

CPS 109 LAB 5

Gurpreet Kaur
Mojan Nooshin Habibi

Agenda

01

Recursion

02

Mutability

03

Quiz

Recursion

- Covered by Mojan

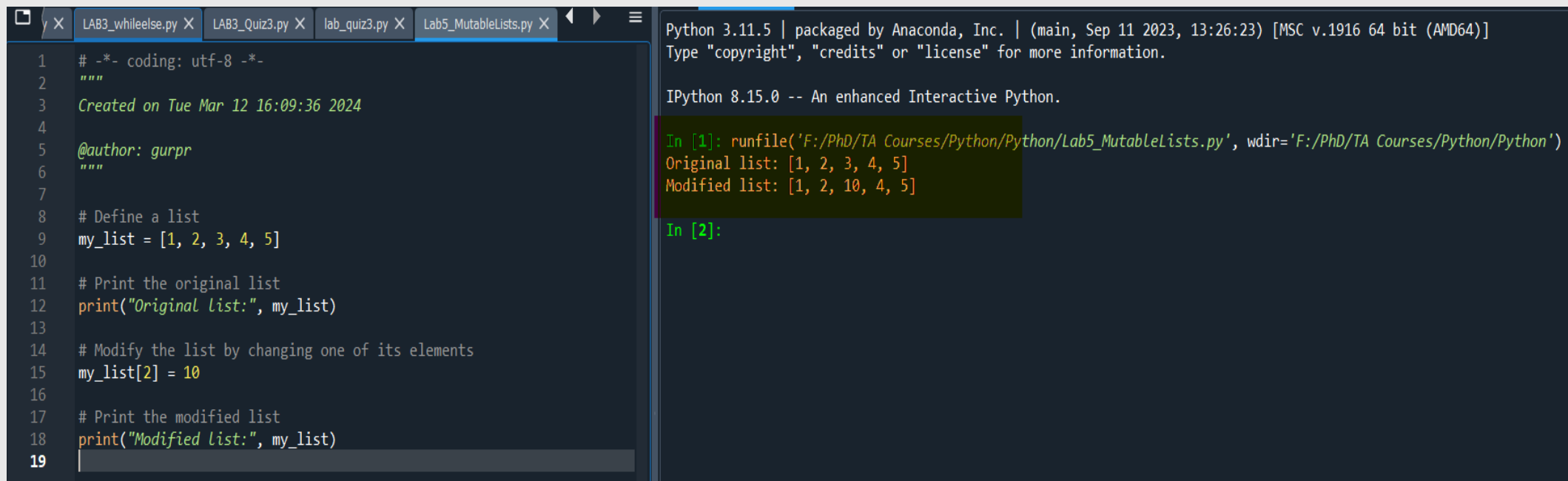
Mutability

- You might have heard us saying that lists are “mutable”, and strings are “immutable”
- Can you tell what does that mean?

Mutability

- Something is **mutable** if its contents can be changed (like a list).
- Something is **immutable** if its contents cannot be changed (like a string or a tuple)

Example :: Lab5_MutableLists.py



The image shows a code editor with four tabs: LAB3_whileelse.py, LAB3_Quiz3.py, lab_quiz3.py, and Lab5_MutableLists.py. The Lab5_MutableLists.py tab is active, displaying the following Python code:

```
1  #-*- coding: utf-8 -*-
2  """
3  Created on Tue Mar 12 16:09:36 2024
4
5  @author: gurpr
6  """
7
8  # Define a list
9  my_list = [1, 2, 3, 4, 5]
10
11 # Print the original list
12 print("Original list:", my_list)
13
14 # Modify the list by changing one of its elements
15 my_list[2] = 10
16
17 # Print the modified list
18 print("Modified list:", my_list)
19
```

To the right of the code editor is an IPython terminal window. It displays the following text:

