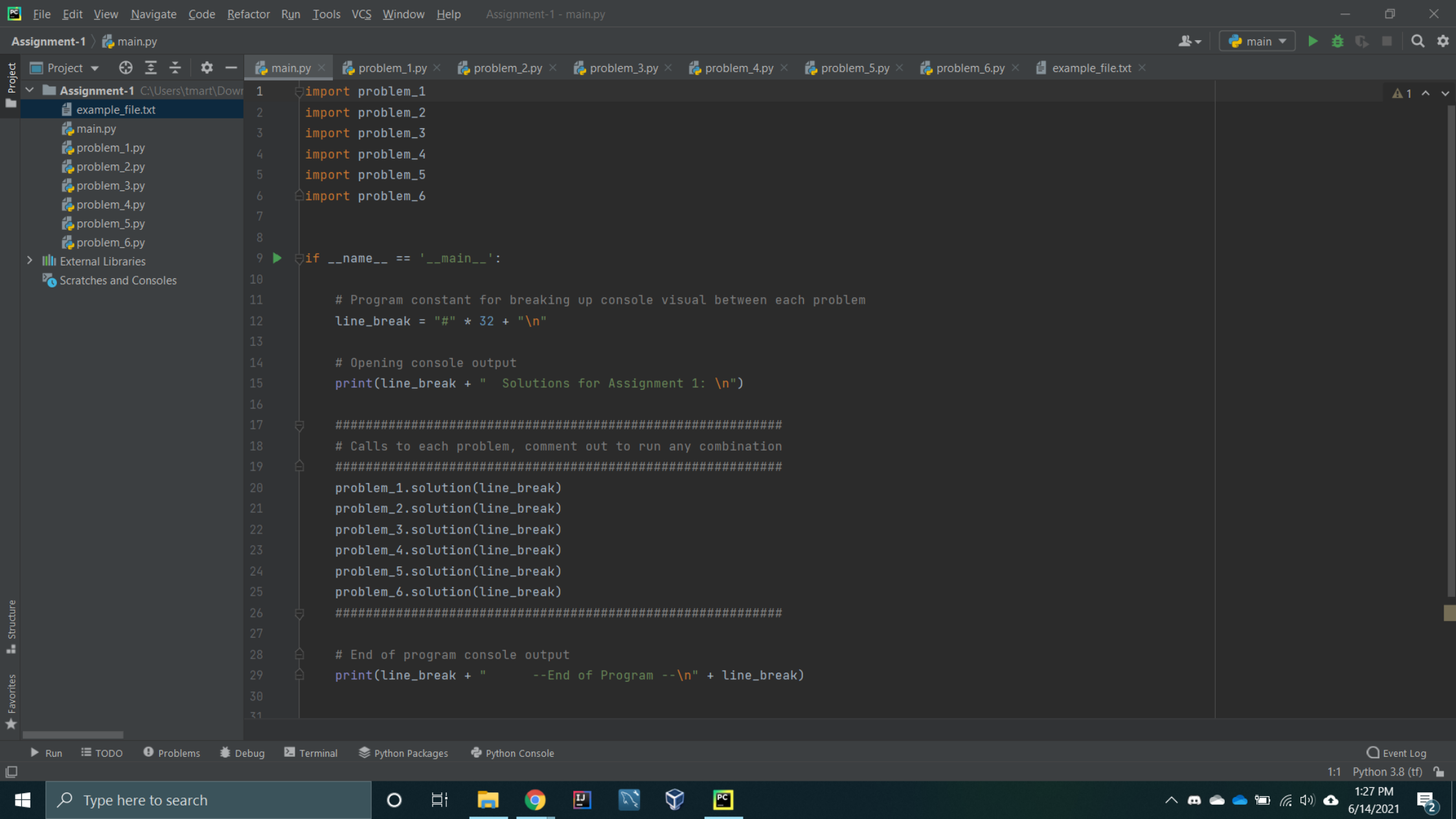
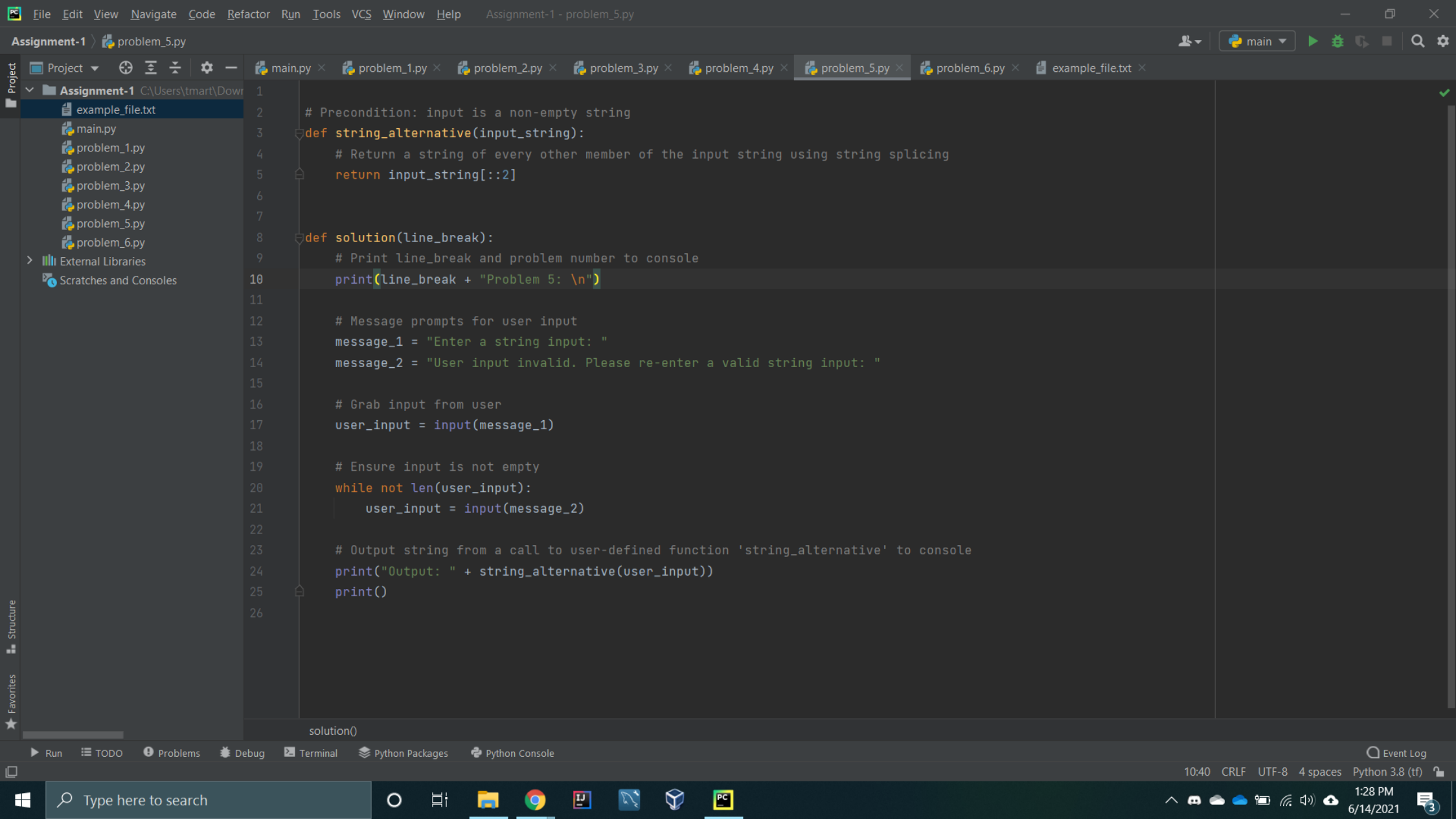
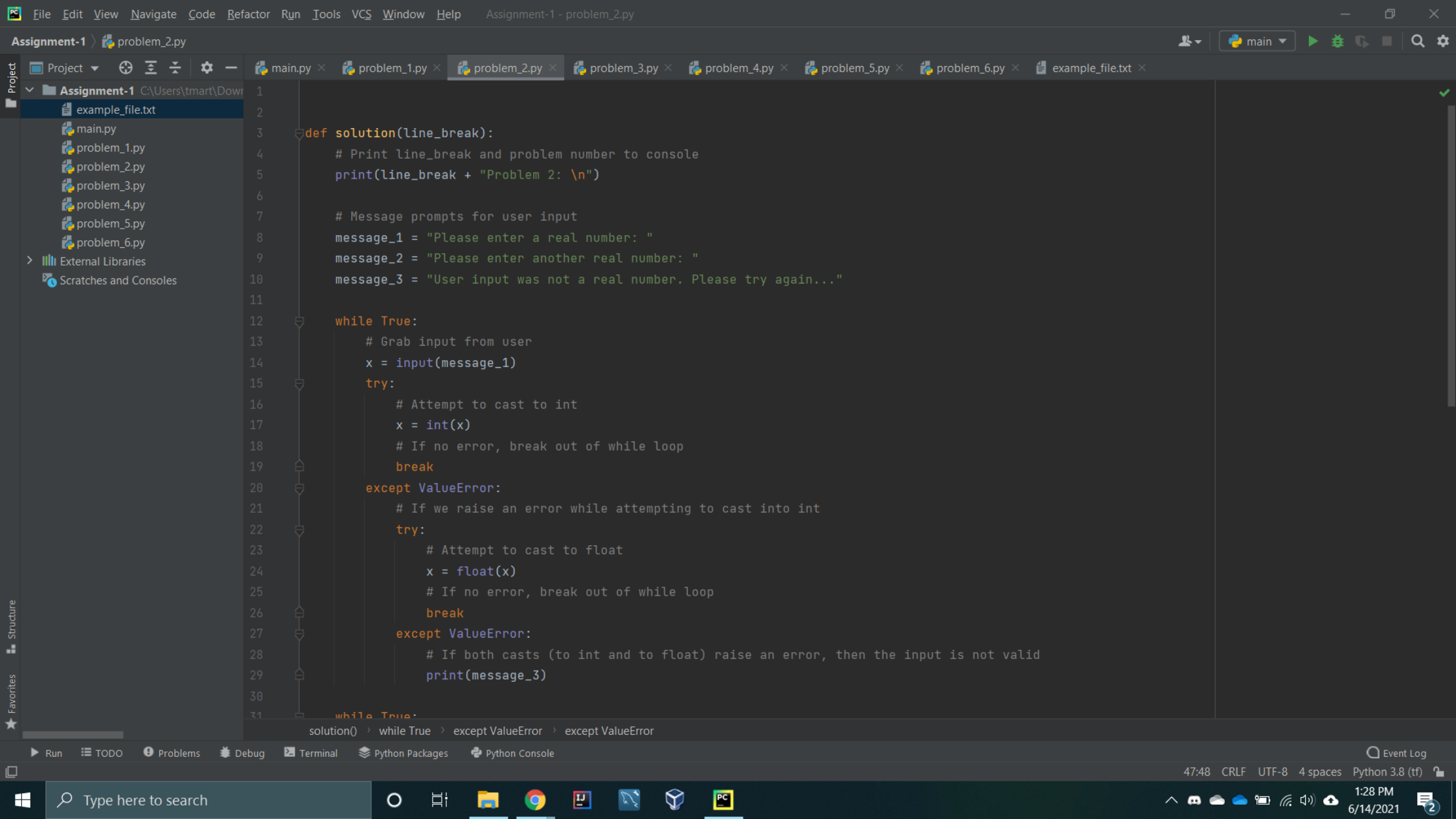
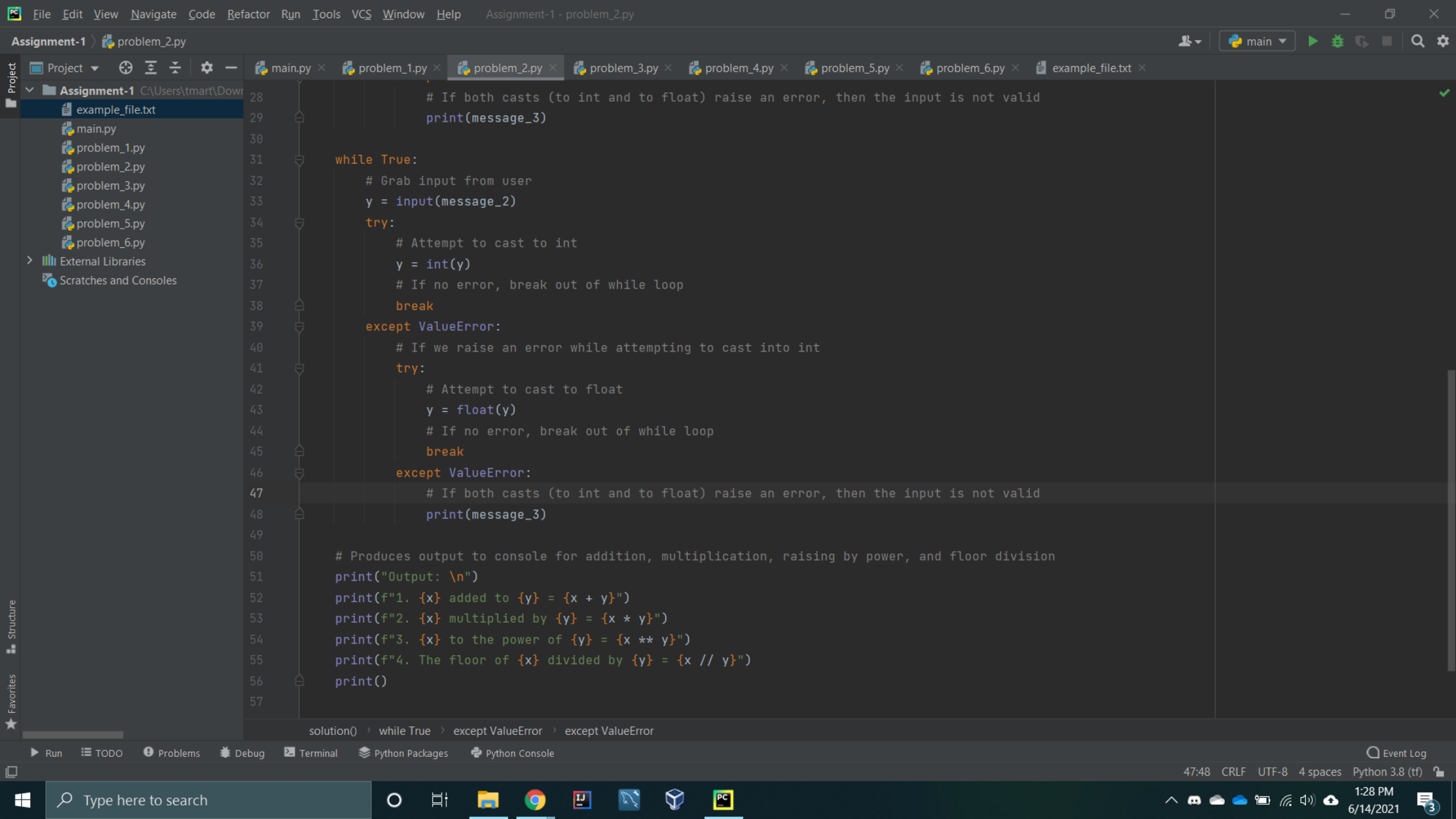

File Edit View Navigate Code Refactor Run Tools VCS Window Help
Assignment-1 - main.py
Assignment-1 main.py
main
Project
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
example_file.txt
Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles
import problem_1
import problem_2
import problem_3
import problem_4
import problem_5
import problem_6
if __name__ == '__main__':
# Program constant for breaking up console visual between each problem
line_break = "#" * 32 + "\n"
# Opening console output
print(line_break + " Solutions for Assignment 1: \n")
##
Calls to each problem, comment out to run any combination
##
problem_1.solution(line_break)
problem_2.solution(line_break)
problem_3.solution(line_break)
problem_4.solution(line_break)
problem_5.solution(line_break)
problem_6.solution(line_break)
##
# End of program console output
print(line_break + " --End of Program --\n" + line_break)
Run
TODO
Problems
Debug
Terminal
Python Packages
Python Console
Event Log
1:1
Python 3.8 (tf)
Type here to search
1:27 PM
6/14/2021

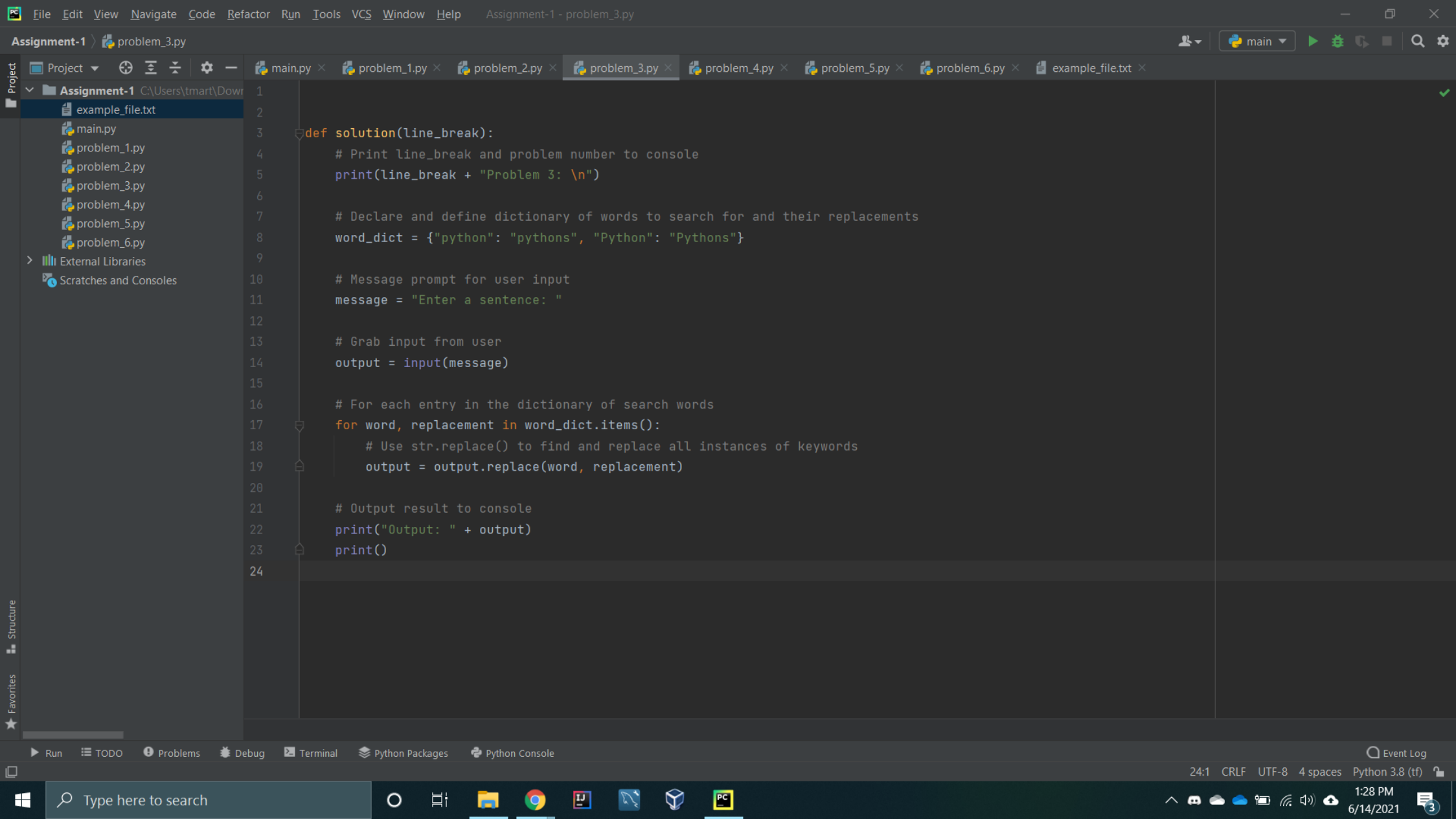

File Edit View Navigate Code Refactor Run Tools VCS Window Help
Assignment-1 - problem_5.py
Assignment-1 problem_5.py
main
Project
main.py problem_1.py problem_2.py problem_3.py problem_4.py problem_5.py problem_6.py example_file.txt
Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles
1 # Precondition: input is a non-empty string
2 def string_alternative(input_string):
3 # Return a string of every other member of the input string using string splicing
4 return input_string[::2]
5
6
7 def solution(line_break):
8 # Print line_break and problem number to console
