Guela Parker

May 29th, 2022

Foundations of Programming (Python)

Assignment 07

https://github.com/guelaparker206/IntroToProg-Python-Mod07

Exception Handling and Pickling features

Introduction

This assignment is about exception handling and pickling features. Covering, definition of the concepts, references and a practical script.

Definition and references

Exception Handling

One of the best sources for Python documentation is The Python Tutorial website docs.python.org. It is a trustful website and the information is easy to digest even for beginners. In the external link: https://docs.python.org/3/tutorial/errors.html it is explained that exceptions is one of the most distinguishable kind of errors in Python programming, it is defined as follow: "Even if a statement or expression is syntactically correct, it may cause an error when an attempt is made to execute it. Errors detected during execution are called exceptions and are not unconditionally fatal."

Another website I find easy to understand is https://realpython.com/python-exceptions/. It explains the code piece by piece, compares the two most common type of errors and includes a cheat sheet!

Pickling

Is defined in the Python documentation as: "Pickling" is the process whereby a Python object hierarchy is converted into a byte stream. External link: https://docs.python.org/3/library/pickle.html#module-pickle

Another website I found useful in learning about the pickling feature in Python is https://www.geeksforgeeks.org/understanding-python-pickling-example/. I like it because it explains with examples in different scenarios how the code works, as well as the advantages of using the pickling feature.

Creating the Python Script for Exception Handling

The script uses the function "while True:" to create a loop if the condition under "try" is not true. It breaks when the user enters an integer.

```
# Title: Error Handling
# Description:  ValueError function
# ChangeLog: (Who, When, What)
# GuelaP,5.29.2022 Create script
# ------------------------------------------------ #

# Data ------------------------------------- #
random_num = int  # Declares the age variable as an integer

# Processing ------------------------------------- #
while True:
    """
    Ask use for a number
    """
    try:
        random_num = input("Please enter a number: ")  # Asks user to enter
their age
        random_num = int(random_num)
        break
    except ValueError:
        print("That is not a valid number, please try again.")
print("You have enter a valid number. Goodbye.")
```

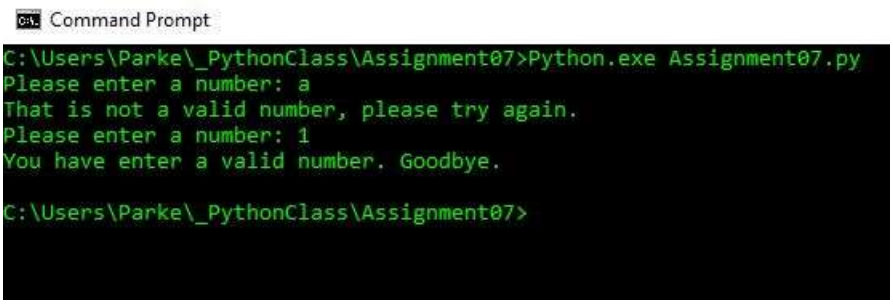

Figure 1. Script running from command prompt.

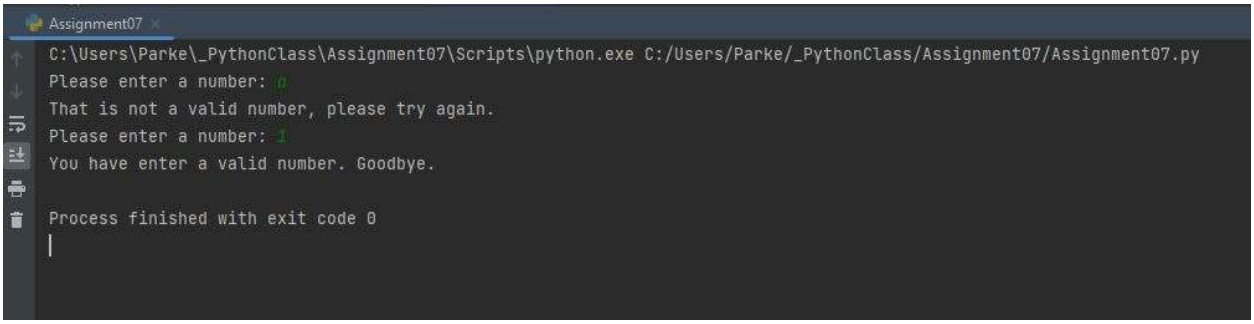

Figure 2. Script running in PyCharm.

Creating the Python Script for Pickling

The script imports code from another file, stores the data with pickle.dump function, and reads the data back with the function pickle.load.

```
# ------------------------------------------------ #
# Title: Pickling
# Description:  Pickle function
# ChangeLog: (Who, When, What)
# GuelaP,5.29.2022 Create script
# ------------------------------------------------ #

# Data ------------------------------------------- #
task_str = '' #Declare variable as string
priority_str = '' #Declare variable as string

# Processing ------------------------------------- #
import pickle  # This imports code from another code file
task_str = str(input("Enter a task: "))
priority_str = str(input("Enter the priority: "))
data_lst = [task_str, priority_str]

# Store the data with the pickle.dump method
objFile = open("Pickle.dat", "ab")
pickle.dump(data_lst, objFile)
objFile.close()

# Read the data back with the pickle.load method
objFile = open("Pickle.dat", "rb")
objFileData = pickle.load(objFile)
objFile.close()
```

Figure 3. Script running from command prompt.

Figure 4. Script running in PyCharm.

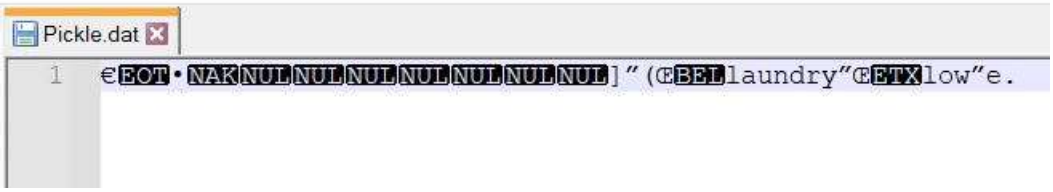

Figure 3. Data stored in file.

Summary

This assignment covered exception handling and pickling features. Their definition of the concepts, references and a practical script.