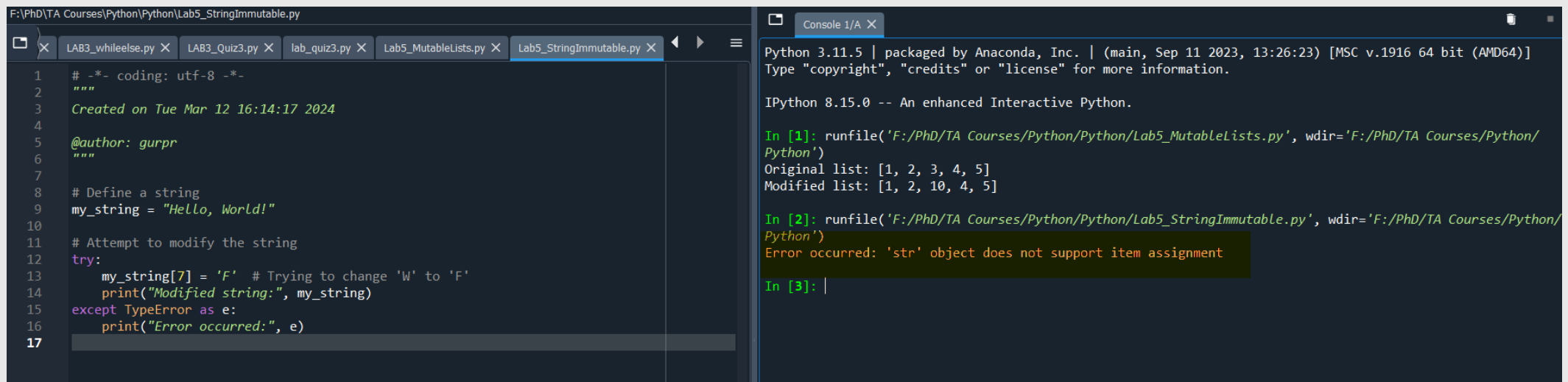
```
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.15.0 -- An enhanced Interactive Python.

In [1]: runfile('F:/PhD/TA Courses/Python/Python/Lab5_MutableLists.py', wdir='F:/PhD/TA Courses/Python/Python')
Original list: [1, 2, 3, 4, 5]
Modified list: [1, 2, 10, 4, 5]

In [2]:
```

Example : Lab5_StringsImmutable.py



The image shows a code editor window with the file `Lab5_StringsImmutable.py` open. The code defines a string `my_string` and attempts to modify it, which results in a `TypeError`. The IPython console on the right shows the execution of the script, including the error message.

```
F:\PhD\TA Courses\Python\Python\Lab5_StringsImmutable.py
LAB3_whileelse.py X LAB3_Quiz3.py X lab_quiz3.py X Lab5_MutableLists.py X Lab5_StringsImmutable.py X
1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Mar 12 16:14:17 2024
4
5  @author: gurpr
6  """
7
8  # Define a string
9  my_string = "Hello, World!"
10
11 # Attempt to modify the string
12 try:
13     my_string[7] = 'F' # Trying to change 'W' to 'F'
14     print("Modified string:", my_string)
15 except TypeError as e:
16     print("Error occurred:", e)
17
```