9 print(line_break + "Problem 5: \n")
10
11 # Message prompts for user input
12 message_1 = "Enter a string input: "
13 message_2 = "User input invalid. Please re-enter a valid string input: "
14
15 # Grab input from user
16 user_input = input(message_1)
17
18 # Ensure input is not empty
19 while not len(user_input):
20 user_input = input(message_2)
21
22 # Output string from a call to user-defined function 'string_alternative' to console
23 print("Output: " + string_alternative(user_input))
24 print()
25
26
solution()
Run TODO Problems Debug Terminal Python Packages Python Console
Event Log
10:40 CRLF UTF-8 4 spaces Python 3.8 (tf)
Type here to search
1:28 PM
6/14/2021

Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles

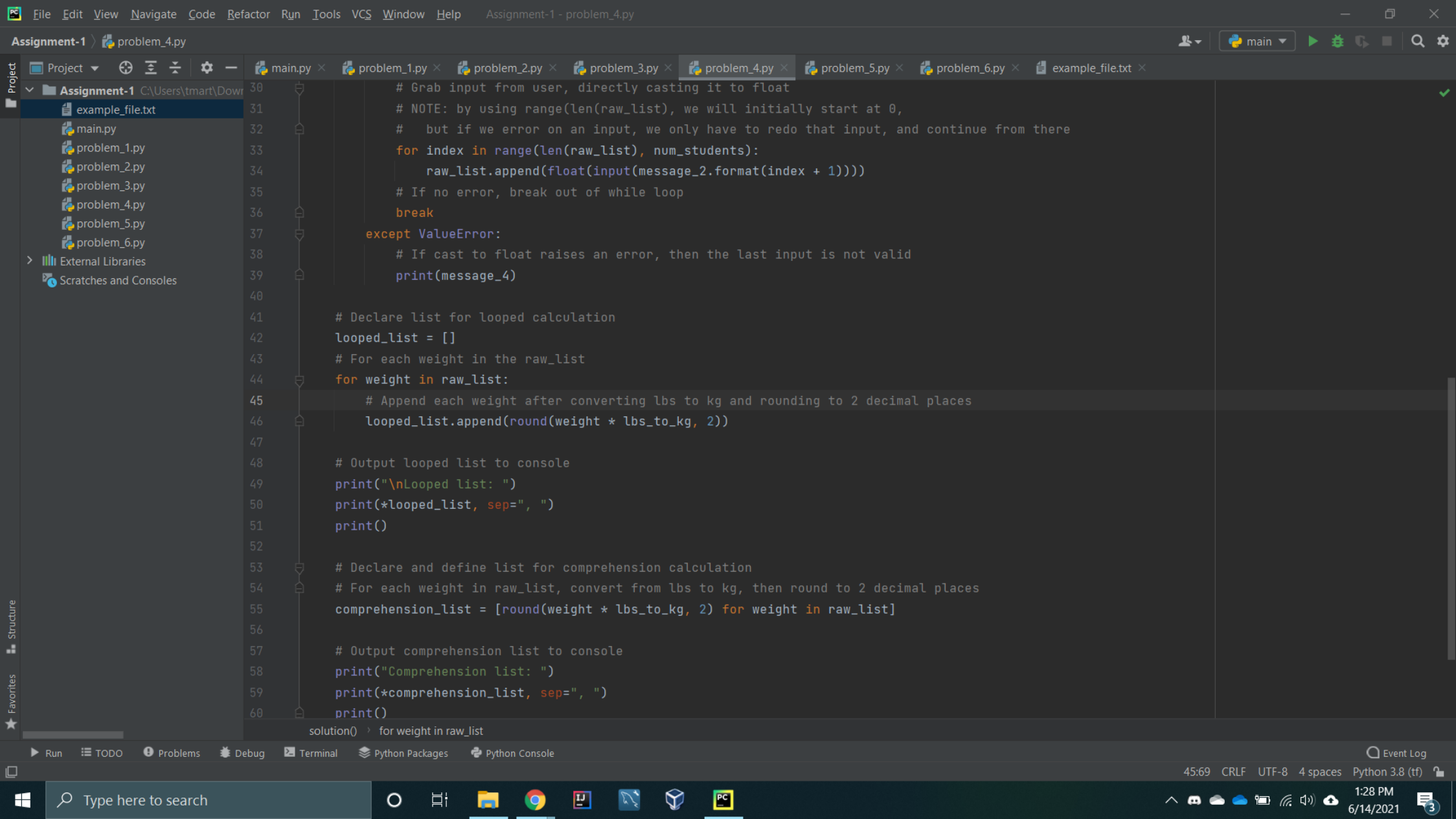
```python
import random


def solution(line_break):
    # Print line_break and problem number to console
    print(line_break + "Problem 1: \n")

    # Number of letters to delete, feel free to change
    num_letters_to_delete = 2

    # Message prompt for user input
    message = "Enter desired word(s): "

    # Grab input from user, instantiating a list in which to store input
    letters_list = list(input(message))

    # For as many letters as we wish to delete (Instructions say 2)
    for x in range(num_letters_to_delete):
        # If there is a member of letters_list (this ensures an IndexError doesn't happen for an empty list)
        if len(letters_list):
            # Pop a random index from letters_list
            letters_list.pop(random.randint(0, len(letters_list) - 1))

    # Join list to a string, then reverse string, then output to console
    print("Output: " + ''.join(letters_list)[::-1])
    print()

```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
Assignment-1 - problem_2.py
