Charles Tung

May 25, 2021

Student ID: 9721184

Foundations of Programming, Python

Assignment 07

https://github.com/charleslautung/IntroToPython\_Module07/

# Assingment07.py

## Introduction

Homework Assignment07 of the Foundations of Programming, Python class involved creating a Assignment07.py script. Assignment07.py script demonstrates storing data in a python pickle binary file and the ability of python to use Error Handling. The script reads in an existing binary python pickle file AppData.dat and appends to it with user input from the keyboard. The ID and the Name are entered at the keyboard.

## Show Current Data

The below screen shot in figure 1 show the current data being shown from AppData.dat):

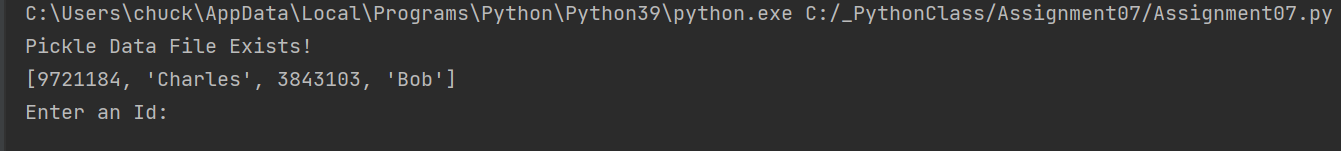


Figure 1: Screen Shot of PyCharm Showing the Current Data of AppData.dat

## Add a New Item

The below screen shot in figure 2 shows the additional ID (5338883) and Name (Ruzann) being added. The additional ID and Name are being added to the AppData.dat as printed below:

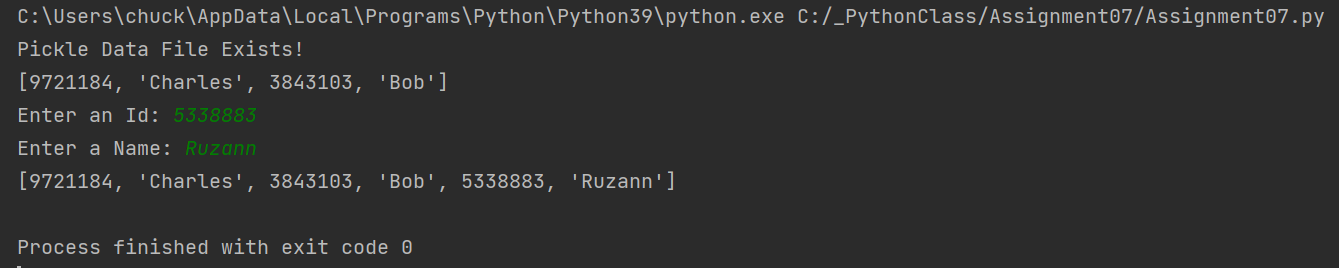


Figure 2: Screen Shot of PyCharm Showing the New Item Being Added

Figure 3 below shows the AppData.dat being read into the script when the script is started again and the contents printed out. Note the additional item of ID: 5338883 and Name: Ruzann was printed:

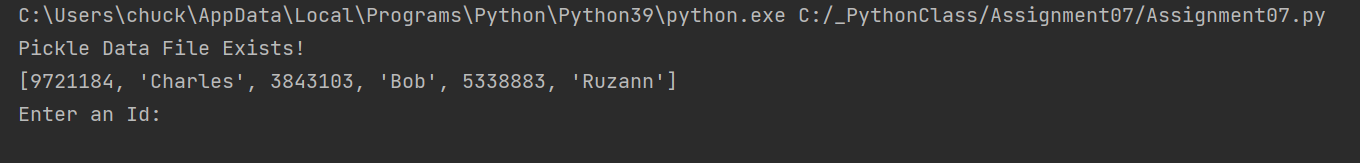


Figure 3: Screen Shot of PyCharm showing the new items in the AppData.dat

## Error Handling

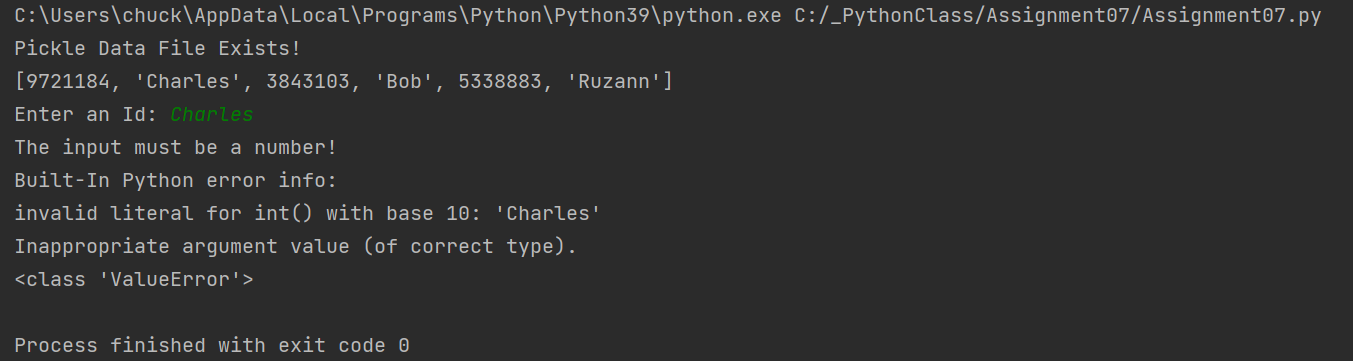
The below figure shows an example of the error handling that was added. Below the incorrect data type is entered into the ID: 

