Fatima Sharif

February 27, 2022

Foundations Of Programming: Python

Assignment07

https://github.com/Fmsharif3/Python-Assignment07

Exception Handling and Pickling in Python

Introduction: In this paper I will explain to you the steps I took to create a new script that demonstrates how Structured error handling and Pickling work in Python.

Exception Handling: While programming we sometimes run into syntax or exception errors. Syntax errors occur when using the wrong indentation or when there are too many brackets. Exception errors on the other hand occurs when syntactically correct code results in an error.

Exceptions Versus Syntax Error: Below are some examples of exception and syntax errors that I found very useful at https://realpython.com/python-exceptions/. Figure 1 displays a syntax error due to the use of too many brackets. While Figure 2 displays an exception error. You'll notice that in figure two Python doesn't display this error as an "exception error" but rather displays the specific exception type as, "ZeroDivisionError." This is because Python comes with many built-in exceptions.

Python Traceback

```
>>> print( 0 / 0 ))
  File "<stdin>", line 1
    print( 0 / 0 ))
                  ^
SyntaxError: invalid syntax
```

Figure 1: Syntax Error

Python Traceback

```
>>> print( 0 / 0)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: integer division or modulo by zero
```

Figure 2: Exception Error

The try and except Block - Handling Exceptions: The easiest way to handle exceptions is to use the try statement with an except clause**.** The try statement allows you to section off some code that could potentially raise an exception. While the except clause allows you to provide a block of statements that are executed only if an exception is raised.

My Example of an Exception Error: After doing some research and going through examples from this week's lecture video I decided to try out the try and except block. I first opened PyCharm created a new project then, in that project file I opened a new python file and began to test out the try and except block. Using the try statement, I ask python to open a text file then decided to use the except clause to print a message to the user if the text file is found. Lastly, using the else clause to print a message if the file cannot be found.

```
# ------------------------------------------------ #
# Title: Lab7-1
# Description: A simple example of storing data in a binary file
# ChangeLog: (FSharif, 2.28.2022, Pickle Demo)
# <Fatima Sharif>,<2.28.2022>,Created Script
# ------------------------------------------------ #

# Exception handling example:

try:
    file = open("AppData.txt", "r")
except IOError:
    print("File not found!")
else:
    print("File found!")

# Pickling Example:
import pickle     # this imports code from another code file!
```

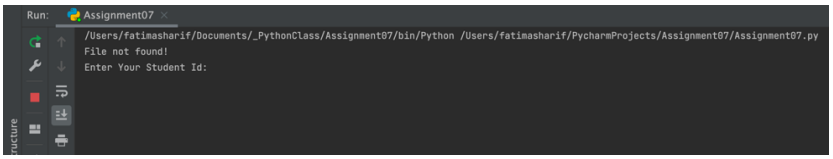
Figure 3: My example Using the try Statement With an except clause

Pickling In Python: Moving onto pickling, I searched the internet for webpages with examples of how to read, write, and save a binary file. While exploring the pickling method, I also learned that pickling it is the best way to save complex data then converting to text files and that you could use the pickle method on tuples, list, and dictionaries. After watching a few YouTube video's, I then decided to try pickling on my own in PyCharm. Below is my example.

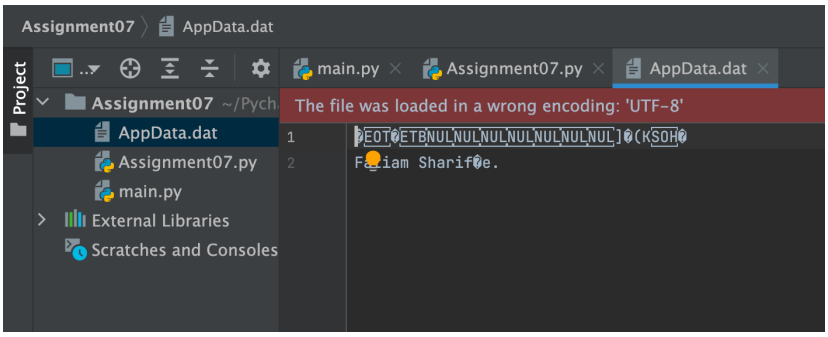
```
# Pickling Example:
import pickle     # this imports code from another code file!

int_StudentId = int(input("Enter Your Student Id: "))
str_StudentName = str(input("Enter Your First And Last Name: "))
lstStudents = [int_StudentId, str_StudentName]
print(lstStudents)

# Now we store the data with the pickle.dump method
objFile = open("AppData.dat", "ab")
pickle.dump(lstStudents, objFile)
objFile.close()

# And, we read the data back with the pickle.load method
objFile = open("AppData.dat", "rb")
objFileData = pickle.load(objFile)   # load() only loads one row of data.
objFile.close()

print(objFileData)
```

Figure 4: Pickling Method example

Output:



Pickling File:



Terminal output:

```
[Fatimas-Air:~ fatimasharif$ python3 /Users/fatimasharif/PycharmProjects/Assignme]
nt07/Assignment07.py
File not found!
Enter Your Student Id: 12
Enter Your First And Last Name: Khalida Sharif
[12, 'Khalida Sharif']
[12, 'Khalida Sharif']
Fatimas-Air:~ fatimasharif$
```

Summary: In this paper I explained to you the steps I took to create a new script that demonstrates how Pickling and Structured error handling work.