Assignment-1 problem_2.py
main
Project
main.py problem_1.py problem_2.py problem_3.py problem_4.py problem_5.py problem_6.py example_file.txt
Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles
1
2
3 def solution(line_break):
4 # Print line_break and problem number to console
5 print(line_break + "Problem 2: \n")
6
7 # Message prompts for user input
8 message_1 = "Please enter a real number: "
9 message_2 = "Please enter another real number: "
10 message_3 = "User input was not a real number. Please try again..."
11
12 while True:
13 # Grab input from user
14 x = input(message_1)
15 try:
16 # Attempt to cast to int
17 x = int(x)
18 # If no error, break out of while loop
19 break
20 except ValueError:
21 # If we raise an error while attempting to cast into int
22 try:
23 # Attempt to cast to float
24 x = float(x)
25 # If no error, break out of while loop
26 break
27 except ValueError:
28 # If both casts (to int and to float) raise an error, then the input is not valid
29 print(message_3)
30
31 while True:
solution() while True except ValueError except ValueError
Run TODO Problems Debug Terminal Python Packages Python Console
Event Log
47:48 CRLF UTF-8 4 spaces Python 3.8 (tf)
Type here to search
1:28 PM
6/14/2021


File Edit View Navigate Code Refactor Run Tools VCS Window Help
Assignment-1 - problem_2.py
Assignment-1 problem_2.py
main
Project
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
example_file.txt
Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles
28 # If both casts (to int and to float) raise an error, then the input is not valid
29 print(message_3)
30
31 while True:
32 # Grab input from user
33 y = input(message_2)
34 try:
35 # Attempt to cast to int
36 y = int(y)
37 # If no error, break out of while loop
38 break
39 except ValueError:
