EECS 1015: LAB #6 – More on Lists, Dictionaries, and Tuples
Assigned: Nov 1, 2021
Due date: Nov 12, 2021 (Friday) [11.59pm Eastern Time]

#Important reminder

1) You must submit your lab via web-submit.

2) Please make sure you correctly submit your file (only a single file, please – `lab6.py`).

3) Please follow the instructions carefully – read the lab carefully to understand everything you need to do.

1. GOALS/OUTCOMES FOR LAB

- To practice using lists, dictionaries, and tuples
- To use for-loops with lists, dictionaries, and tuples
- To practice writing your own code and functions

2. LAB 6 – TASK/INSTRUCTIONS

Task 0: [This will be the same for all labs]: Start your code with comments that include this lab ID, your full name, email address, and student id as follows:

```
# Lab 6
# Author: Michael S. Brown
# Email: msb99898@aol.com
# Student ID: 10233030
# Section A or B
```

This lab only has two main tasks. You will need to use the start code at this link. Several pre-defined global variables are required. Link to code here: https://trinket.io/python/c68d8b0081

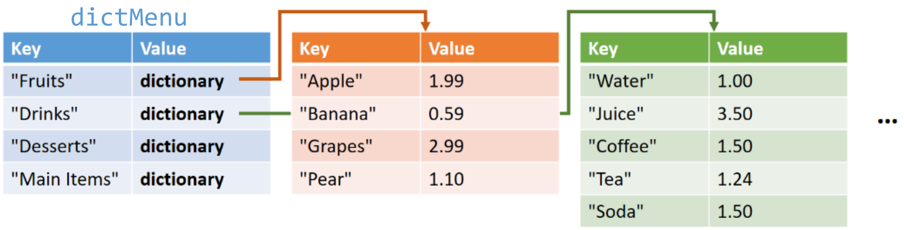
```
# variables for task 1
dictMenu = {'Fruits': {'Apple': 1.99, 'Banana': 0.59, 'Kiwi': 1.1, 'Grapes': 2.99, 'Pear': 2.15}, 'Drinks':
{'Water': 1.0, 'Juice': 3.5, 'Coffee': 1.5, 'Soda': 1.5, 'Tea': 1.25}, 'Dessert': {'Ice Cream': 2.99, 'Pie':
2.5, 'Cake': 2.75}, 'Main Dishes': {'Masala Dosa': 4.25, 'Jianbing': 2.88, 'Falafel': 5.15, 'Pizza': 8.5}}

# variables for task 2
decoder = {80: 'P', 121: 'y', 116: 't', 104: 'h', 111: 'o', 110: 'n', 105: 'i', 115: 's', 99: 'c', 108: 'l',
46: '.', 32: ' ', 44: ',', 45: '-', 95: '_', 40: '(', 42: '*', 41: ')', 47: '/', 92: '\\', 61: '=', 39: "'",
124: '|', 96: '`', 58: ':', 59: ';'}
msg1 = [[80, 121, 116, 104, 111, 110], [105, 115], [99, 111, 111, 108, 46]]
msg2 = [[32, 32, 32, 44, 45, 46], [32, 95, 40, 42, 95, 42, 41, 95], [40, 95, 32, 32, 111, 32, 32, 95, 41],
[32, 32, 47, 32, 111, 32, 92], [32, 40, 95, 47, 32, 92, 95, 41, 32]]
msg3 = [[32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 40], [32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 41], [32, 32,
32, 32, 32, 95, 95, 46, 46, 45, 45, 45, 46, 46, 95, 95], [32, 44, 45, 61, 39, 32, 32, 47, 32, 32, 124, 32,
32, 92, 32, 32, 96, 61, 45, 46], [58, 45, 45, 46, 46, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 46, 46, 45,
45, 59], [32, 92, 46, 44, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 95, 44, 46, 47]]
```

Video for this lab: https://www.eecs.yorku.ca/~mbrown/EECS1015_Lab6.mp4

Instructions for each task are given on the following pages.

