**Outline**

* tbd

**Objectives**

* tbd

**Materials**

* tbd

**Level 1: Reading a Text File**

1. Open a new Python Repl and run the following program.

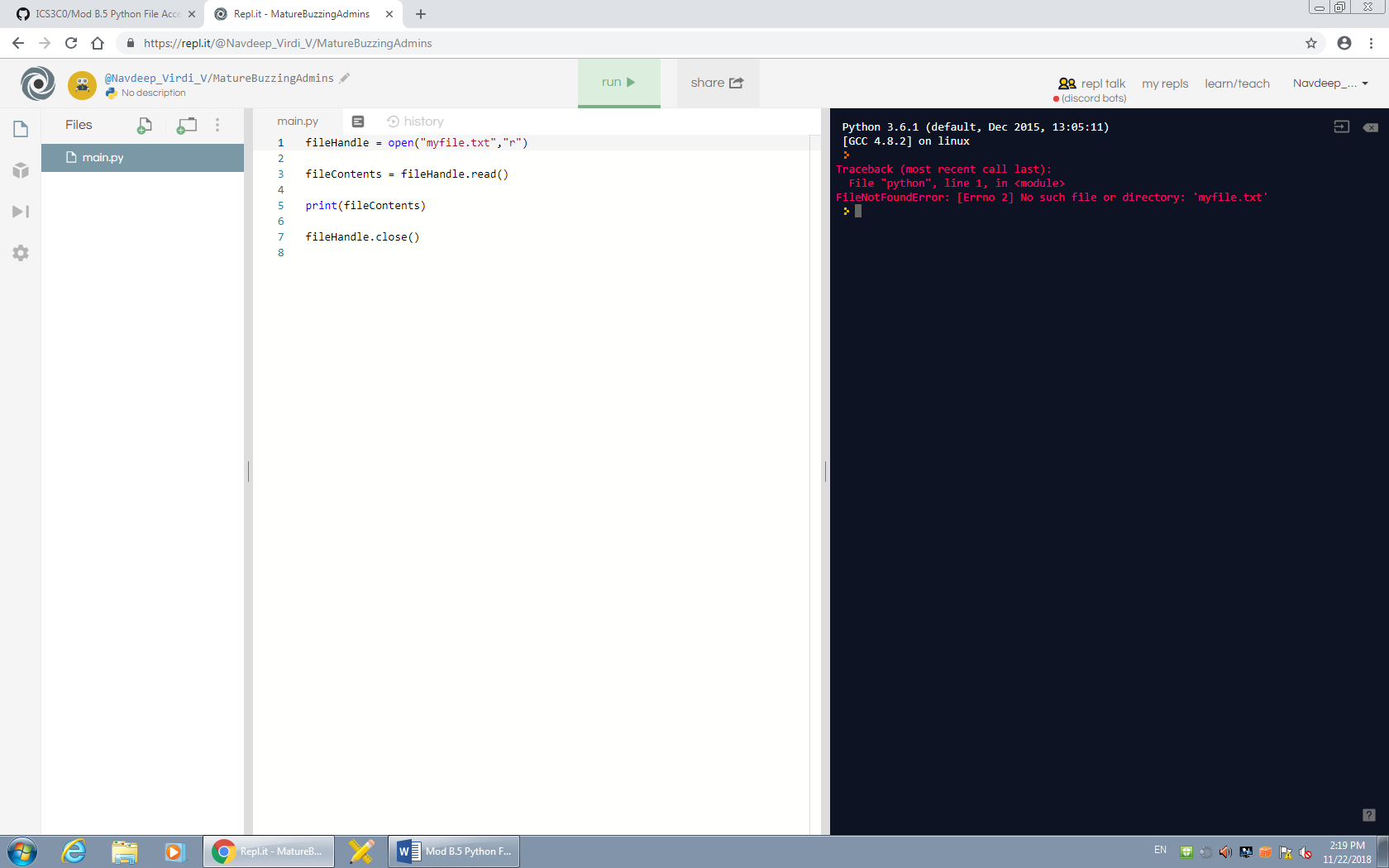
fileHandle = open("myfile.txt","r")

fileContents = fileHandle.read()