40 # If we raise an error while attempting to cast into int
41 try:
42 # Attempt to cast to float
43 y = float(y)
44 # If no error, break out of while loop
45 break
46 except ValueError:
47 # If both casts (to int and to float) raise an error, then the input is not valid
48 print(message_3)
49
50 # Produces output to console for addition, multiplication, raising by power, and floor division
51 print("Output: \n")
52 print(f"1. {x} added to {y} = {x + y}")
53 print(f"2. {x} multiplied by {y} = {x * y}")
54 print(f"3. {x} to the power of {y} = {x ** y}")
55 print(f"4. The floor of {x} divided by {y} = {x // y}")
56 print()
57
solution() while True except ValueError except ValueError
Run TODO Problems Debug Terminal Python Packages Python Console
Event Log
47:48 CRLF UTF-8 4 spaces Python 3.8 (tf)
Type here to search
1:28 PM
6/14/2021


File Edit View Navigate Code Refactor Run Tools VCS Window Help
Assignment-1 - problem_3.py
Assignment-1 problem_3.py
main
Project
main.py problem_1.py problem_2.py problem_3.py problem_4.py problem_5.py problem_6.py example_file.txt
Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles
1
2
3 def solution(line_break):
4 # Print line_break and problem number to console
5 print(line_break + "Problem 3: \n")
6
7 # Declare and define dictionary of words to search for and their replacements
8 word_dict = {"python": "pythons", "Python": "Pythons"}
9
10 # Message prompt for user input
11 message = "Enter a sentence: "
12
13 # Grab input from user
14 output = input(message)
15
16 # For each entry in the dictionary of search words
17 for word, replacement in word_dict.items():
18 # Use str.replace() to find and replace all instances of keywords
19 output = output.replace(word, replacement)
20
21 # Output result to console
22 print("Output: " + output)
23 print()
24
Structure