TASK 1 – Printing dictionary of dictionaries [5 points – dictionary of dictionaries]

In the starting code, there is a dictionary variable `dictMenu`. This dictionary has keys that are strings. Each string represents a food category on the menu (e.g., "Fruits", "Drinks", .."). The keys of `dictMenu`have values that are *dictionaries*. These associated dictionaries have `key:value` pairs that represent the item's name as a string and its associated price as a float.
See diagram below giving a visual representation of `dictMenu.`



dictMenu

Key	Value
"Fruits"	**dictionary**
"Drinks"	**dictionary**
"Desserts"	**dictionary**
"Main Items"	**dictionary**

Key	Value
"Apple"	1.99
"Banana"	0.59
"Grapes"	2.99
"Pear"	1.10

Key	Value
"Water"	1.00
"Juice"	3.50
"Coffee"	1.50
"Tea"	1.24
"Soda"	1.50

…

Task 1 is to print out all the items in the `dictMenu` in the format shown below (all floats should be formatted with 2 decimal places). This task is not very complicated (only a few lines of code), but you need to think carefully about how the data is organized. Note that it is OK if your order is different than what is shown below:

```
------ Task 1 ------
----Fruits----
$1.99    Apple
$0.59    Banana
$1.10    Kiwi
$2.99    Grapes
$2.15    Pear
----Drinks----
$1.00    Water
$3.50    Juice
$1.50    Coffee
$1.50    Soda
$1.25    Tea
----Dessert----
$2.99    Ice Cream
$2.50    Pie
$2.75    Cake
----Main Dishes----
$4.25    Masala Dosa
$2.88    Jianbing
$5.15    Falafel
$8.50    Pizza
```

keys from main dictMenu. Format their output as shown. (Note – it is OK if your program prints out in a different order)

value:key pair of the dictionaries associated with each dictMenu's key. Format their output as shown. (Note – it is OK if your program prints out in a different order)

TASK 2 – Message decoding [5 points – list of lists, using dictionaries]

For task2, you'll use four variables in the starting code: `decoder`, `msg1`, `msg2`, `msg3`.

The `decoder` dictionary consists of integer keys that represent codes. The value of each key is a single character string.

The message strings (`msg1`, `msg2`, `msg3`) are lists of lists. Each inner list is a sequence of integers (code) that represents a character in a string. Use the `decoder` dictionary to look up the correct character to convert the list to a string.
See diagram below:

The decoder variable is a dictionary where the key in an integer (code) and the associated value is a string with a single character.

```
decoder = { code:'Y', code:'Z', code:'X' }
```

The **msg1** variable(s) [there are three in your lab, msg1, msg2, msg3] are lists of lists. Each inner list is a set of code numbers that represent a single character in a string.

```
msg1 = [  [code, code, code, .., code],  . . . ,  [code, code, code .., code]  ]
```

Convert the lists of codes to a string

[code, code, ..., code]

[code, code, ..., code]

➡ decoder
Use the dictionary to decode the codes into their associated characters.

➡ Final string
"String1"
"StringN"

```
Example
decoder = { 80:'H', 14:'o', 101:'e', 93:'l'}

code list                  decoded by dictionary          final string
[80, 101, 93, 93, 14]    -> 'H', 'e', 'l', 'l', 'o'     ->   "Hello"
```

Task 2 is to decode all three messages and print them out. In the following, I show the results of decoding messages: msg1, msg2. *I'm not showing msg3, so you can find out what it is on your own. Your output needs to have all three strings.*

You have complete freedom to implement this task in anyway you want as long as the output is correct. You may introduce your own functions. Before printing each message, print the message ID (i.e., 1, 2, or 3) above it as shown below:

```
------ Task 2 ------
--- Message 1 ----
Python
is
cool.
--- Message 2 ----
   ,-.
 _(*_*)_
(_  o  _)
  / o \
 (_/ \_)
--- Message 3 ----
```

(message 3 not show)

Example for msg1
```
msg1 = [[80, 121, 116, 104, 111, 110],
[105, 115], [99, 111, 111, 108, 46]]
```

`msg1` variable is a list of 3 lists.
Each list is decoded with the help of the **decoder** dictionary to produce three strings as described above.

