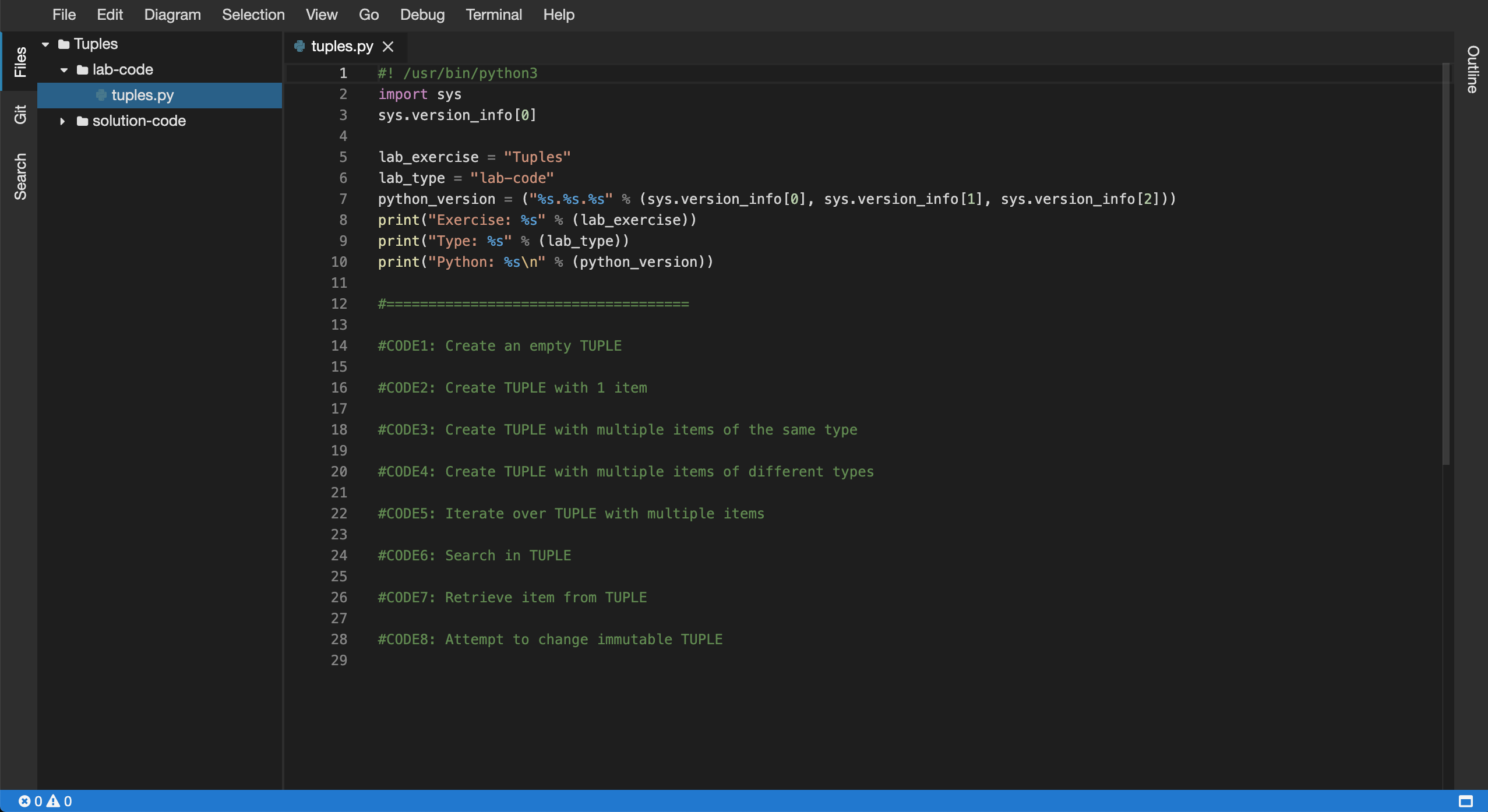
Tuples

Open the tuples.py source file within the editor. Take some time to review the uncompleted code within this file to understand the intended design:



5. Next, you will be required to complete the code in the following source files:

* tuples.py

6. Within the editor ensure that the tuples.py file now has focus.

7. Replace the //Code1: comment, create an empty TUPLE:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE1: Create an empty TUPLE

tup1 = ()

print("CODE1:")

print(f"tup1 = {tup1}")

print(f"data type = {type(tup1)}")

print(f"length = {len(tup1)}")

print()