Favorites
Run TODO Problems Debug Terminal Python Packages Python Console
Event Log
24:1 CRLF UTF-8 4 spaces Python 3.8 (tf)
Type here to search
1:28 PM
6/14/2021

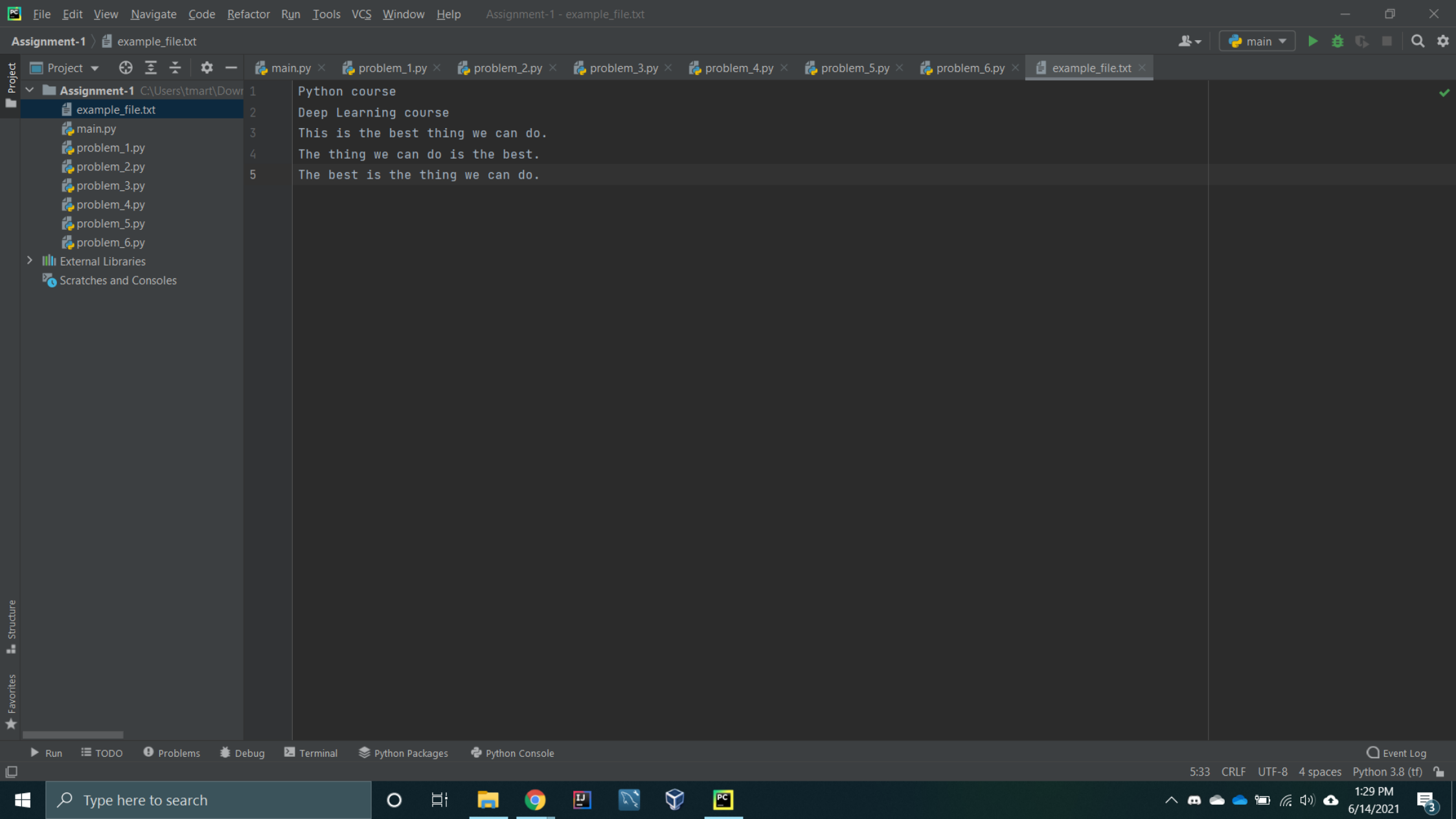
```
1
2
3   def solution(line_break):
4       # Print line_break and problem number to console
5       print(line_break + "Problem 4: \n")
6
7       # Conversion Constant
8       lbs_to_kg = 0.453592
9
10      # Message prompts for user input
11      message_1 = "Please enter the number of student weights would you like to compute: "
12      message_2 = "Enter the #{} student's weight (in lbs.): "
13      message_3 = "User input was not a whole number. Please try again..."
14      message_4 = "User input was not a real number. Please try again..."
15
16      while True:
17          try:
18              # Attempt to cast user input directly to int
19              num_students = int(input(message_1))
20              # If no error, break out of while loop
21              break
22          except ValueError:
23              # If cast to int raises an error, then the input is not valid
24              print(message_3)
25
26      # Declare list into which we will put the students' weights as input from user
27      raw_list = []
28      while True:
29          try:
30              # Grab input from user, directly casting it to float
31              # NOTE: by using range(len(raw_list)), we will initially start at 0
```

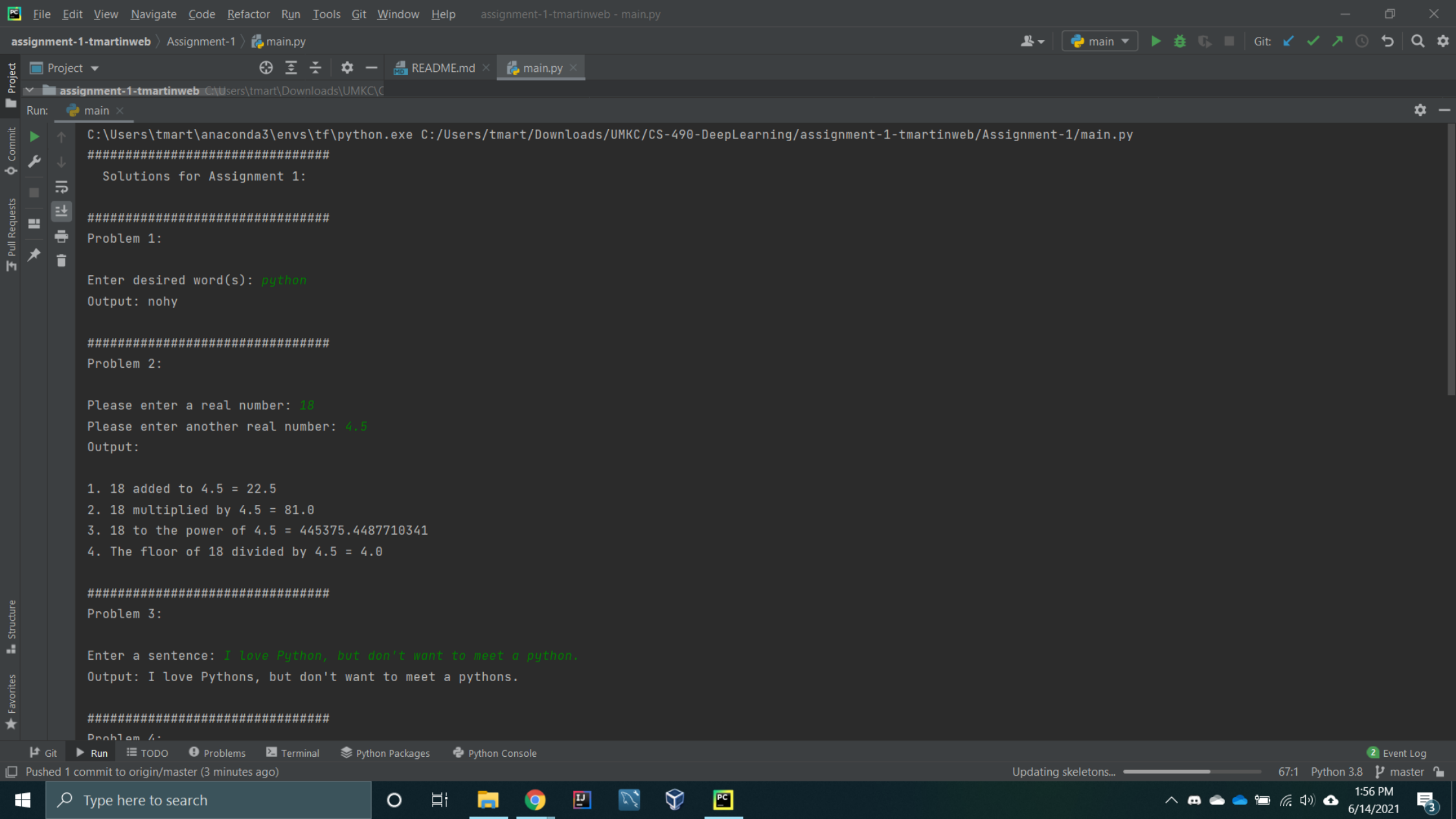

File Edit View Navigate Code Refactor Run Tools VCS Window Help
Assignment-1 - problem_4.py
Assignment-1 problem_4.py
main
Project
main.py problem_1.py problem_2.py problem_3.py problem_4.py problem_5.py problem_6.py example_file.txt
Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles
30 # Grab input from user, directly casting it to float