Print each string on its own line to get the final decoded message.

3. GRADING SCHEME (Maximum number of points possible 10)

To get full marks you need to make sure you follow the instructions correctly. The following will be our grading scheme for the Lab components specified in Section 2 of this document.

Task 0: (0 points, but deduction if you skip this part)

- Filename **must** be "lab6.py" (all lowercase, no spaces)
- The Python comments at the beginning of your program **must** include your name, email, and York student id (this is important for grading)
- *If your file name is incorrect, or you do not put in the required information, we will deduct -5 points (Why are we so harsh? Because if you don't put in your name and student id, it can be very difficult for the TAs to determine whose submission this is.)*

Main Task :

- 2 Tasks [-5 points for each that doesn't work properly]
- You can't receive below a 0.

-No submission – 0 points
-Any submission up to one week after the due date is 50% off the total marks
-Any submission after one week of the due date will not be marked and treated as no submission.

See pages below on how to submit your lab code.

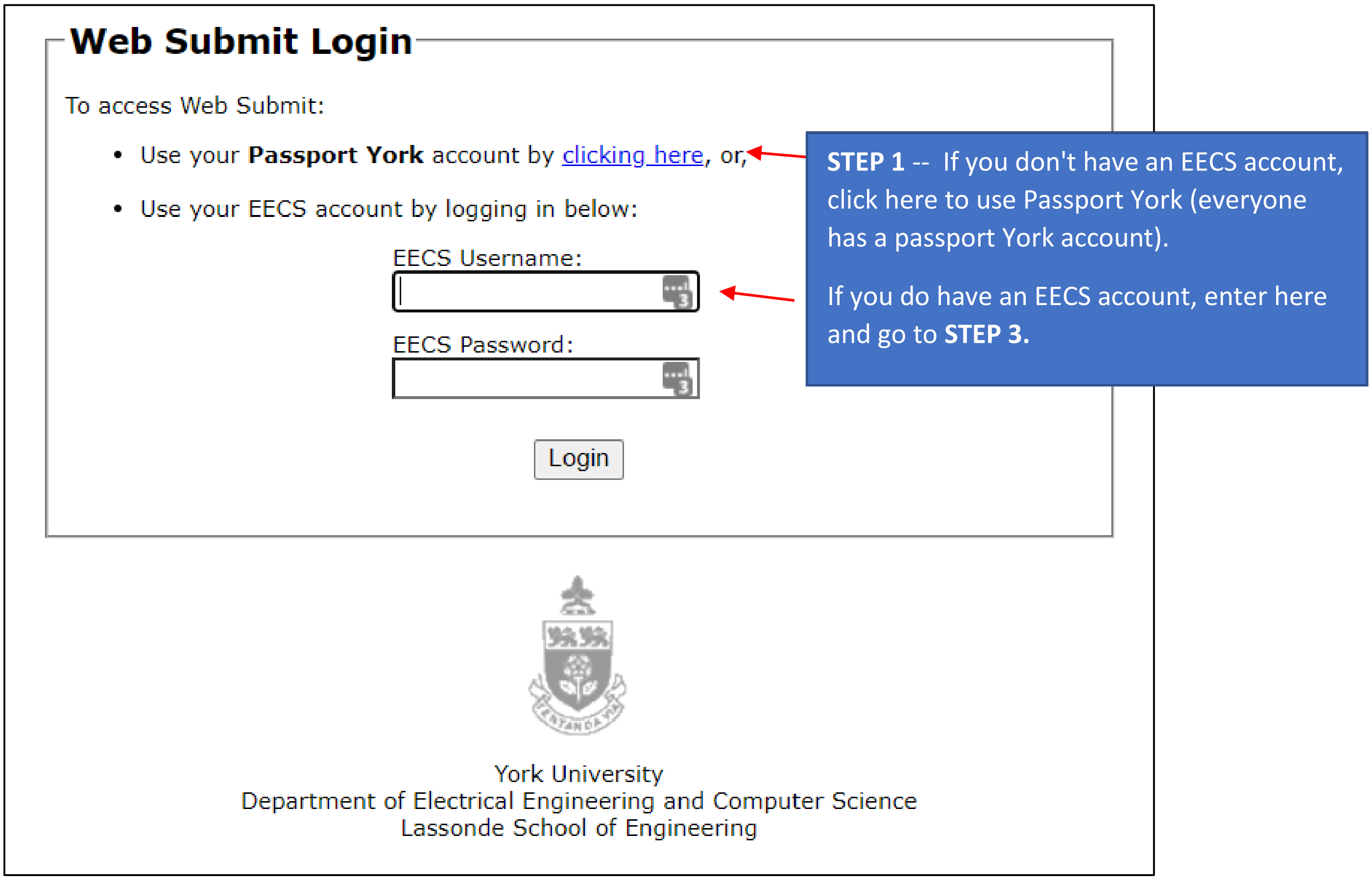
MAKE SURE TO SELECT **Lab6** with websubmit

