Naga Anusha.Perali

February 26, 2022

Python Script Assignment 7

GitHub Repository Link: https://github.com/NagaAusha/ITFnd100-mod7

Exception Handling and Python Pickling

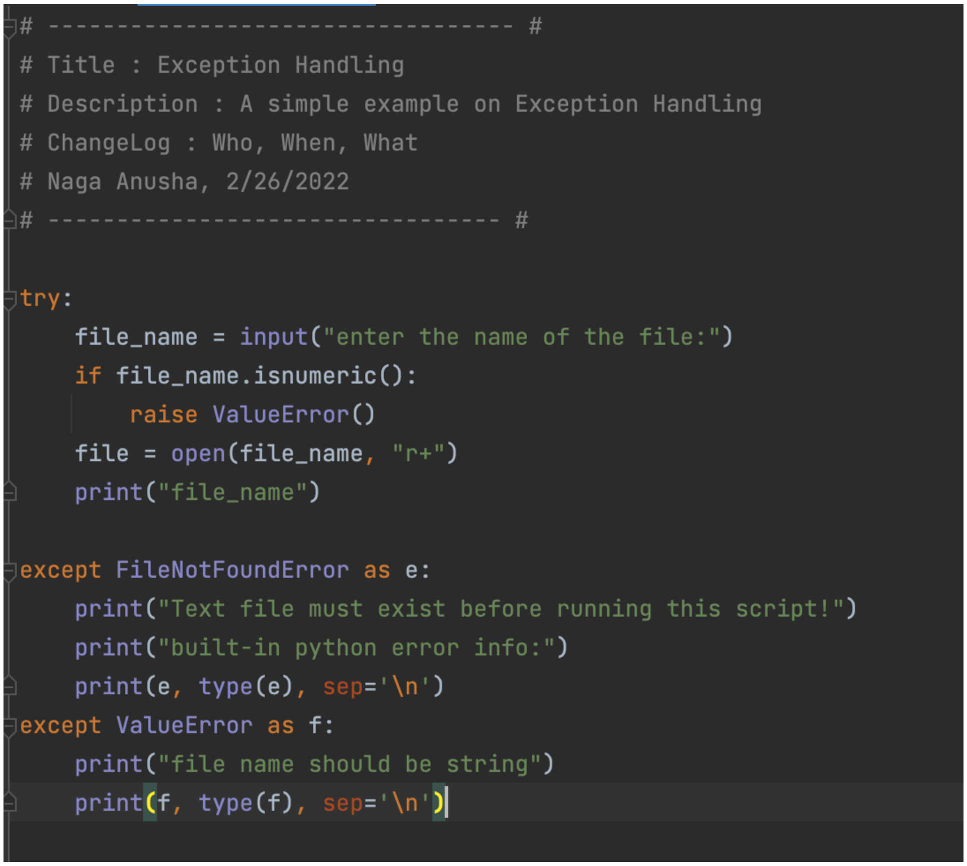
Introduction

In this assignment, I'm going to explain the steps that I have used to create a Python Program where user entered data is saved in a binary file format using Pickling and how to handle exceptions in the python code.

Exception Handling

When we run a program, we might get some errors as a result of that program may break. To continue the flow of the program without breaking we will use "Exception". Python has many built-in exceptions that are raised when your program encounters an error. If the program contains a code that may throw the exception, we must place that code in the **try** block. The **try** block must be followed with the **except** statement, which contains a block of code that will be executed if there is some exception in the try block. Exception might be a single one or multiple below is the example for exception handling. (Picture 1)

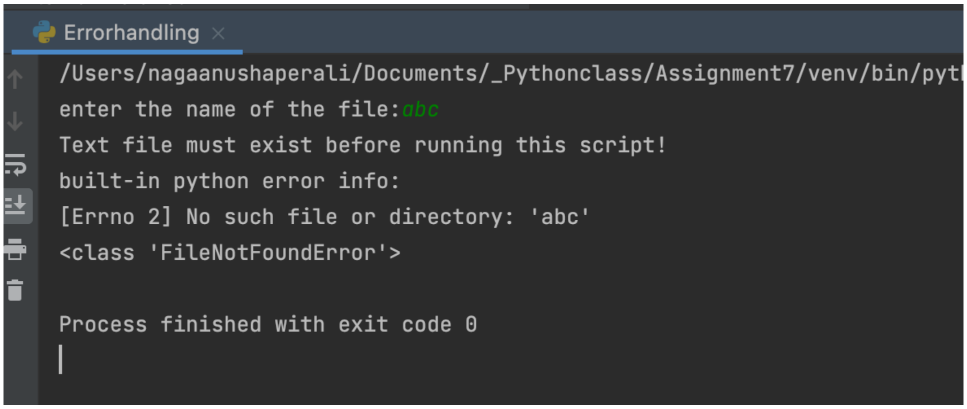
And also, I found this article helpful in the process of understanding how exactly exceptions are used https://www.programiz.com/python-programming/exception-handling (External Link).



```
# ---------------------------------- #
# Title : Exception Handling
# Description : A simple example on Exception Handling
# ChangeLog : Who, When, What
# Naga Anusha, 2/26/2022
# --------------------------------- #

try:
    file_name = input("enter the name of the file:")
    if file_name.isnumeric():
        raise ValueError()
    file = open(file_name, "r+")
    print("file_name")

except FileNotFoundError as e:
    print("Text file must exist before running this script!")
    print("built-in python error info:")
    print(e, type(e), sep='\n')
except ValueError as f:
    print("file name should be string")
    print(f, type(f), sep='\n')
```

