**Open File in Read Mode**

obj\_file = open('ToDo.txt', 'r')

for row in obj\_file:

row = row.split(',')

dic = {row[0]: row[1].rstrip('\n')}

lst\_table.append(dic)

obj\_file.close()

1. The dictionary is missing keys for the task and priority. The values are what are being pulled from the text file. See the example below. Same thing for the add\_item function

dicRow = {"Task":strData.split(",")[0],"Priority":strData.split(",")[1]}

1. Also I recommend making this code a function and making it execute prior to the while loop. The first thing your program would do is read the text and converts the text to the dictionary/table row

**Save Function**

def save\_data():

""" This function saves the current table of dictionaries to ToDo.txt file. """

obj\_file = open('C:\\\_PythonClass\\Mod5\\ToDo.txt', 'w')

for dic in lst\_table:

str\_dic = str(dic)

str\_dic = str\_dic.replace('{', '')

str\_dic = str\_dic.replace('}', '')

str\_dic = str\_dic.replace('\'', '')

str\_dic = str\_dic.replace(':', ',')

str\_dic = str\_dic.replace(', ', ',')

obj\_file.write(str\_dic + '\n')

obj\_file.close()

print('Data saved to ToDo.txt')

1. This code can be simplified with a ‘for’ loop. See below. tblData is your table of dictionary rows

file = open("C:\\_PythonClass\ToDo.txt","w")

#pull values from the dictionary and turn them into a list; save to file

for x in tblData:

x.values()

this\_list = list(x.values())

file.write(this\_list[0] + "," + this\_list[1])

file.close()