Note, if you use the new experimental testing platform it can perform websubmit for you!

4. SUBMISSIONS (EECS web-submit)

You will submit your lab using the EECS web submit.

Click on the following URL: https://webapp.eecs.yorku.ca/submit



Web Submit Login

To access Web Submit:

- Use your **Passport York** account by clicking here, or,
- Use your EECS account by logging in below:

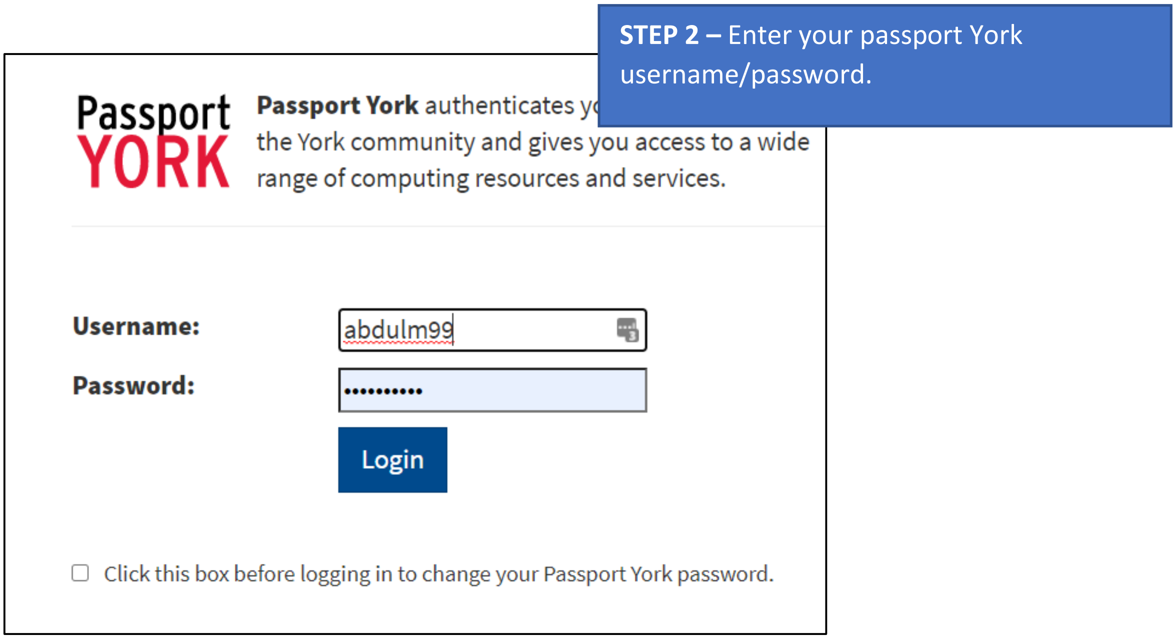
EECS Username:

EECS Password:

York University
Department of Electrical Engineering and Computer Science
Lassonde School of Engineering

STEP 1 -- If you don't have an EECS account, click here to use Passport York (everyone has a passport York account).