31 # NOTE: by using range(len(raw_list), we will initially start at 0,
32 # but if we error on an input, we only have to redo that input, and continue from there
33 for index in range(len(raw_list), num_students):
34 raw_list.append(float(input(message_2.format(index + 1))))
35 # If no error, break out of while loop
36 break
37 except ValueError:
38 # If cast to float raises an error, then the last input is not valid
39 print(message_4)
40
41 # Declare list for looped calculation
42 looped_list = []
43 # For each weight in the raw_list
44 for weight in raw_list:
45 # Append each weight after converting lbs to kg and rounding to 2 decimal places
46 looped_list.append(round(weight * lbs_to_kg, 2))
47
48 # Output looped list to console
49 print("\nLooped list: ")
50 print(*looped_list, sep=", ")
51 print()
52
53 # Declare and define list for comprehension calculation
54 # For each weight in raw_list, convert from lbs to kg, then round to 2 decimal places
55 comprehension_list = [round(weight * lbs_to_kg, 2) for weight in raw_list]
56
57 # Output comprehension list to console
58 print("Comprehension list: ")
59 print(*comprehension_list, sep=", ")
60 print()
solution() for weight in raw_list
Run TODO Problems Debug Terminal Python Packages Python Console
Event Log
45:69 CRLF UTF-8 4 spaces Python 3.8 (tf)
Type here to search
1:28 PM
6/14/2021

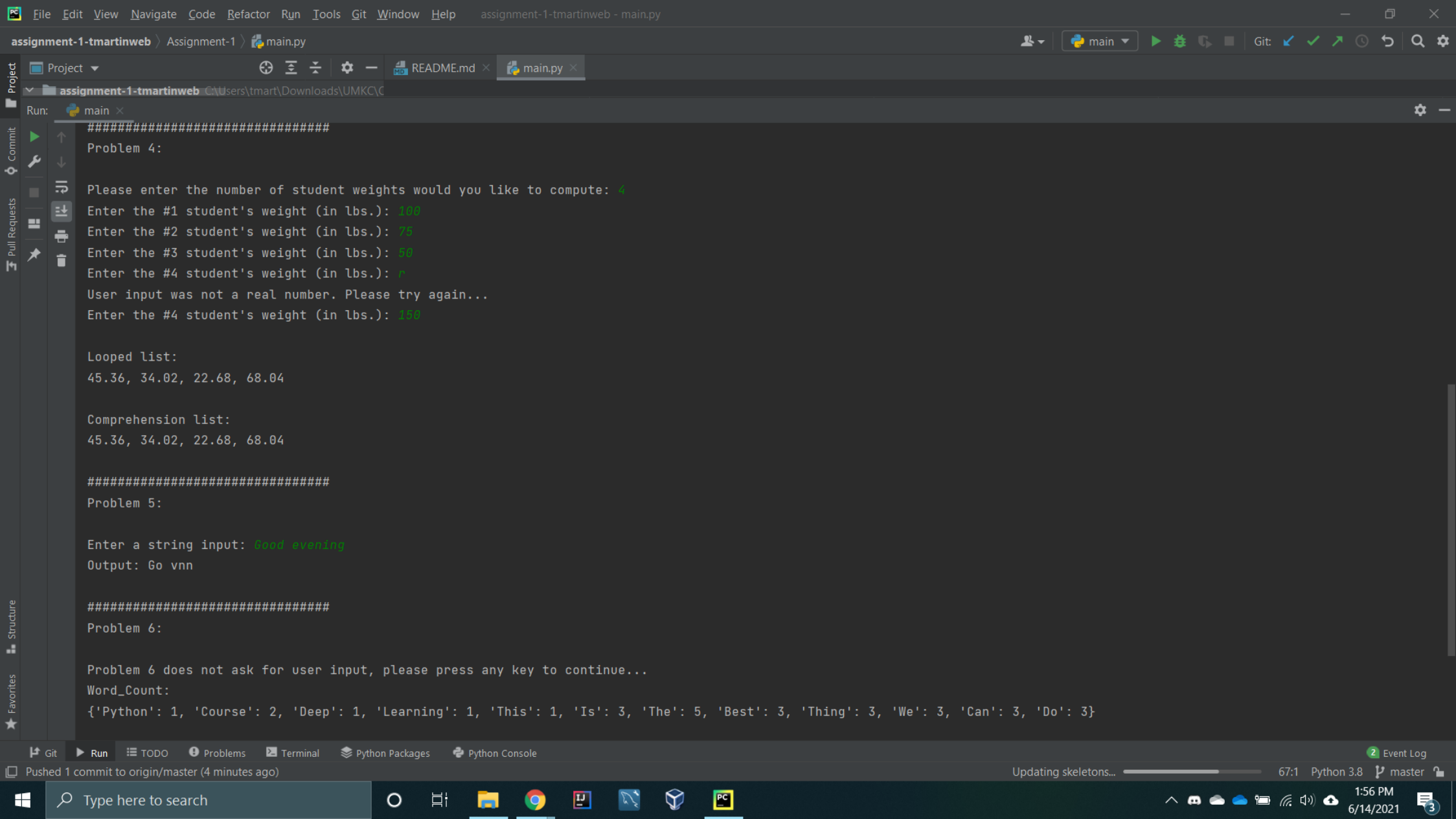
```python
import re


def solution(line_break):
    # Print line_break and problem number to console
    print(line_break + "Problem 6: \n")

    # I decided to put a pause for the beginning of problem 6,
    # since it doesn't require user input, it's easy for the console to 'get ahead' of the user
    message = "Problem 6 does not ask for user input, please press any key to continue..."