Figure 4: Showing the Error Handling when the Incorrect Data Type is Entered into ID

## Windows Command Screen Output

Below shows the screen shot (figure 5) from the windows command screen showing that the identical image from the PyCharm screenshot in Figure 3 on the previous page:

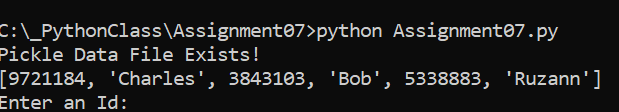


Figure 5: Windows Command Output of Assigment07.py Script

## AppData.dat Binary File

Below shows the notepad screenshot of the python pickle file binary file AppData.dat in figure 6 below:

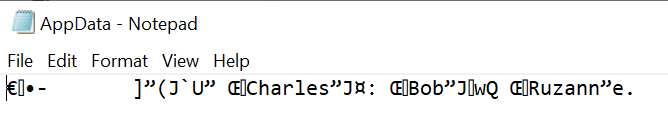


Figure 6: Notepad Screen Shot of AppData.dat

## Source Code

The source code for Assigment07.py was as shown in the PyCharm terminal as shown below in figure 7:

# ------------------------------------------------- #  
# Title: Assignment07  
# Description: Example of storing data in a binary file with Error Handling  
# ChangeLog: (Who, When, What)  
# CTUNG,<5.25.2021>,Created Scripted to read in pickle file AppData.dat  
# CTUNG,<5.25.2021>,Created function save\_data\_to\_file() to save data to AppData.dat  
# CTUNG,<5.25.2021>,Created function read\_data\_from\_file() read data tp AppData.dat  
# CTUNG,<5.25.2021>,Added Error Handling to read\_data\_from\_file()  
# CTUNG,<5.25.2021>,Added user input for lstCustomer and intID with error handling  
# CTUNG,<5.25.2021>,Added data to be appended to list lstCustomer[]  
# ------------------------------------------------- #  
import pickle # This imports code from another code file!  
import os.path # This imports code used to check if file exists  
  
# Data -------------------------------------------- #  
strFileName = 'AppData.dat'  
lstCustomer = []  
  
# Processing -------------------------------------- #  
def save\_data\_to\_file(file\_name, list\_of\_data):  
 *""" saves pickle file* ***:param*** *file\_name: (string) file name of pickle file* ***:param*** *list\_of\_data: (list) list of data* ***:return****: none  
 """* file = open(file\_name, "wb")  
 pickle.dump(list\_of\_data, file)  
 file.close()  
  
def read\_data\_from\_file(file\_name):  
 *"""Reopen and unpickle the pickled content and read to obj* ***:param*** *file\_name: (string) with name of the pikcle file* ***:return****: list of data  
 """* try:  
 with open(file\_name, "rb") as f:  
 list\_of\_data = pickle.load(f)  
 print(list\_of\_data)  
 f.close()  
 return list\_of\_data  
 except FileNotFoundError as e:  
 print("The pickle file must exist before running this script!")  
 print("Built-In Python error info: ")  
 print(e, e.\_\_doc\_\_, type(e), sep='\n')  
 quit()  
 except Exception as e:  
 print("There was a non-specific error!")  
 print("Built-In Python error info: ")  
 print(e, e.\_\_doc\_\_, type(e), sep='\n')  
 quit()  
  
# Presentation ------------------------------------ #  
if os.path.exists(strFileName):  
 print("Pickle Data File Exists!")  
 # Checks if file exists and reads in data  
 lstCustomer = read\_data\_from\_file(strFileName)  
else:  
 print("Pickle File Does Not Exists!")  
 print("Pickle File Will be Created!")  
  
try:  
 intID = int(input("Enter an Id: "))  
except ValueError as e:  
 print("The input must be a number!")  
 print("Built-In Python error info: ")  
 print(e, e.\_\_doc\_\_, type(e), sep='\n')  
 quit()  
  
try:  
 strName = str(input("Enter a Name: "))  
except Exception as e:  
 print("There was a non-specific error!")  
 print("Built-In Python error info: ")  
 print(e, e.\_\_doc\_\_, type(e), sep='\n')  
 quit()  
  
# Appends New Data to Existing List  
lstCustomer.append(intID)  
lstCustomer.append(strName)  
  
# Store the list object into a binary file  
save\_data\_to\_file(strFileName, lstCustomer)  
  
# Read the data from the file into a new list object and display the contents  
read\_data\_from\_file(strFileName)

Figure 7: Screenshot of Source Code to Assigment07.py Written in PyCharm’s Terminal

The script reads in an existing AppData.dat python pickle file. The import os.path() was used to check if the AppData.dat pickle file existed. One functions were written read\_data\_from\_file() unpacks the existing AppData.dat pickle file and reads it in. The other function save\_data\_from\_file() saves the data to the AppData.dat pickle file.

## Error Handling and Pickling Examples:

I thought this website had a good discussion about the amending to a pickle file that I used to research the method to append to a pickle file. The pickle file needs to be unpack and read in first then saved.

[How to update a pickle file in Python? - GeeksforGeeks](https://www.geeksforgeeks.org/how-to-update-a-pickle-file-in-python/)

This website has a good discussion on Erro Handling that I used to increase my knowledge:

[How to Best Use Try-Except in Python - 9 Tips for Beginners (techbeamers.com)](https://www.techbeamers.com/use-try-except-python/)

## Summary

This paper summarizes some of the Assigment07.py script. The script demonstrated the ability of Python to use read and write a python pickle binary file and error handling. Error handling is used to help the user debug the python. The binary file is an alternative storage to a text file.