Picture 1: Code for Exception handling

In the above program I asked user to enter the name of the file if the entered file is not there then ‘FileNotFound’ exception will be raised, if user enter the name of the file as a number, then ‘Value Error’ Exception will be raised below is the result when I run the program.

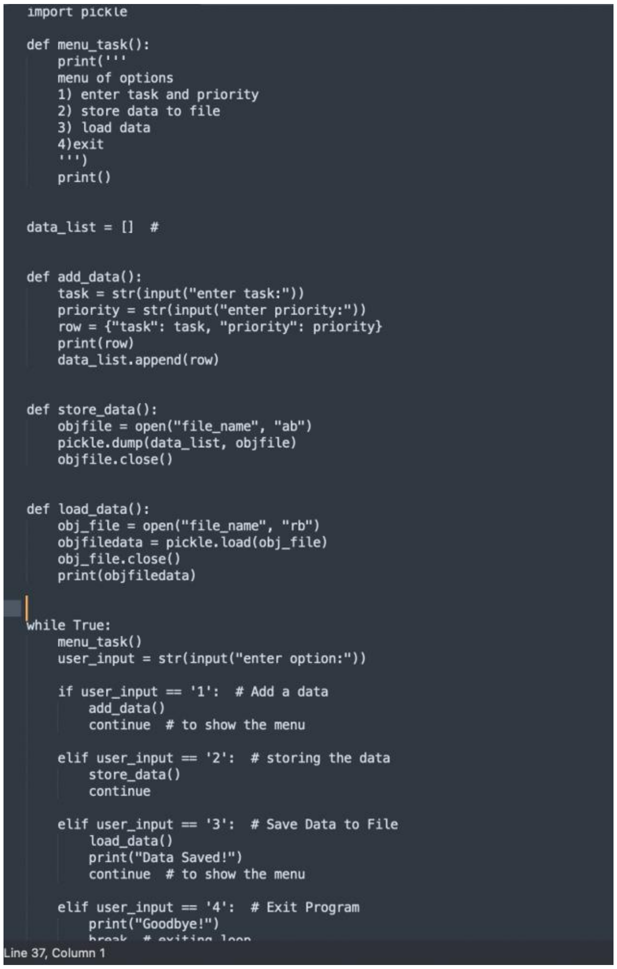


```
/Users/nagaanushaperali/Documents/_Pythonclass/Assignment7/venv/bin/pyt
enter the name of the file:abc
Text file must exist before running this script!
built-in python error info:
[Errno 2] No such file or directory: 'abc'
<class 'FileNotFoundError'>

Process finished with exit code 0
```

Picture 2: FileNotFound error

Python Pickling

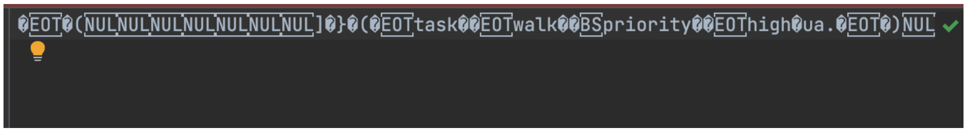
Pickling is used to save the data in to Binary form instead of a plain text to use pickle in your program we need to import it by using “import” keyword. In another words Pickling is a way to convert (list, dict, etc.) into a character stream. I found this article helpful for me to understand more about Pickling and how does that work. https://www.geeksforgeeks.org/understanding-python-pickling-example/ (External Link). Below is the example of pickling



```
import pickle

def menu_task():
    print('''
    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit
    ''')
    print()


data_list = []  #


def add_data():
    task = str(input("enter task:"))
    priority = str(input("enter priority:"))
    row = {"task": task, "priority": priority}
    print(row)
    data_list.append(row)


def store_data():
    objfile = open("file_name", "ab")
    pickle.dump(data_list, objfile)
    objfile.close()


def load_data():
    obj_file = open("file_name", "rb")
    objfiledata = pickle.load(obj_file)
    obj_file.close()
    print(objfiledata)


while True:
    menu_task()
    user_input = str(input("enter option:"))

    if user_input == '1':  # Add a data
        add_data()
        continue  # to show the menu

    elif user_input == '2':  # storing the data
        store_data()
        continue

    elif user_input == '3':  # Save Data to File
        load_data()
        print("Data Saved!")
        continue  # to show the menu

    elif user_input == '4':  # Exit Program
        print("Goodbye!")
        break  # exiting loop
```

