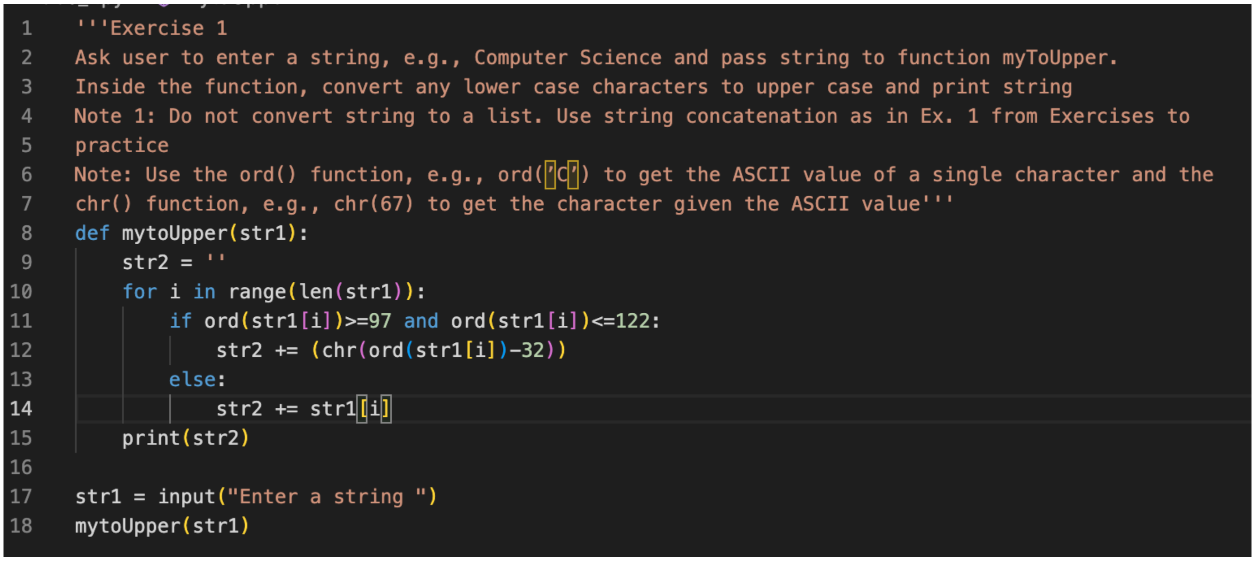
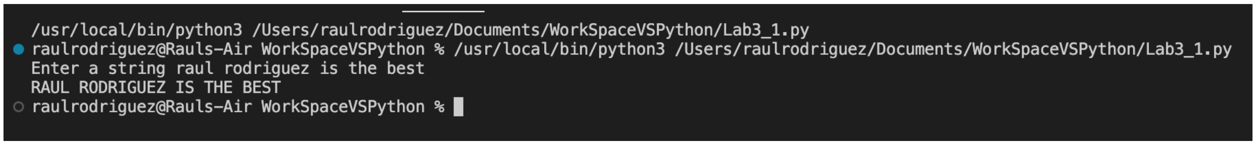
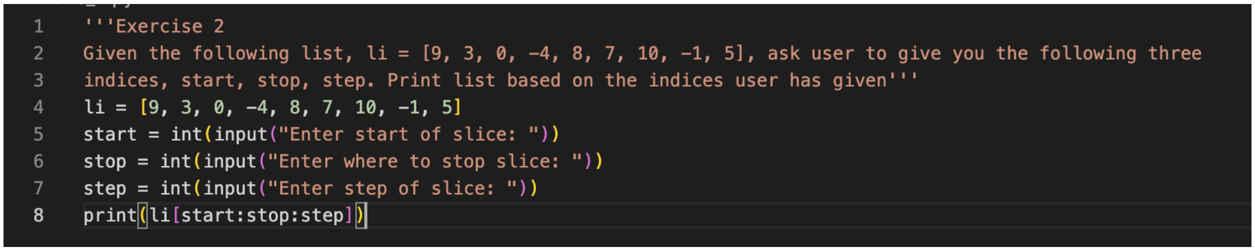
Raul Rodriguez
Python Intro to Data Science



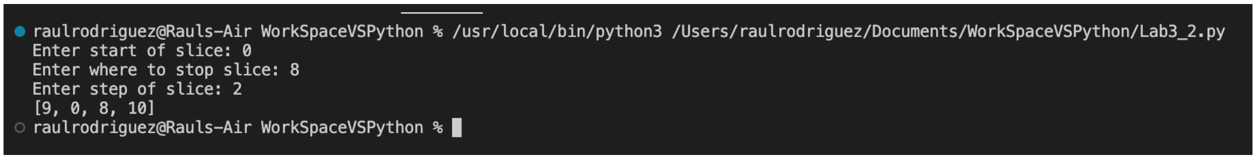
```python
'''Exercise 1
Ask user to enter a string, e.g., Computer Science and pass string to function myToUpper.
Inside the function, convert any lower case characters to upper case and print string
Note 1: Do not convert string to a list. Use string concatenation as in Ex. 1 from Exercises to
practice
Note: Use the ord() function, e.g., ord('C') to get the ASCII value of a single character and the
chr() function, e.g., chr(67) to get the character given the ASCII value'''
def mytoUpper(str1):
    str2 = ''
    for i in range(len(str1)):
        if ord(str1[i])>=97 and ord(str1[i])<=122:
            str2 += (chr(ord(str1[i])-32))
        else:
            str2 += str1[i]
    print(str2)

str1 = input("Enter a string ")
mytoUpper(str1)
```