8. Replace the //Code2: comment, create a TUPLE with 1 item:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE2: Create TUPLE with 1 item

tup2 = ("cloudacademy",)

print("CODE2:")

print(f"tup2 = {tup2}")

print(f"data type = {type(tup2)}")

print(f"length = {len(tup2)}")

print()

9. Replace the //Code3: comment, create a TUPLE with multiple items of the same type:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE3: Create TUPLE with multiple items of the same type

tup3 = (1, 2, 3, 4, 5)

print("CODE3:")

print(f"tup3 = {tup3}")

print(f"data type = {type(tup3)}")

print(f"length = {len(tup3)}")

print()

10. Replace the //Code4: comment, create a TUPLE with multiple items of different types:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE4: Create TUPLE with multiple items of different types

tup4 = ("cloud", "academy", 1, True, False)

print("CODE4:")

print(f"tup4 = {tup4}")

print(f"data type = {type(tup4)}")

print(f"length = {len(tup4)}")

print()

11. Replace the //Code5: comment, iterate over a TUPLE with multiple items:

Note: Ensure the indentation within the following code block is maintained within the editor view when you perform the copy and paste operation.

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE5: Iterate over TUPLE with multiple items

print("CODE5:")

for item in tup4:

print(item)

print()

12. Replace the //Code6: comment, search within the TUPLE:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE6: Search in TUPLE

print("CODE6:")

print ("cloud" in tup4)

print ("blah" in tup4)

print()

13. Replace the //Code7: comment, retrieve an item from the TUPLE:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE7: Retrieve item from TUPLE

print("CODE7:")

item0 = tup4[0]

item1 = tup4[1]

print(f"item0 = {item0}")

print(f"item1 = {item1}")

print()

14. Replace the //Code8: comment, attempt to change an immutable TUPLE:

Note: Ensure the indentation within the following code block is maintained within the editor view when you perform the copy and paste operation.

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

#CODE8: Attempt to change immutable TUPLE

print("CODE8:")

try:

tup4[0] = "not possible!"

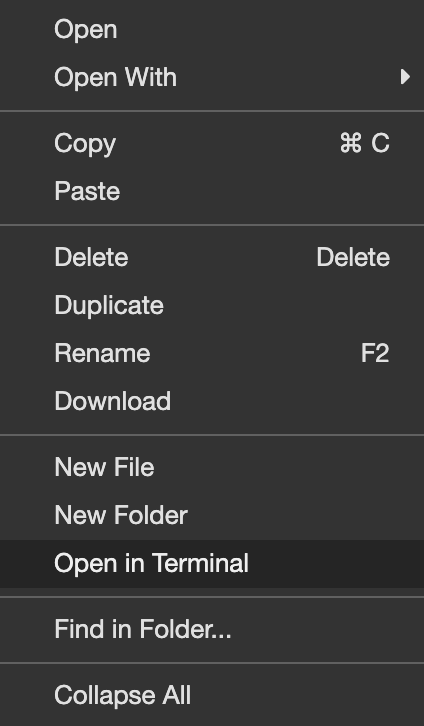
except:

print("Tuples are immutable!!")

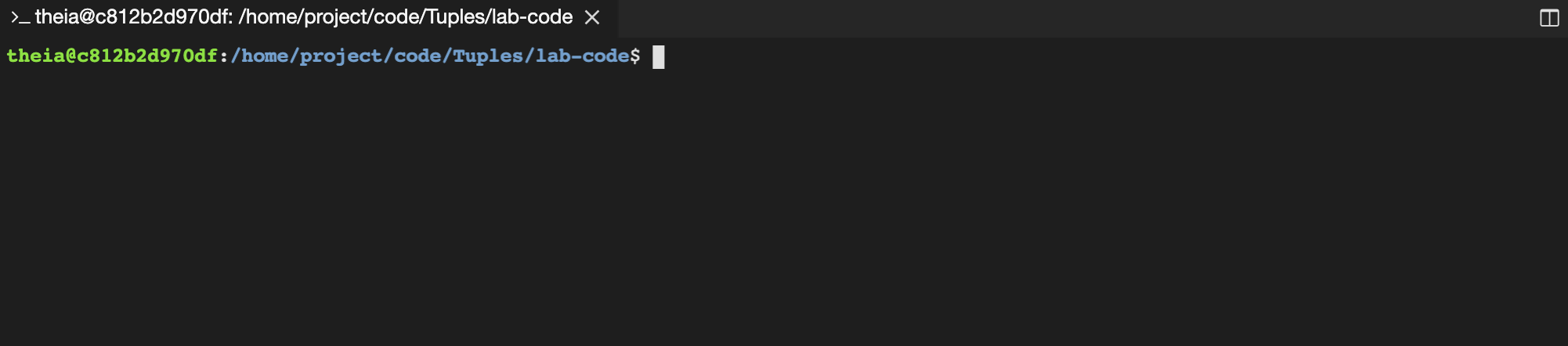
print()

