Hayford Teye

Date: 02/26/23

Course: Foundation of Programming (Python)

Assignment 7

Pickling and Structured Error Handling

Introduction

In this assignment I will explain the steps to create a python script that discuses pickling and structure error handling. Pickling is the process whereby a python object hierarchy is converted into a byte stream and unpickling is the inverse operation whereby a byte stream is converted back into an object hierarchy.

The **try block** lets you test a block of code for errors. **The except block** let you handle the error. The final block lets you execute code, regardless of the results of the try- and except block.

Creating my Scripts

I began creating my script in the PyCharm editor tool, I create a folder in a location I can find my file easily and name my script **Pickling_Handling.py** within the Assignment07 folder. I declare my variables and user prompt in the following form:

```
11  lstPickle = []
12  strGiftFile = "BirthdayGiftList.dmp"
13  strRecipient = input("Enter the gift recipient: ")
14
```

Figure 1: A screenshot of variable declaration from my script

I chose Birthday gift list to demonstrate my pickling. In this script I used the **input()** function to ask the user to enter the name of a gift recipient, age and the idea of the gift which will be giving to the person. The input data is then place into a list which will be pickled into BirthdayGiftList.dmp file.

Import pickle modules was used in the script to transform a complex object into a byte stream and it can transform the byte stream into an object with the same internal structure.

```
9   import pickle  # import pickle modules
10
```

Figure 2: A screenshot of import pickle modules.

The **open()** function opens a file in text format by default. To open a file in binary format, add **'b'** to the mode parameter. The "**rb"** mode opens the file in binary format for reading, while **"wb**" mode opens the file in binary format for writing.

ValueError an exception that occurs when a function receives an argument of the correct data type but an inappropriate value. This error usually occurs in mathematical operations that require a certain kind of value.

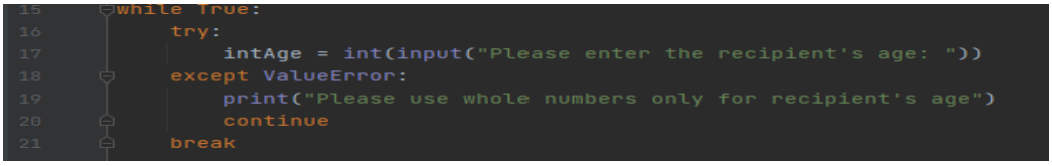

Figure 3: A screenshot of Try – except error handling block.

Final Pickling and Structured Error Handling Script

The script was modified in the PyCharm editing tool, and after my code was completely running to my satisfaction, I updated the changelog and add notes to the script explaining each action performing in every line in my code and verify the binary file is working. Below is a screenshot of my final script running in PyCharm and verifying that the file has data.

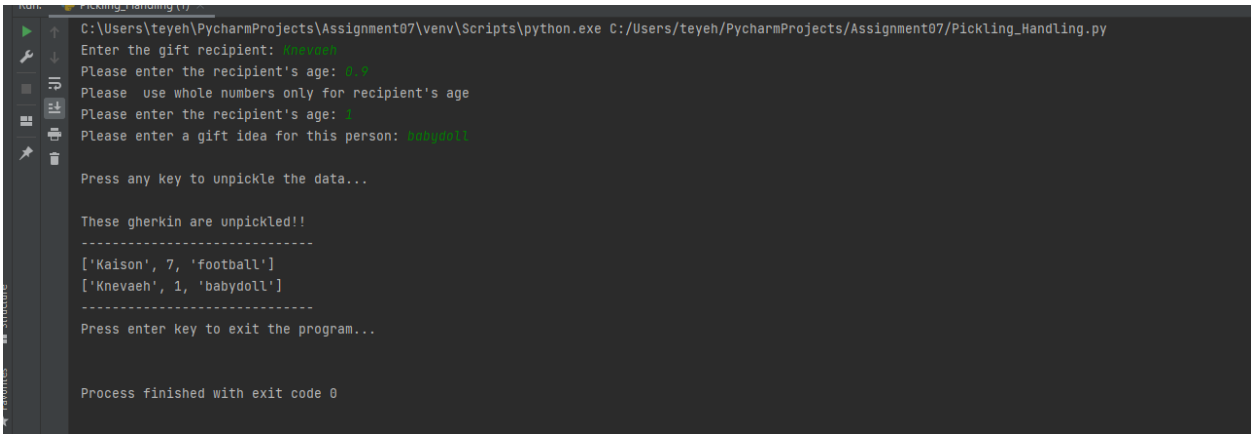

Figure 4: A screenshot of the script running in PyCharm.

