```
# tutorial wk 4
# files and exceptions
# author: B. Schoen-Phelan
# date: 14 Oct 2020
# additional files: simpleTextFile.txt, hurricanes.csv, dummy.pdf, a.out
# classes, method vs function, self, exceptions and files (and debugger)


# class myTutorial:
#     def __init__(self): # self gives us a handle or a
pointer to the class object itself
#         print('hello tutorial')
#         a = 3 + 5
#         print(a)
#
#
# mT = myTutorial()

# def my_calculation(): # notice no self in here
#    print('hi tutorial')
#    a = 3+5
#    print(a)
#
# my_calculation()

class my_Tutorial:
  def __init__(self):
     # pass
     self.a = 0

  def my_calculation(self):
     self.a = 3+5
     # a = 7 # this is a different a that is valid just
inside this method
     # print(a)

mT = my_Tutorial()
```

```
mT.my_calculation()
print(mT.a)
class myTutorial:
    def __init__(self): # self gives us a handle or a pointer to the class object itself
        print('hello tutorial')
        a = 3 + 5
        b = 1
        if a > b: # the error lies here, so use the debugger with step over and check the values and the program flow
            print(a, " is smaller than ", b)
        else:
            print(a, " is smaller than ", b)



mT = myTutorial()


# simple open and read to screen
# fo = open("simpleTextFile.txt")
# print(fo.read())
# print(fo.read(2))
# print(fo.readline())
# print(fo.readlines())
# fo.close()

# explicit looping through file
# fo = open("simpleTextFile.txt")
# for line in fo:
    # print(line)

    # with split word
    # word = line.split()
    # print(word)
# fo.close()

# with open("simpleTextFile.txt", "r") as my_file:
#     print(my_file.read())

# different write options
# with open("mynewFile.txt", "w") as my_file:
  # w clears file content
  # my_file.write("this is my new text file")
  # my_file.write("I like cake")

  # then with new line
  # my_file.write("this is my new text file \n")
  # my_file.write("I like cake")

# with open("mynewFile.txt", "a") as my_file:
#     my_file.write("this is a new line") #appends right at the end of a character

# with open("mynile.txt", "r+") as my_file:
#     # throws error message
#     my_file.write("this is a new line")

# with open("mynewFile.txt", "r+") as my_file:
  # throws error message
  # my_file.write("\nthis is a new line again")

# with open("mynewFile.txt", "w+") as my_file:
  # throws error message
  # my_file.write("THIS IS COMPLETELY NEW CONTENT")

# with open("mynewFile.txt", "a+") as my_file:
#     # throws error message
#     my_file.write("\nAdding new stuff")

# try a different type of file
# try a csv file
```

```
with open("hurricanes.csv", "w+") as my_file: #try first just with a+ then
difference to r+, then w+
    # throws error message
    print(my_file.read())
    my_file.write("\"Jan20\", 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11")
```

```
# with open("dummy.pdf", "rb") as my_file: #try first just with r
    # throws error message
    # print(my_file.read())
```

```
# try:
#     fo = open("simpleTextFile.txt", "r") # try with
s.txt for a no file
# except Exception as e:
#     print(e)
# else: # else is called  if there aren't any
exceptions
#     print(fo.readline())
#     fo.close()

# example with a file that exists but is in a wrong
format
# try:
#     fo = open("a.out", "r")
# except Exception as e:
#     print(e)
# else: # else is called  if there aren't any
exceptions
#     try:
#         print(fo.readline())
#     except Exception as e:
#         print(e)
#     fo.close()
# files and exceptions
```