Picture 3: Python Pickling.

In the above program I displayed the four menu options to choose and perform the operation such add the data, load the data, save the data and exit the program. I started of importing the pickle that which imports the code from another file, the entered data was captured and then opens up the file open (“file_name”, “ab”) and then use pickle. Load () to load the data. If the user chooses option1 data will be added or continue to choose another option until user exits the program. Check image below for the result.

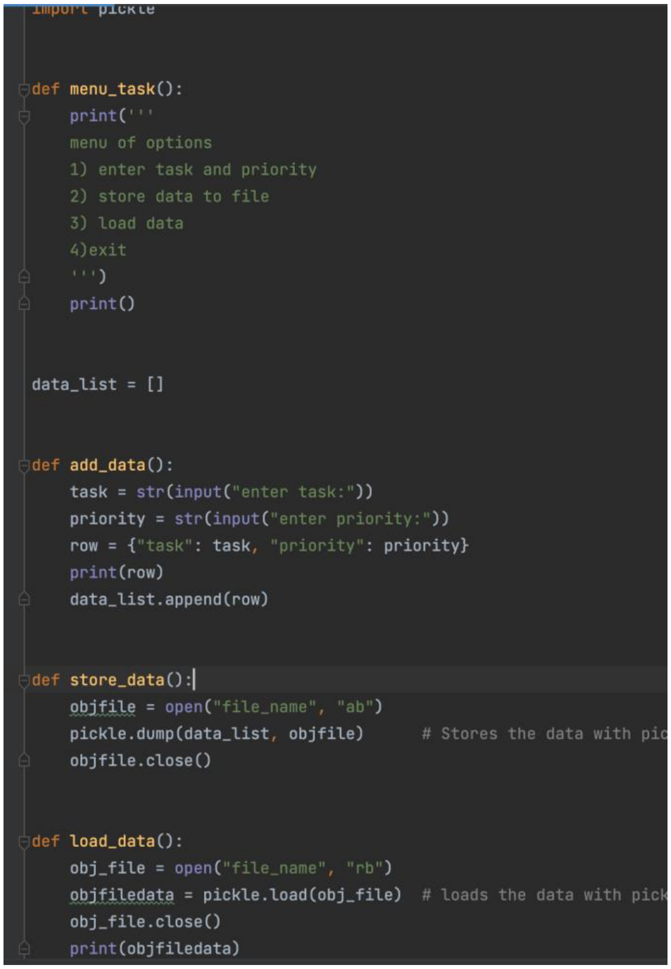


```
�EOT�(NULNULNULNULNULNULNUL]�}�(�EOTtask��EOTwalk��BSpriority��EOThigh�ua.�EOT�)NUL
```