    input(message)

    # Name (and path, if required) of file to process
    filename = "example_file.txt"
    # Declare dictionary to store words counted (key) and their counts (value)
    word_dict = {}

    # Open file to read with auto-close method
    with open(filename, 'r') as file:
        for line in file:
            # Discard all non-alphanumeric and non-whitespace characters
            scrubbed_line = re.sub(r"[^\w\s]", "", line)
            # Discard all new lines
            scrubbed_line = re.sub(r"\n", "", scrubbed_line)
            # Create list from split string
            words_in_line = scrubbed_line.split(" ")

            for word in words_in_line:
                # Modify for consistent key capitalization
                cap_word = word.lower().capitalize()
                # If the word is in the dictionary
```

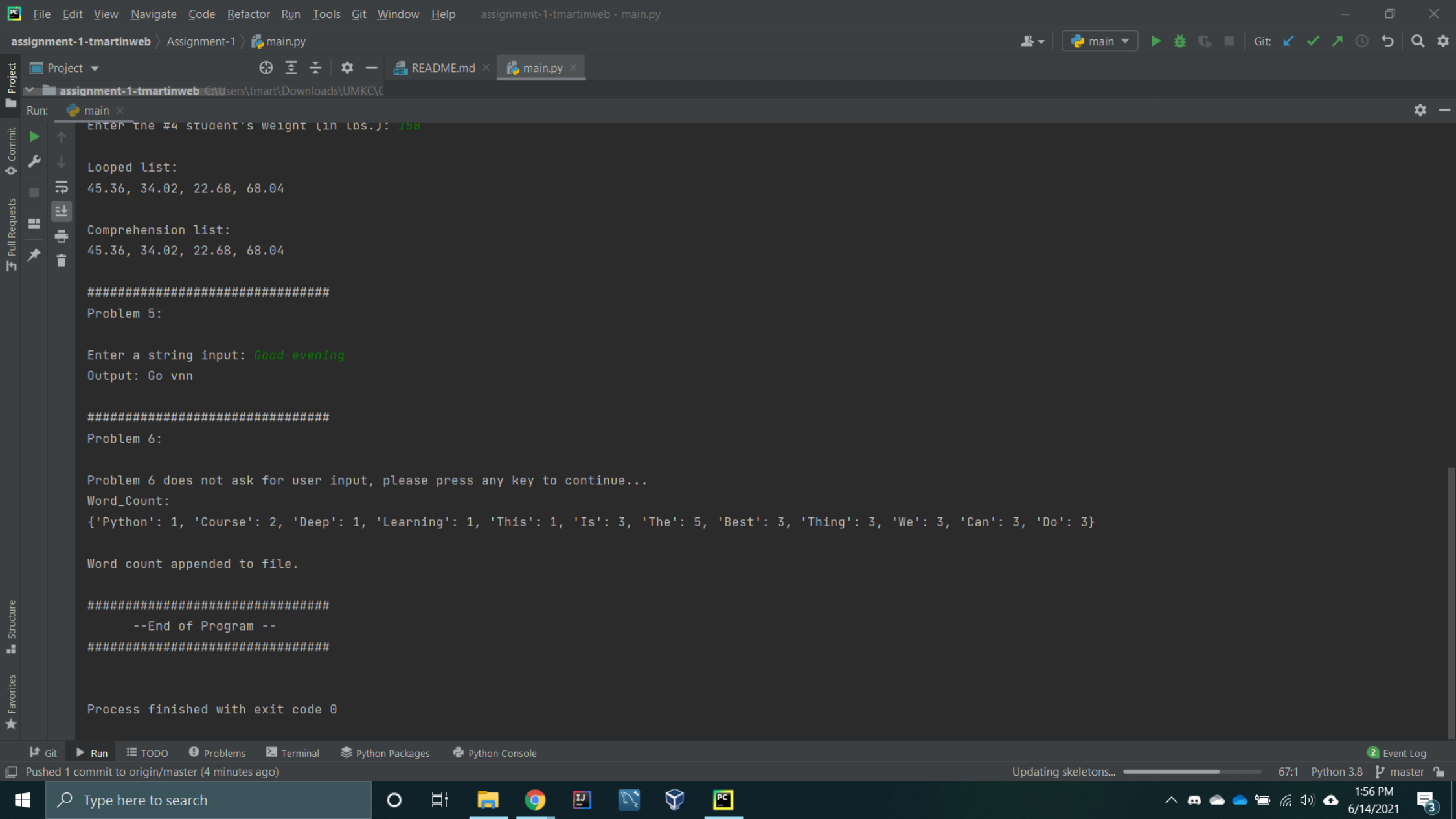
Assignment-1 C:\Users\tmart\Dowr
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles

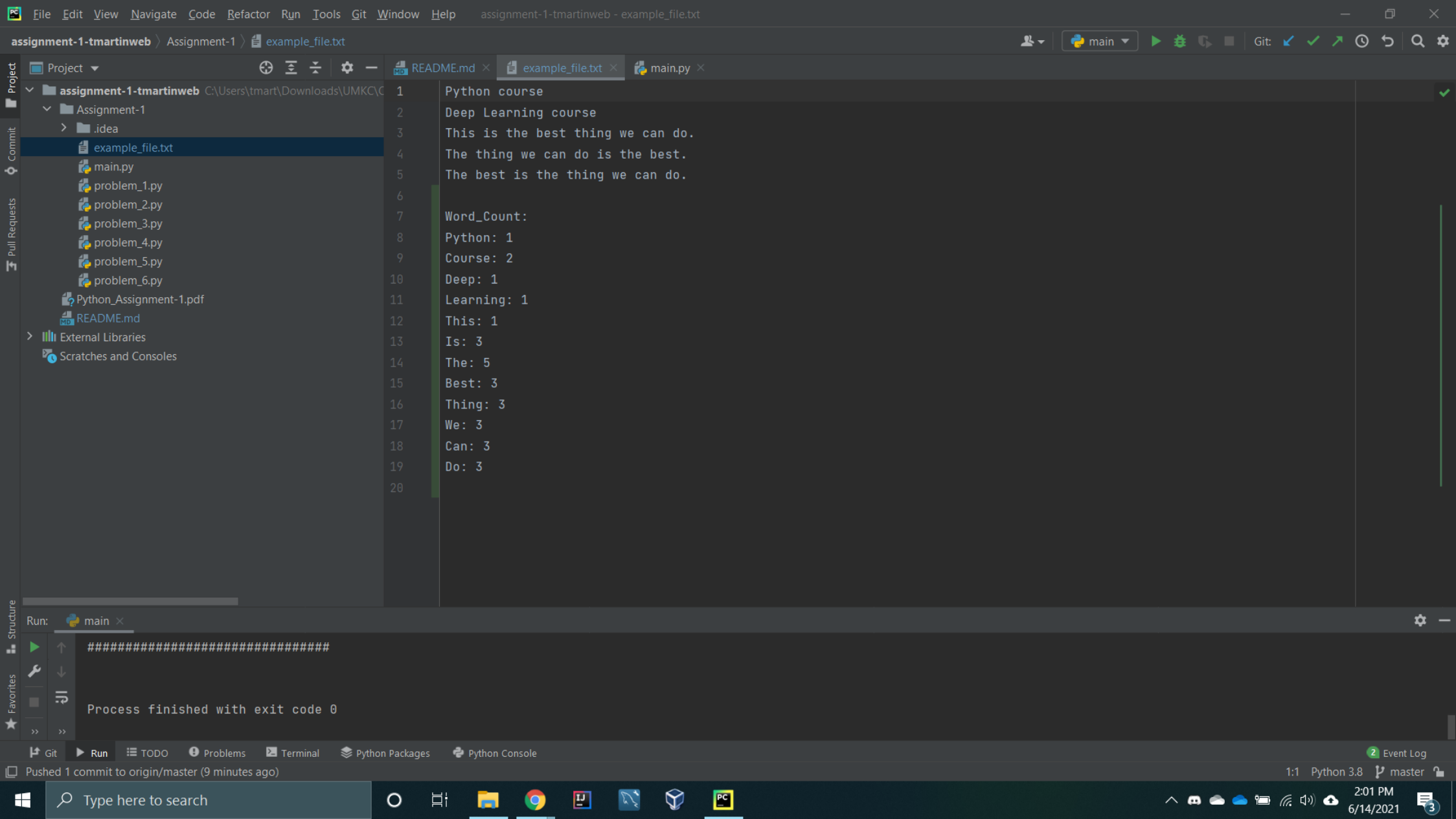
```
            for word in words_in_line:
                # Modify for consistent key capitalization
                cap_word = word.lower().capitalize()