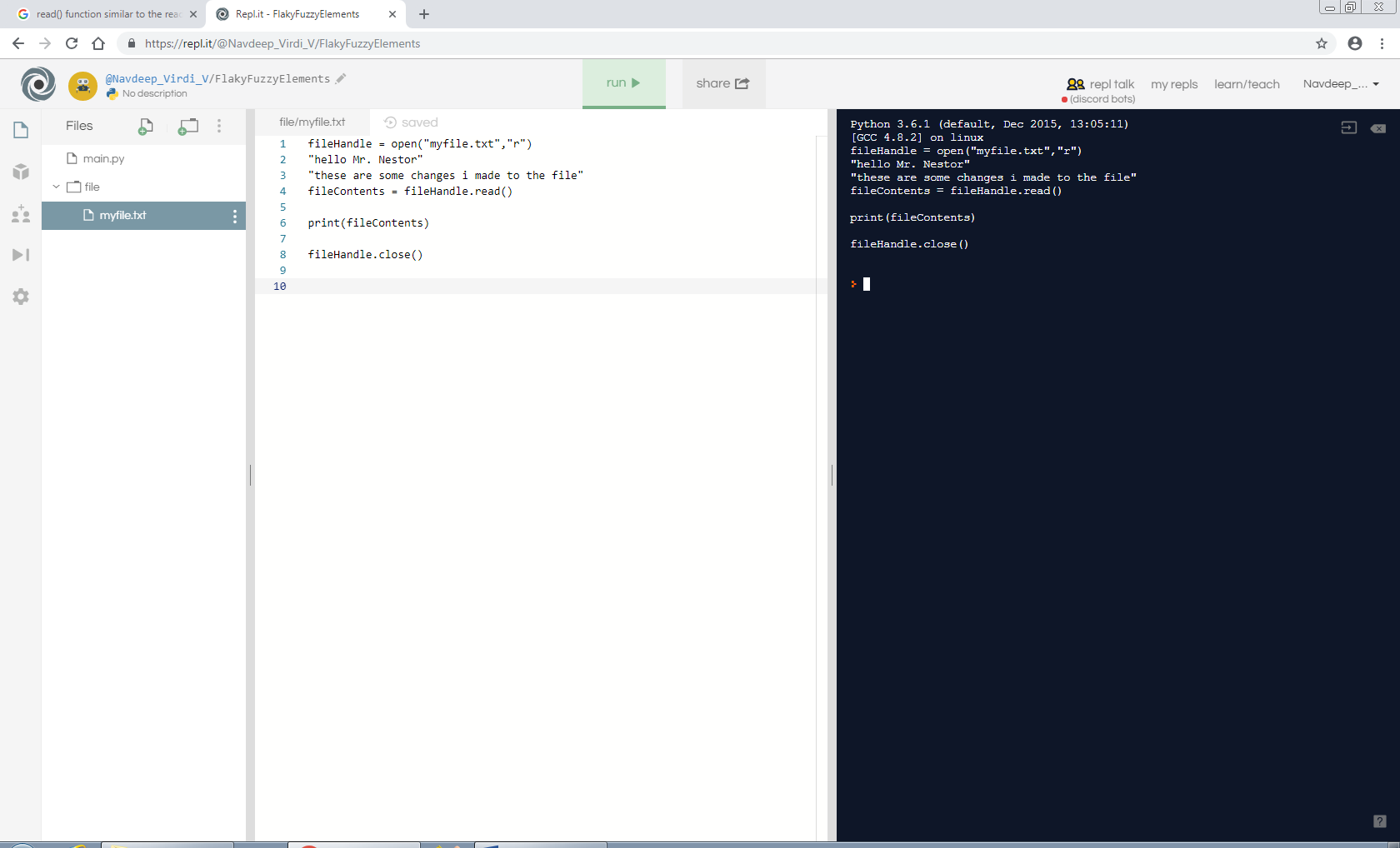
print(fileContents)

fileHandle.close()

1. Why does this program produce a run-time error?

It produce a run time error because the file is not found which means there’s nothing for it to open.  