15. Ok the tuples.py Python script is now ready to be executed by the Python interpreter. To do so you will need to use the embedded terminal to launch Python.

16. Within the Files tree view (lefthand side menu), select the lab-code folder and right click and select the Open in Terminal option:

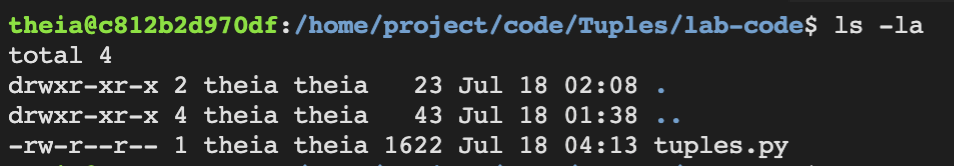


17. Access to the embedded terminal is now provided in the bottom pane of the current view:

  
18. Peform a directory listing on the current directory to ensure the presence of the tuples.py file, like so:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

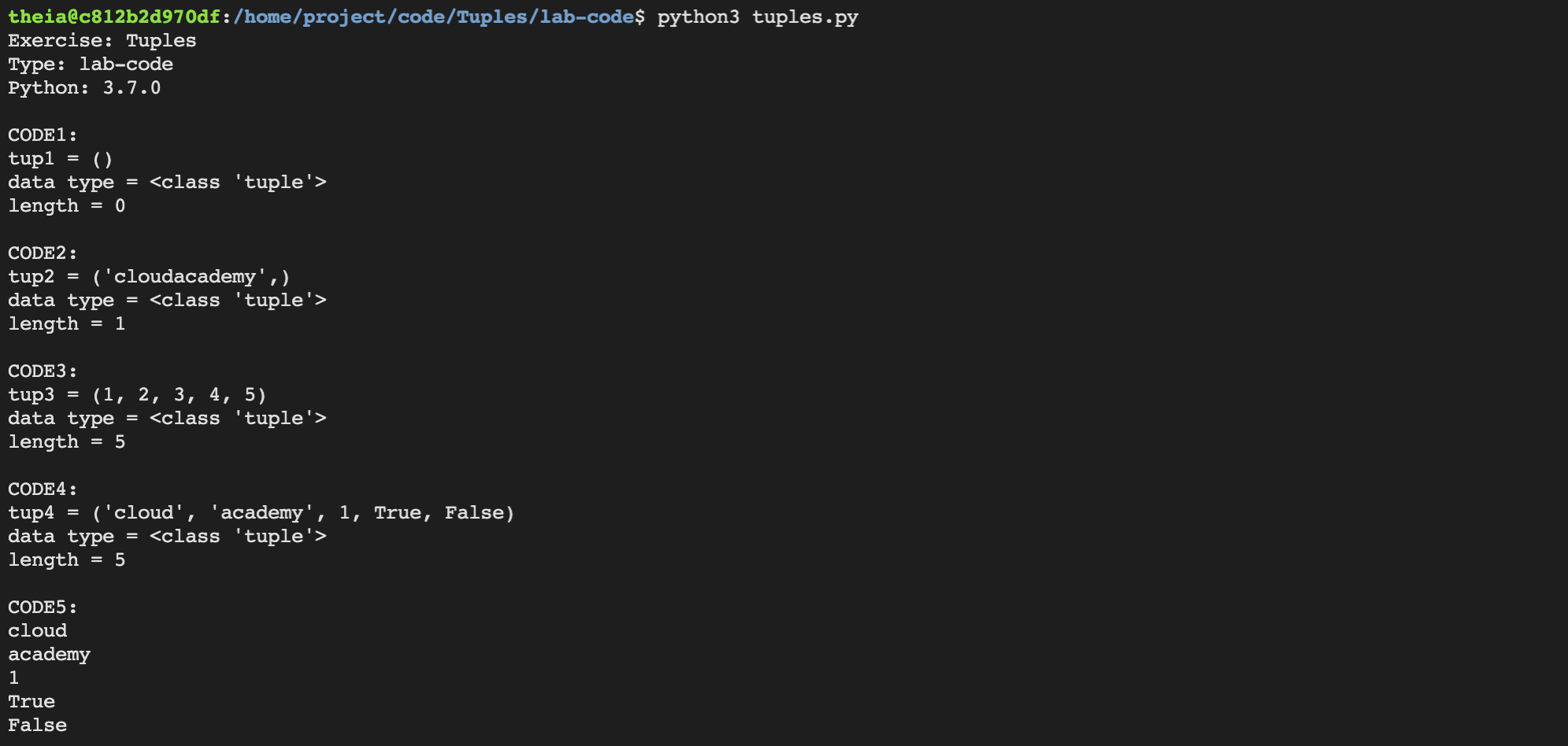
ls -la



19. Let's now execute the updated tuples.py script using the python3 command. Within the terminal enter the following command:

[**Copy code**](https://app.qa.com/lab/coding-python-collection-types/exercise-1-tuples/?context_id=637&context_resource=lp)

python3 tuples.py





20. Great! Your tuples.py script has executed successfully! This demonstrates how to work with Tuples.

21. Try updating the tuples.py script and then re-executing using the same steps above. See the following documentation for ideas:

[https://docs.python.org/3.3/tutorial/datastructures.html#tuples-and-sequences](https://docs.python.org/3.3/tutorial/datastructures.html" \l "tuples-and-sequences)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Solution code

#! /usr/bin/python3

import sys

sys.version\_info[0]

lab\_exercise = "Tuples"

lab\_type = "solution-code"

python\_version = ("%s.%s.%s" % (sys.version\_info[0], sys.version\_info[1], sys.version\_info[2]))

print("Exercise: %s" % (lab\_exercise))

print("Type: %s" % (lab\_type))

print("Python: %s\n" % (python\_version))

#====================================

#CODE1: Create an empty TUPLE

tup1 = ()

print("CODE1:")

print(f"tup1 = {tup1}")