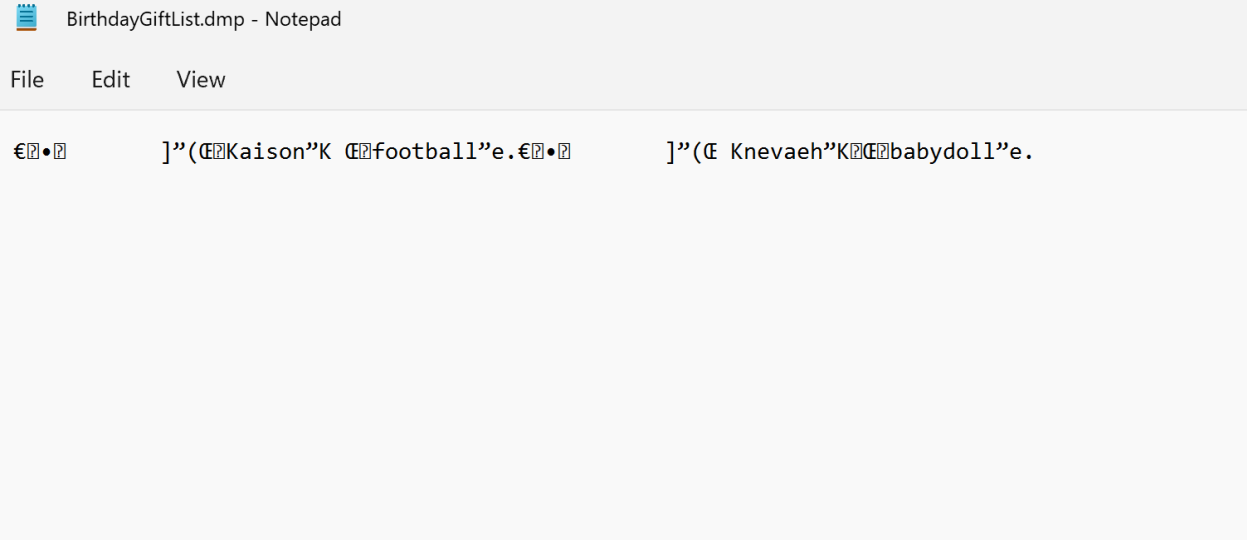

Figure 5: Verifying the dump file.

Running the Script in Command Prompt

I right click on the script file **Pickling_Handling.py** in my Assignment07 folder in the PyCharm IDE and navigate to find the path copy it and open it in command shell to run the script and follow the command prompt to enter user data. Below is screenshot of script running in a command window.

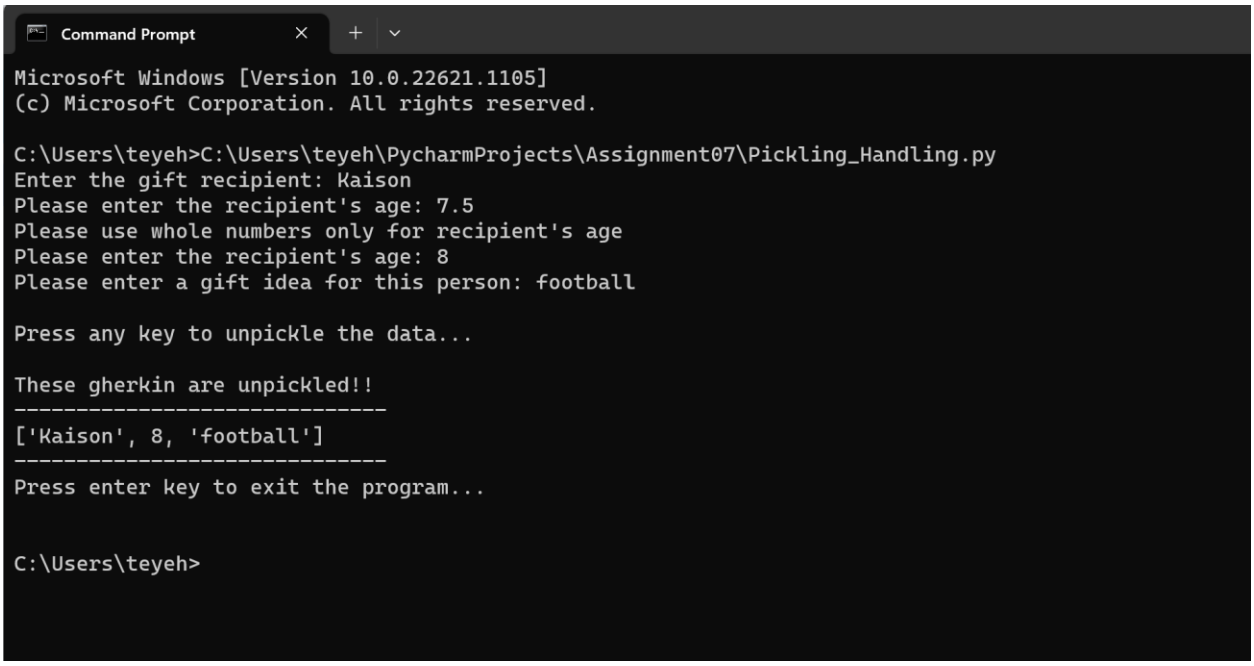

Figure 6: A screenshot of the script running in command window.

Summary

After reading through the notes, videos, and reviewing the website I gain better understanding to execute my work successfully and demonstrate my knowledge by performing assignment 7.

Pickling Research URL

https://docs.python.org/3/library/pickle.html#:~:text=%E2%80%9CPickling%E2%80%9D%20is%20the%20process%20whereby,back%20into%20an%20object%20hierarchy.

I found the above URL useful with the examples giving on the module interface explain the dumps() , pickler, Unpickler and .load() function to my likening.

Exception Handling Research URL

https://docs.python.org/3/tutorial/errors.html

The above URL was very useful with detailed explanation on errors and exception with more examples with more tutorials to my likening.