                # If the word is in the dictionary...
                if cap_word in word_dict:
                    # Add 1 to its count
                    word_dict[cap_word] = word_dict[cap_word] + 1
                else:
                    # Create a new key with count 1
                    word_dict[cap_word] = 1

    # Output word count to console
    print("Word_Count: ")
    print(word_dict)

    # Open file to append, with auto-close method
    with open(filename, 'a') as file:
        # Append heading to file
        file.write("\n\nWord_Count: \n")
        # For each key/value pair in dictionary...
        for key, count in word_dict.items():
            # Append word and count to file, line by line
            file.write(str(key + ": " + str(count)) + "\n")

    # Print status to console
    print("\nWord count appended to file.")
    print()
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
Assignment-1 - example_file.txt
Assignment-1 example_file.txt
main
Project
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
example_file.txt
Assignment-1 C:\Users\tmart\Dow
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
External Libraries
Scratches and Consoles
1 Python course
2 Deep Learning course
3 This is the best thing we can do.
4 The thing we can do is the best.
5 The best is the thing we can do.
Structure
Favorites
Run
TODO
Problems
Debug
Terminal
Python Packages
Python Console
Event Log
5:33 CRLF UTF-8 4 spaces Python 3.8 (tf)
Type here to search
1:29 PM
6/14/2021


File Edit View Navigate Code Refactor Run Tools Git Window Help
assignment-1-tmartinweb - main.py
assignment-1-tmartinweb 〉 Assignment-1 〉 main.py
main
Git:
Project
README.md
main.py
assignment-1-tmartinweb C:\Users\tmart\Downloads\UMKC\C
Run: main
C:\Users\tmart\anaconda3\envs\tf\python.exe C:/Users/tmart/Downloads/UMKC/CS-490-DeepLearning/assignment-1-tmartinweb/Assignment-1/main.py
################################
Solutions for Assignment 1:
################################
Problem 1:
Enter desired word(s): python
Output: nohy
################################
Problem 2:
Please enter a real number: 18
Please enter another real number: 4.5
Output:
1. 18 added to 4.5 = 22.5
2. 18 multiplied by 4.5 = 81.0
3. 18 to the power of 4.5 = 445375.4487710341
4. The floor of 18 divided by 4.5 = 4.0
################################
Problem 3:
Enter a sentence: I love Python, but don't want to meet a python.
Output: I love Pythons, but don't want to meet a pythons.
################################
Problem 4:
Git
Run
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
Pushed 1 commit to origin/master (3 minutes ago)
Updating skeletons...
67:1
Python 3.8
master
Type here to search
1:56 PM
6/14/2021


File Edit View Navigate Code Refactor Run Tools Git Window Help
assignment-1-tmartinweb - main.py
assignment-1-tmartinweb 〉 Assignment-1 〉 main.py
main
Git:
Project
README.md
main.py
assignment-1-tmartinweb C:\Users\tmart\Downloads\UMKC\C
Run: main
################################
Problem 4:
Please enter the number of student weights would you like to compute: 4
Enter the #1 student's weight (in lbs.): 100
Enter the #2 student's weight (in lbs.): 75
Enter the #3 student's weight (in lbs.): 50
Enter the #4 student's weight (in lbs.): r
User input was not a real number. Please try again...
Enter the #4 student's weight (in lbs.): 150
Looped list:
45.36, 34.02, 22.68, 68.04
Comprehension list:
45.36, 34.02, 22.68, 68.04
################################
Problem 5:
Enter a string input: Good evening
Output: Go vnn
################################
Problem 6:
Problem 6 does not ask for user input, please press any key to continue...
Word_Count:
{'Python': 1, 'Course': 2, 'Deep': 1, 'Learning': 1, 'This': 1, 'Is': 3, 'The': 5, 'Best': 3, 'Thing': 3, 'We': 3, 'Can': 3, 'Do': 3}
Git Run TODO Problems Terminal Python Packages Python Console
Event Log
Pushed 1 commit to origin/master (4 minutes ago)
Updating skeletons...
67:1 Python 3.8 master
Type here to search
1:56 PM
6/14/2021


File Edit View Navigate Code Refactor Run Tools Git Window Help
assignment-1-tmartinweb - main.py
assignment-1-tmartinweb 〉 Assignment-1 〉 main.py
main
Git:
Project
README.md
main.py
assignment-1-tmartinweb C:\Users\tmart\Downloads\UMKC\C
Run: main
Enter the #4 student's weight (in lbs.): 150
Looped list:
45.36, 34.02, 22.68, 68.04
Comprehension list:
45.36, 34.02, 22.68, 68.04
################################
Problem 5:
Enter a string input: Good evening
Output: Go vnn
################################
Problem 6:
Problem 6 does not ask for user input, please press any key to continue...
Word_Count:
{'Python': 1, 'Course': 2, 'Deep': 1, 'Learning': 1, 'This': 1, 'Is': 3, 'The': 5, 'Best': 3, 'Thing': 3, 'We': 3, 'Can': 3, 'Do': 3}
Word count appended to file.
################################
--End of Program --
################################
Process finished with exit code 0
Git Run TODO Problems Terminal Python Packages Python Console
Event Log
Pushed 1 commit to origin/master (4 minutes ago)
Updating skeletons...
67:1 Python 3.8 master
Type here to search
1:56 PM
6/14/2021


File Edit View Navigate Code Refactor Run Tools Git Window Help
assignment-1-tmartinweb - example_file.txt
assignment-1-tmartinweb 〉 Assignment-1 〉 example_file.txt
main
Git:
Project
README.md
example_file.txt
main.py
assignment-1-tmartinweb C:\Users\tmart\Downloads\UMKC\C
Assignment-1
.idea
example_file.txt
main.py
problem_1.py
problem_2.py
problem_3.py
problem_4.py
problem_5.py
problem_6.py
Python_Assignment-1.pdf
README.md
External Libraries
Scratches and Consoles
Python course
Deep Learning course
This is the best thing we can do.
The thing we can do is the best.
The best is the thing we can do.
Word_Count:
Python: 1
Course: 2
Deep: 1
Learning: 1
This: 1
Is: 3
The: 5
Best: 3
Thing: 3
We: 3
Can: 3
Do: 3
Run: main
##################################
Process finished with exit code 0
Git Run TODO Problems Terminal Python Packages Python Console
Event Log
Pushed 1 commit to origin/master (9 minutes ago)
1:1 Python 3.8 master
Type here to search
2:01 PM
6/14/2021