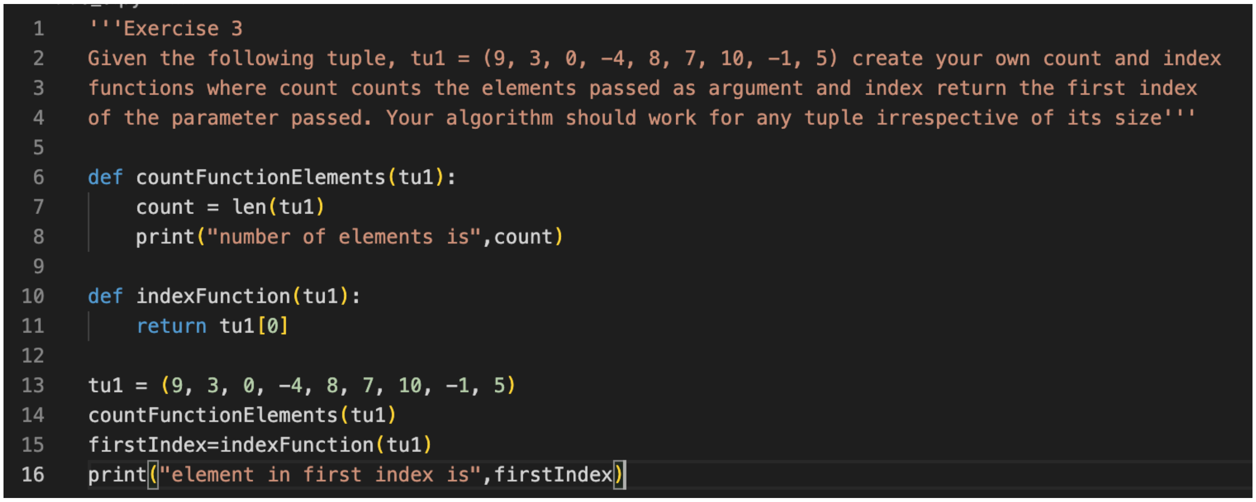
```
 /usr/local/bin/python3 /Users/raulrodriguez/Documents/WorkSpaceVSPython/Lab3_1.py
● raulrodriguez@Rauls-Air WorkSpaceVSPython % /usr/local/bin/python3 /Users/raulrodriguez/Documents/WorkSpaceVSPython/Lab3_1.py
 Enter a string raul rodriguez is the best
 RAUL RODRIGUEZ IS THE BEST
○ raulrodriguez@Rauls-Air WorkSpaceVSPython % █
```



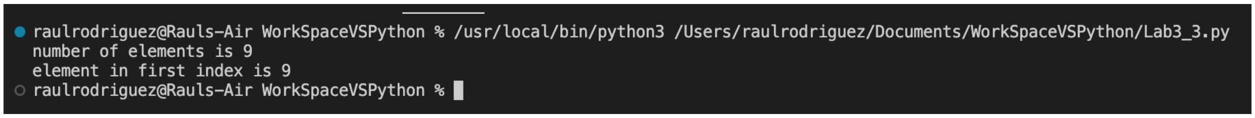
```python
'''Exercise 2
Given the following list, li = [9, 3, 0, -4, 8, 7, 10, -1, 5], ask user to give you the following three
indices, start, stop, step. Print list based on the indices user has given'''
li = [9, 3, 0, -4, 8, 7, 10, -1, 5]
start = int(input("Enter start of slice: "))
stop = int(input("Enter where to stop slice: "))
step = int(input("Enter step of slice: "))
print(li[start:stop:step])
```



```
● raulrodriguez@Rauls-Air WorkSpaceVSPython % /usr/local/bin/python3 /Users/raulrodriguez/Documents/WorkSpaceVSPython/Lab3_2.py
 Enter start of slice: 0
 Enter where to stop slice: 8
 Enter step of slice: 2
 [9, 0, 8, 10]
○ raulrodriguez@Rauls-Air WorkSpaceVSPython % █
```



```python
'''Exercise 3
Given the following tuple, tu1 = (9, 3, 0, -4, 8, 7, 10, -1, 5) create your own count and index
functions where count counts the elements passed as argument and index return the first index
of the parameter passed. Your algorithm should work for any tuple irrespective of its size'''

def countFunctionElements(tu1):
    count = len(tu1)
    print("number of elements is",count)

def indexFunction(tu1):
    return tu1[0]

tu1 = (9, 3, 0, -4, 8, 7, 10, -1, 5)
countFunctionElements(tu1)
firstIndex=indexFunction(tu1)
print("element in first index is",firstIndex)
```



```
● raulrodriguez@Rauls-Air WorkSpaceVSPython % /usr/local/bin/python3 /Users/raulrodriguez/Documents/WorkSpaceVSPython/Lab3_3.py
  number of elements is 9
  element in first index is 9
○ raulrodriguez@Rauls-Air WorkSpaceVSPython % 
```