If you do have an EECS account, enter here and go to **STEP 3.**



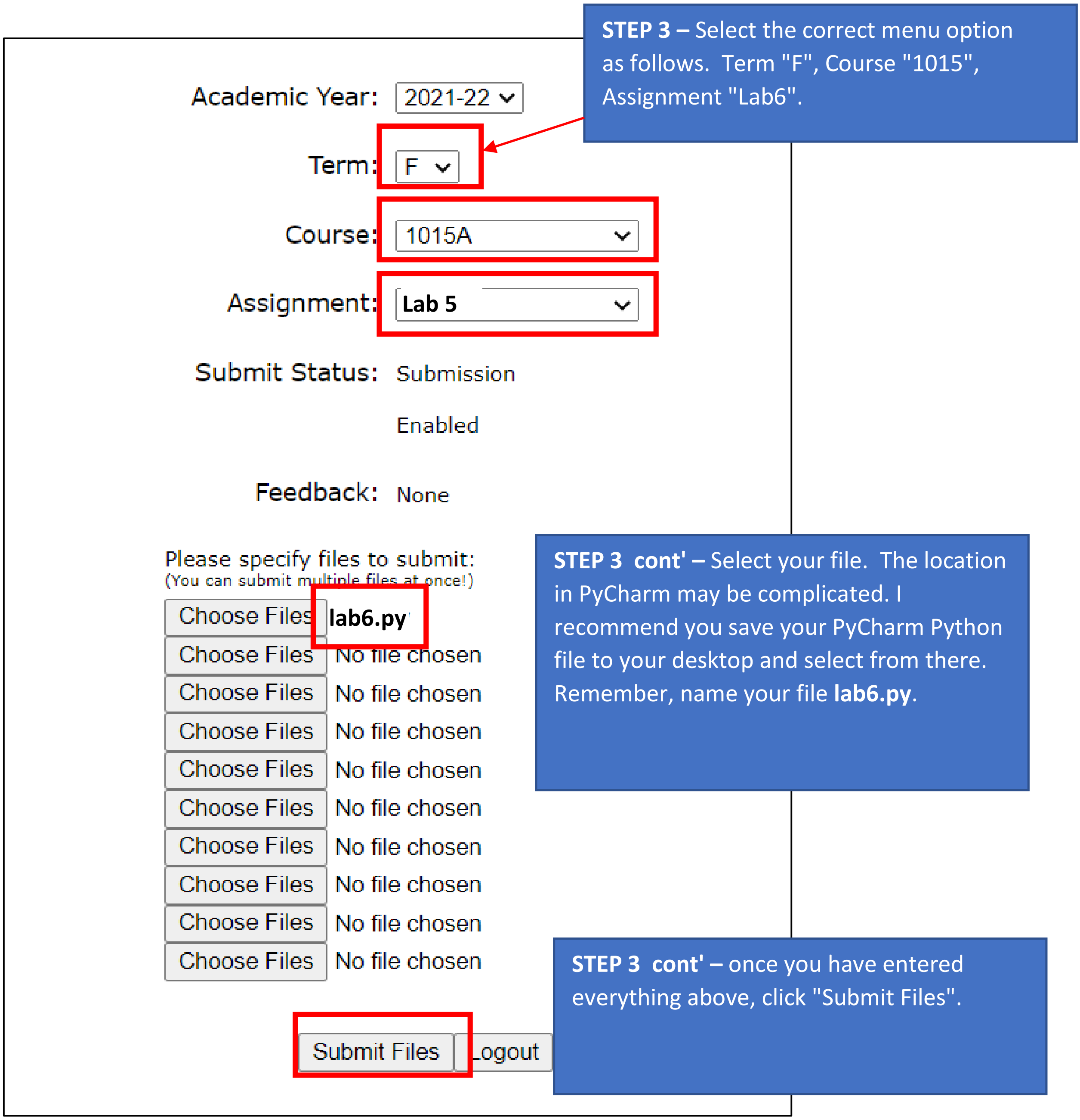
STEP 2 – Enter your passport York username/password.