Picture 4: Data store in binary format

Pickling with Error Handling

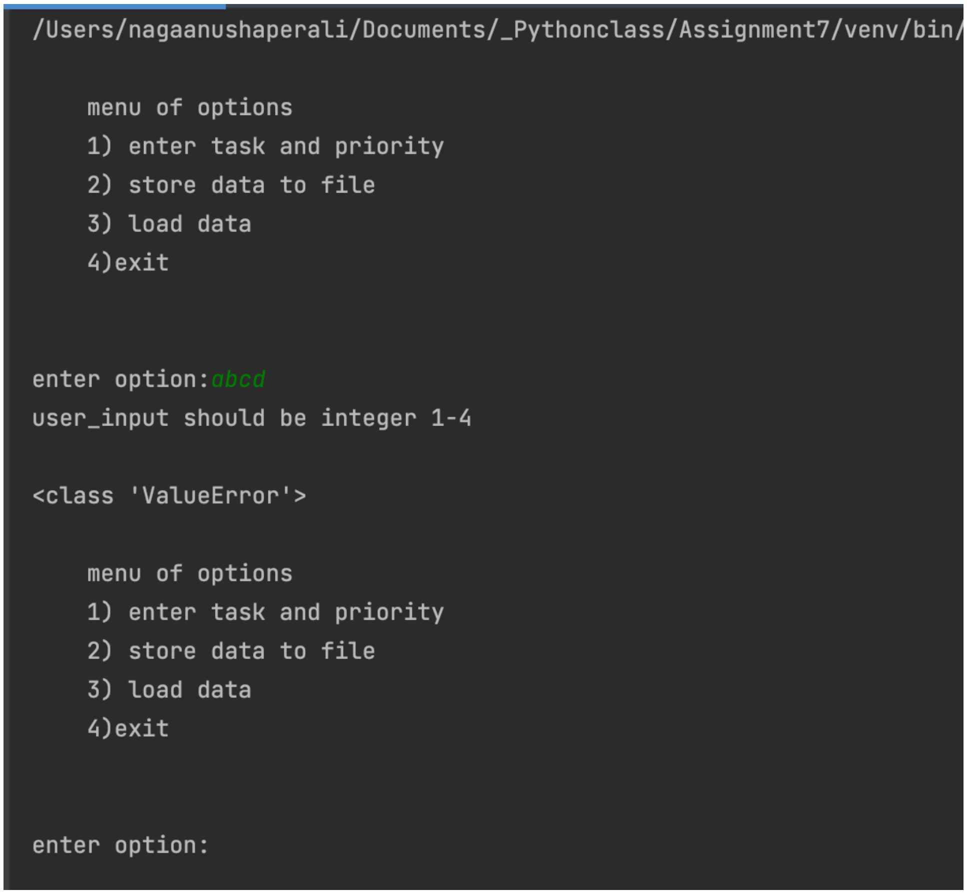
I created a simple program storing the data to a binary file with exception handling here I have thrown a single exception for to keep it simple and like same as the above I ask user to choose one option from the menu of options, here I used while loop to loop within the program until user selects to Exit the program and kept try and catch exception to catch the error and continue through program. When the user entered the data in a numeric format that will be caught “value error” will be raised. check Picture 3 and 5



```
import pickle


def menu_task():
    print('''
    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit
    ''')
    print()


data_list = []


def add_data():
    task = str(input("enter task:"))
    priority = str(input("enter priority:"))
    row = {"task": task, "priority": priority}
    print(row)
    data_list.append(row)


def store_data():
    objfile = open("file_name", "ab")
    pickle.dump(data_list, objfile)      # Stores the data with pic
    objfile.close()


def load_data():
    obj_file = open("file_name", "rb")
    objfiledata = pickle.load(obj_file)  # loads the data with pick
    obj_file.close()
    print(objfiledata)
```

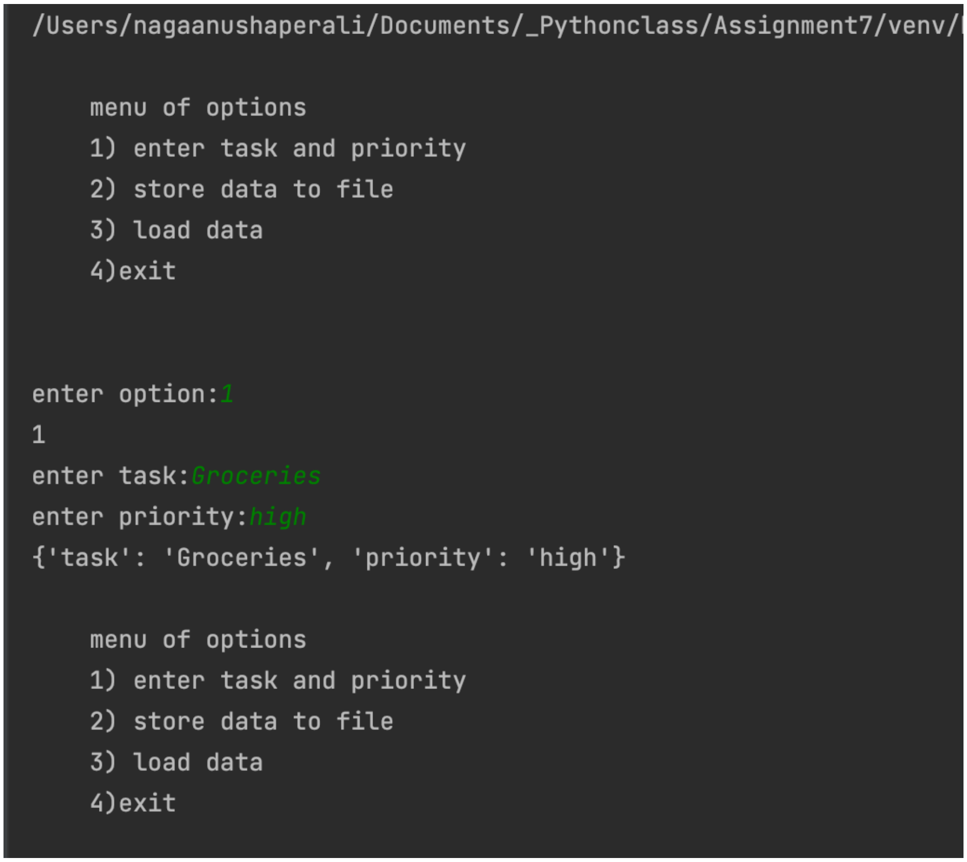
Picture 5: Code for Pickling with error handling.