Console 1/A X

```
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

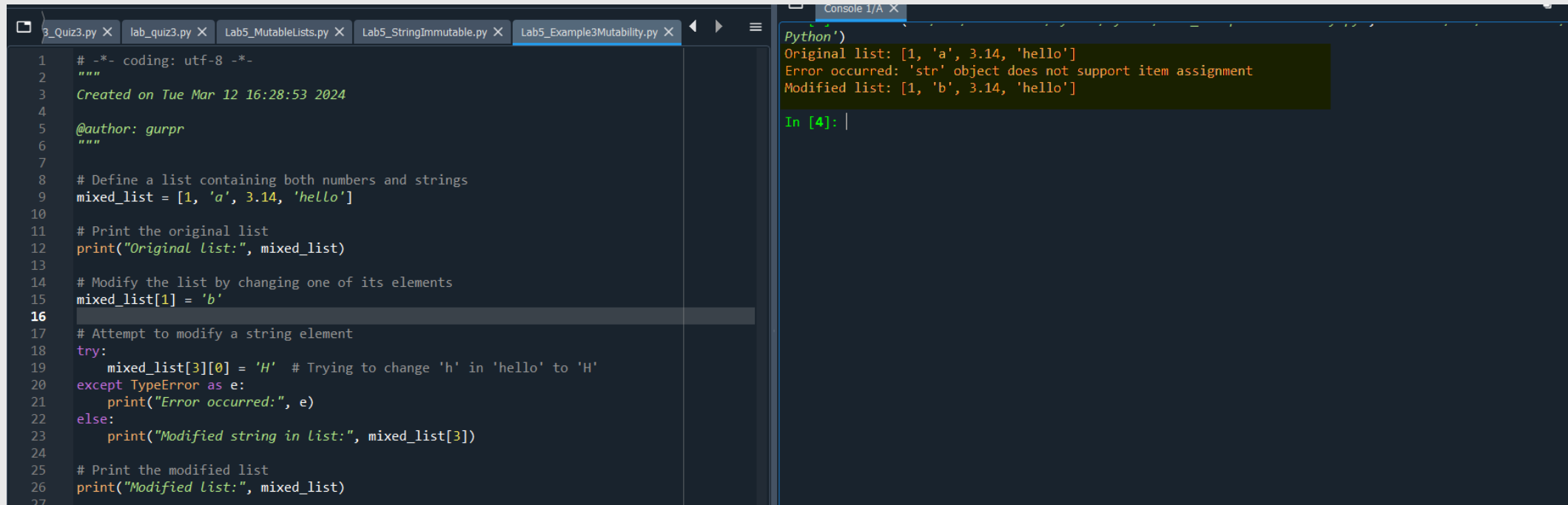
IPython 8.15.0 -- An enhanced Interactive Python.

In [1]: runfile('F:/PhD/TA Courses/Python/Python/Lab5_MutableLists.py', wdir='F:/PhD/TA Courses/Python/
Python')
Original list: [1, 2, 3, 4, 5]
Modified list: [1, 2, 10, 4, 5]

In [2]: runfile('F:/PhD/TA Courses/Python/Python/Lab5_StringsImmutable.py', wdir='F:/PhD/TA Courses/Python/
Python')
Error occurred: 'str' object does not support item assignment

In [3]:
```

Combined Example : Example3Mutability.py



The screenshot shows a Python IDE with a file explorer at the top displaying several files: 3_Quiz3.py, lab_quiz3.py, Lab5_MutableLists.py, Lab5_StringImmutable.py, and Lab5_Example3Mutability.py. The main editor window shows the code for Lab5_Example3Mutability.py. The code defines a mixed list, prints it, modifies an element, and attempts to modify a string element, which results in a TypeError. The console on the right shows the output of the script, including the original list, the error message, and the modified list.

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Mar 12 16:28:53 2024
4
5  @author: gurpr
6  """
7
8  # Define a list containing both numbers and strings
9  mixed_list = [1, 'a', 3.14, 'hello']
10
11 # Print the original list
12 print("Original list:", mixed_list)
13
14 # Modify the list by changing one of its elements
15 mixed_list[1] = 'b'
16
17 # Attempt to modify a string element
18 try:
19     mixed_list[3][0] = 'H' # Trying to change 'h' in 'hello' to 'H'
20 except TypeError as e:
21     print("Error occurred:", e)
22 else:
23     print("Modified string in list:", mixed_list[3])
24
25 # Print the modified list
26 print("Modified list:", mixed_list)
27
```

Console 1/A X

Python')
Original list: [1, 'a', 3.14, 'hello']
Error occurred: 'str' object does not support item assignment
Modified list: [1, 'b', 3.14, 'hello']

In [4]: |

Thanks!!

Any Questions !!

Quiz Time !!

Lab #5 Quiz Task)

Write a function that performs a “Faro shuffle” of an input list. A Faro shuffle splits the list into two halves, left and right, and then interleaves the elements of each half to create a new list.

Give them the link below, have them look it up and see how it’s done.

https://en.wikipedia.org/wiki/Faro_shuffle

Their function will have two parameters – the first being the incoming list, and the second being a Boolean. If the Boolean is True, perform an “out shuffle”, and if the Boolean is false, perform an “in shuffle”. Once again, have them read up on the meaning of this from the link above. Some sample inputs and outputs are below.

list	out?	Expected result
[1, 2, 3, 4, 5, 6, 7, 8]	True	[1, 5, 2, 6, 3, 7, 4, 8]
[1, 2, 3, 4, 5, 6, 7, 8]	False	[5, 1, 6, 2, 7, 3, 8, 4]
[]	True	[]
['bob', 'jack']	True	['bob', 'jack']
['bob', 'jack']	False	['jack', 'bob']