Passport York authenticates y the York community and gives you access to a wide range of computing resources and services.

Username: abdulm99

Password: ••••••••••

□ Click this box before logging in to change your Passport York password.



Academic Year: 2021-22

Term: F

Course: 1015A

Assignment: Lab 5

Submit Status: Submission Enabled

Feedback: None

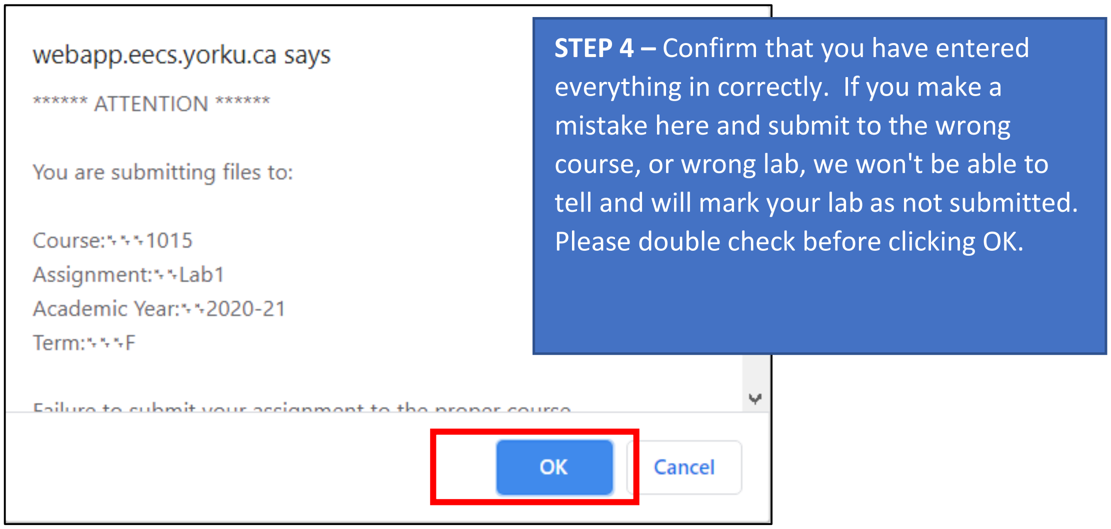
Please specify files to submit:
(You can submit multiple files at once!)

Choose Files lab6.py
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Submit Files
Logout

STEP 3 – Select the correct menu option as follows. Term "F", Course "1015", Assignment "Lab6".

STEP 3 cont' – Select your file. The location in PyCharm may be complicated. I recommend you save your PyCharm Python file to your desktop and select from there. Remember, name your file **lab6.py**.

STEP 3 cont' – once you have entered everything above, click "Submit Files".