1. Add a text file to your project as follows:
   * Click on “Add File” icon in the files pane/window.
   * Type “myfile.txt” and return.
   * “myfile.txt” is now open in the editor pane/window.
   * Type some text into “myfile.txt”
   * Make sure to add several lines of text
2. Switch back to main.py pile and run the program. What gets printed out? Explain the result.



1. Load and run the following program.

fileHandle = open("myfile.txt","r")

line = fileHandle.readline()

count = 1

while line :

print("Line ", count, " : ",line.strip())

line = fileHandle.readline()

count += 1

fileHandle.close()

1. Compare and contrast the output of the first and second program
   1. How is the read() function similar to the readline() function?

read() and readline() both reads and prints out text

* 1. How is the read() function different from the readline() function?

readline() reads a small amount of text while the other (“read()”) reads the whole thing

1. Research the Python open() function for file I/O (input / output).
   1. How do you specify which file to open?

fileHandle = open("file/myfile.txt","r") by changing the highlighted part to the file you have named it.

* 1. Modify the program to open a different file.

fileHandle = open("file/myfile.txt","r")

* 1. How would you open a file in a sub-directory?

By opening up the folder

* 1. Modify the program to open a file in a sub-directory.

fileHandle = open("file/myfile.txt","r")

fileContents = fileHandle.read()

print(fileContents)

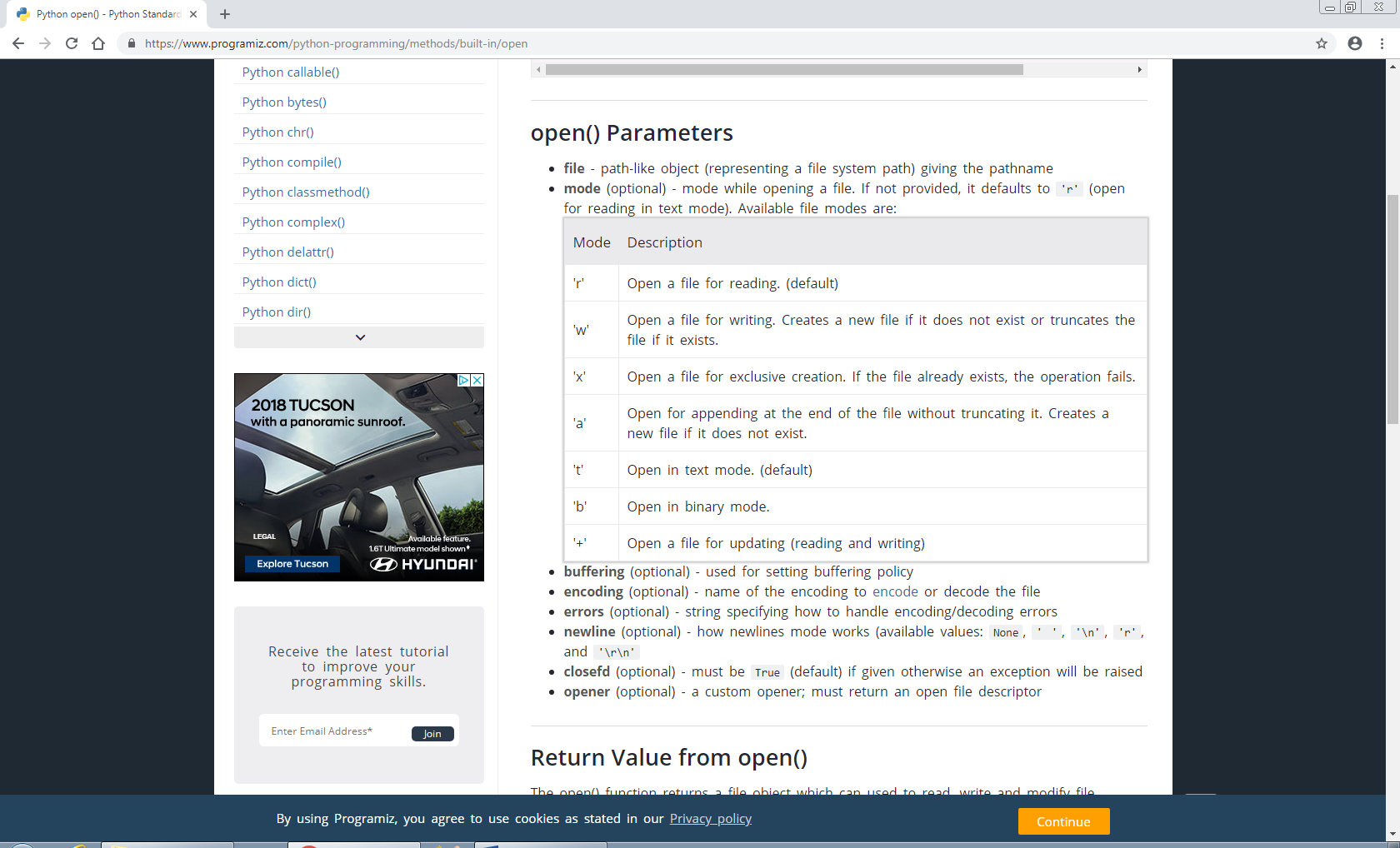
fileHandle.close()