print(f"data type = {type(tup1)}")

print(f"length = {len(tup1)}")

print()

#CODE2: Create TUPLE with 1 item

tup2 = ("cloudacademy",)

print("CODE2:")

print(f"tup2 = {tup2}")

print(f"data type = {type(tup2)}")

print(f"length = {len(tup2)}")

print()

#CODE3: Create TUPLE with multiple items of the same type

tup3 = (1, 2, 3, 4, 5)

print("CODE3:")

print(f"tup3 = {tup3}")

print(f"data type = {type(tup3)}")

print(f"length = {len(tup3)}")

print()

#CODE4: Create TUPLE with multiple items of different types

tup4 = ("cloud", "academy", 1, True, False)

print("CODE4:")

print(f"tup4 = {tup4}")

print(f"data type = {type(tup4)}")

print(f"length = {len(tup4)}")

print()

#CODE5: Iterate over TUPLE with multiple items

print("CODE5:")

for item in tup4:

print(item)

print()

#CODE6: Search in TUPLE

print("CODE6:")

print ("cloud" in tup4)

print ("blah" in tup4)

print()

#CODE7: Retrieve item from TUPLE

print("CODE7:")

item0 = tup4[0]

item1 = tup4[1]

print(f"item0 = {item0}")

print(f"item1 = {item1}")

print()

#CODE8: Attempt to change immutable TUPLE

print("CODE8:")

try:

tup4[0] = "not possible!"

except:

print("Tuples are immutable!!")

print()

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

output console

Exercise: Tuples

Type: solution-code

Python: 3.7.0

CODE1:

tup1 = ()

data type = <class 'tuple'>

length = 0

CODE2:

tup2 = ('cloudacademy',)

data type = <class 'tuple'>

length = 1

CODE3:

tup3 = (1, 2, 3, 4, 5)

data type = <class 'tuple'>

length = 5

CODE4:

tup4 = ('cloud', 'academy', 1, True, False)

data type = <class 'tuple'>

length = 5

CODE5:

cloud

academy

1

True

False

CODE6:

True

False

CODE7:

item0 = cloud

item1 = academy

CODE8:

Tuples are immutable!!

theia@production-session-93142-6dd54db79c-z8dtq:/home/project/code/Tuples/solution-code$