webapp.eecs.yorku.ca says

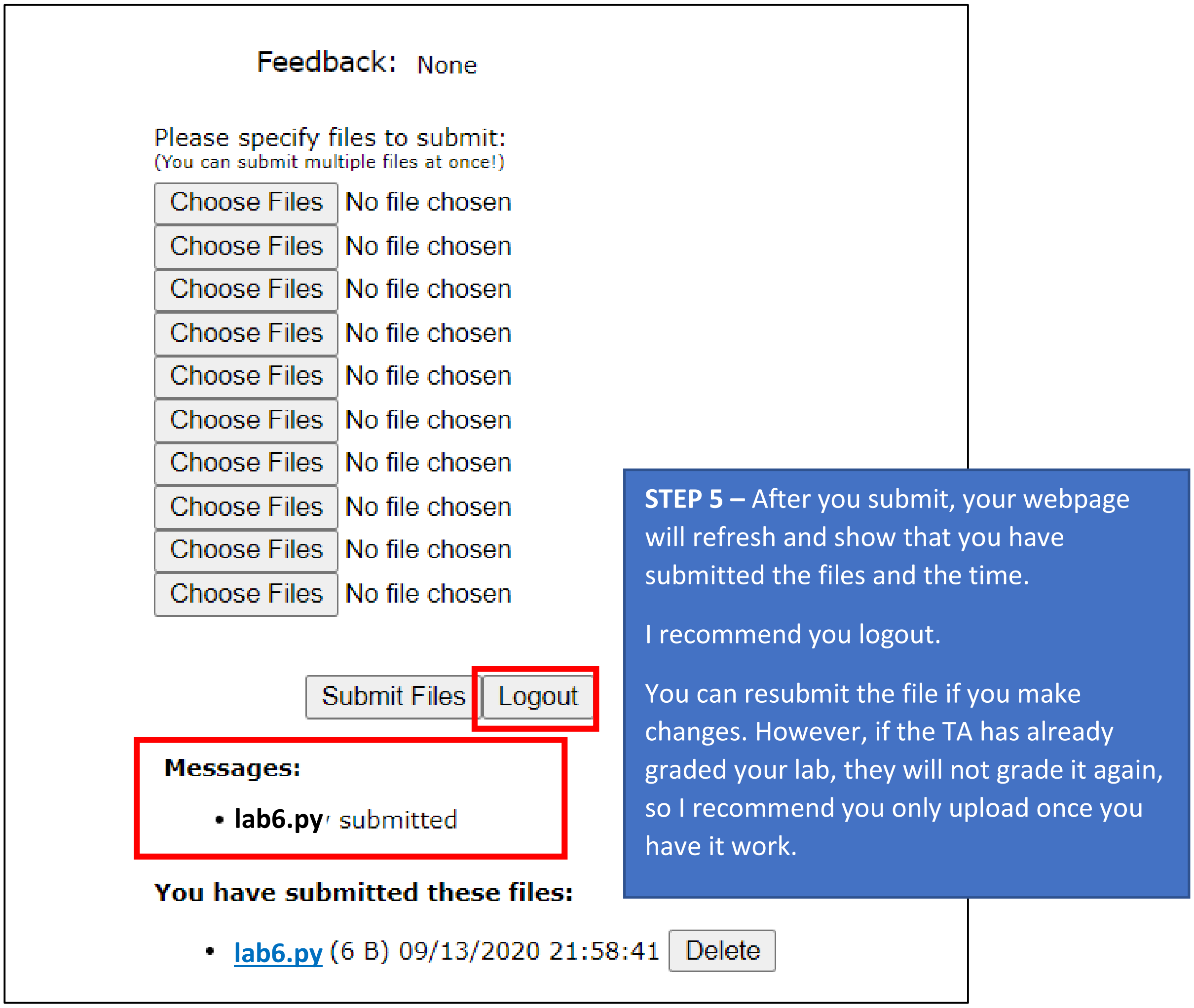
****** ATTENTION ******

You are submitting files to:

Course: 1015
Assignment: Lab1
Academic Year: 2020-21
Term: F

Failure to submit your assignment to the proper course

STEP 4 – Confirm that you have entered everything in correctly. If you make a mistake here and submit to the wrong course, or wrong lab, we won't be able to tell and will mark your lab as not submitted. Please double check before clicking OK.



Feedback: None

Please specify files to submit:
(You can submit multiple files at once!)

Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen
Choose Files No file chosen

Messages:

- **lab6.py**' submitted

You have submitted these files:

- lab6.py (6 B) 09/13/2020 21:58:41

STEP 5 – After you submit, your webpage will refresh and show that you have submitted the files and the time.

I recommend you logout.

You can resubmit the file if you make changes. However, if the TA has already graded your lab, they will not grade it again, so I recommend you only upload once you have it work.

For more details on *websubmit*, see EECS department instructions:

https://wiki.eecs.yorku.ca/dept/tdb/services:submit:websubmit