1. Research the Python open() function for file I/O (input / output).
   1. What does the file mode “r” mean?

The file mode “r” means that it is a reading file

* 1. What other file modes can be used? List and explain their meanings.

There is a r and w (these are some modes that I’m aware of)



**Level 2: Writing a Text File**

fileHandle = open("myfile.txt","r")

**Level 3: Pixels & RGB**

import turtle

myPen = turtle.Turtle()

# These variables track the position of the turtle pen

posX = 0

posY = 0

# These variables define the image information.

# Each pixel in the image has a (r,g,b) value

# The complete image is simply a list of pixels

pixelAddress = 0

pixelMemory = [

(10,0,0),(20,0,0),(30,0,0),(40,0,0),(50,0,0),(60,0,0),(70,0,0),(80,0,0),(90,0,0),(100,0,0),(110,0,0),(120,0,0),(130,0,0),

(140,0,0),(150,0,0),(160,0,0),(170,0,0),(180,0,0),(190,0,0),(200,0,0),(210,0,0),

(220,0,0),(230,0,0),(240,0,0),(250,0,0),(260,0,0),(270,0,0),(280,0,0),(290,0,0),

(300,0,0),(290,0,0),(280,0,0),(270,0,0),(260,0,0),(250,0,0),(240,0,0),(230,0,0),

(220,0,0),(210,0,0),(200,0,0),(190,0,0),(180,0,0),(170,0,0),(160,0,0),(150,0,0),

(140,0,0),(130,0,0),(120,0,0),(110,0,0),(100,0,0),(90,0,0),(80,0,0),(70,0,0),

(60,0,0),(50,0,0),(40,0,0),(30,0,0),(20,0,0),(10,0,0),(20,0,0),(30,0,0),

(40,0,0),(50,0,0),(60,0,0)

]

# This user defined function draws a single image pixel

def drawPixel(rgb) :

global posX

myPen.down()

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(18)

posX = posX + 18

# This user defined function starts a new row of pixels

def newRow() :

global posX

global posY

myPen.up()

myPen.left(180)

myPen.forward(posX)

myPen.left(90)

myPen.forward(18)

myPen.left(90)

myPen.down()

posX = 0

posY = posY + 18

# THE MAIN PROGRAM CODE STARTS HERE

#

# Draw eight rows of the image.

# Each row contains eight pixels

for row in range (8) :

for column in range(8) :

drawPixel(pixelMemory[pixelAddress])

pixelAddress += 1

newRow()