Below are the Pictures of the result



```
/Users/nagaanushaperali/Documents/_Pythonclass/Assignment7/venv/bin/

    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit


enter option:abcd
user_input should be integer 1-4

<class 'ValueError'>

    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit


enter option:
```

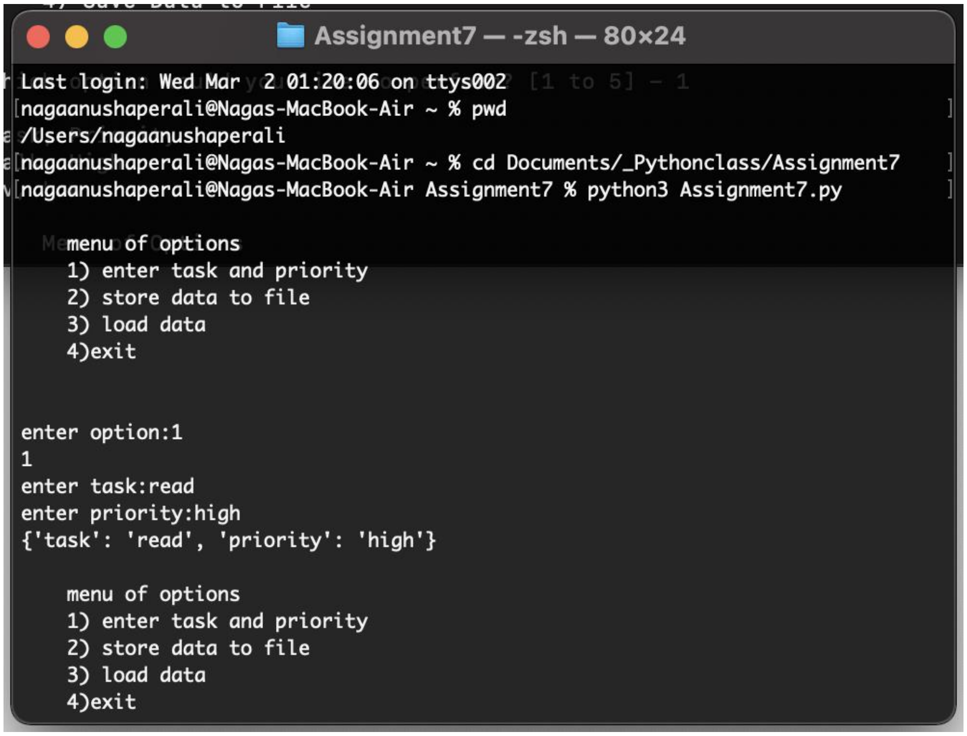
Picture 6: ValueError exception raised.



```
/Users/nagaanushaperali/Documents/_Pythonclass/Assignment7/venv/

    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit


enter option:1
1
enter task:Groceries
enter priority:high
{'task': 'Groceries', 'priority': 'high'}

    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit
```

Picture 7: adds the data to file

I also ran the same program in terminal window for the results refer picture below.



```
Last login: Wed Mar  2 01:20:06 on ttys002
nagaanushaperali@Nagas-MacBook-Air ~ % pwd
/Users/nagaanushaperali
nagaanushaperali@Nagas-MacBook-Air ~ % cd Documents/_Pythonclass/Assignment7
nagaanushaperali@Nagas-MacBook-Air Assignment7 % python3 Assignment7.py

    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit


enter option:1
1
enter task:read
enter priority:high
{'task': 'read', 'priority': 'high'}

    menu of options
    1) enter task and priority
    2) store data to file
    3) load data
    4)exit
```

Picture 8: Result in terminal

Summary

In this Assignment by referring to module 7 video and by referring the information in the text book I learned to work with Error handling and Python Pickling and perform some operations, and I also learn how to Create folder, file and create a markdown file and formatting the page in GitHub.