Katrina Taylor

August 24, 2020

Foundations of Programming: Python

Assignment07

<https://github.com/katrinataylor/ITFnd100-Mod7>

**Pickling and Exception Handling**

# **Introduction**

The goal of this assignment is to research pickling and exception handling in Python and demonstrate how these work with new scripts.

# **Pickling**

Pickling is used specifically in Python to convert an object in memory to a byte-stream (and back again). Pickling can be referred to as serialization in other languages. This is often used to save some space when storing a large amount of data.

To pickle, you must always first load the function use “import pickle”. To write data to a file object while pickling, you use the “dump” method. And finally, to read a pickled object you can use the “load” method (Figure 1).

# ------------------------------------------------- #  
# Title: Assignment07-1  
# Description: An example of pickling  
# ChangeLog: (Who, When, What)  
# Katrina Taylor,8.24.2020,Created Script  
# ------------------------------------------------- #  
  
import pickle # Load the Pickle function  
  
# Create an object to save (in this case a dictionary of kids vocabulary)  
kids\_dict = {"a": "apple", "b": "banana", "c": "carrot"}  
  
# Write the kids dictionary to a file using the dump method  
objFile = open("KidsDictionary.dat", "ab")  
pickle.dump(kids\_dict, objFile)  
objFile.close()  
  
# Read the kids dictionary from the saved file using load method  
objFile = open("KidsDictionary.dat", "rb")  
objFileData = pickle.load(objFile)  
objFile.close()  
print(objFileData)

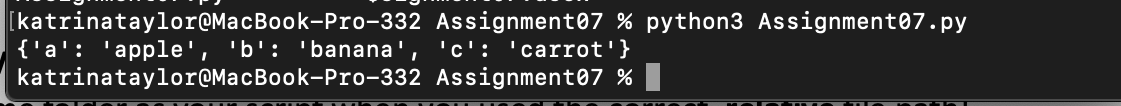
***Figure 1: An example of using the Dump and Load methods of Pickling***

The above example provided the expected output in PyCharm (Figure 2), as well as when ran in the Terminal (Figure 3).

A screenshot of a computer

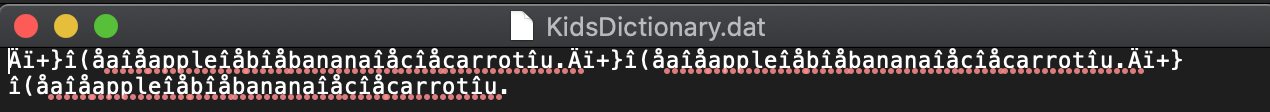
Description automatically generated

***Figure 2: The output of the pickling example script in PyCharm***



***Figure 3: The output of the pickling example script in the Terminal***

Without using the load pickling method to read the byte-stream, the data is not easy to understand (Figure 4).



***Figure 4: The byte-stream file***

The following resources also provide a good introduction to pickling in Python because they assume the viewer has little knowledge around this subject area:

* <https://www.youtube.com/watch?v=2Tw39kZIbhs>
* <https://medium.com/@lokeshsharma596/what-is-pickle-in-python-3d9f261498b4>

# **Exception Handling**

When requesting user input, errors may occur. For example, they may provide a numeric value when they should have provided alpha characters or vice versa. When errors occur, Python will provide error messages that are not the most user-friendly.

In order to help the end user more easily understand the issue, exception handling can be added to scripts. You can use a try-except block of code to go about exception handling (Figure 5).

# ------------------------------------------------- #  
# Title: Assignment07-2  
# Description: An example of exception handling  
# ChangeLog: (Who, When, What)  
# Katrina Taylor,8.24.2020,Created Script  
# ------------------------------------------------- #  
try:  
 print("Welcome to Kelley Blue Book 2.0!")  
 miles = int(input("How many miles are logged on your car? "))  
except ValueError as e: # Error message for when a numeric value is not provided  
 print("Only numbers can be inputted.")  
except Exception as e: # Error message for all other errors  
 print("There was a non-specific error!")  
 print("Built-In Python error info: ")  
 print(e, e.\_\_doc\_\_, type(e), sep='\n')  
else: # When there is no error  
 print("You inputted " + miles + "miles driven using your car.")

***Figure 5: An example of using try-except to handle errors***

The above example provided an easy to understand error and works in PyCharm (Figure 6), as well in the Terminal (Figure 7).

A screenshot of a cell phone

Description automatically generated

***Figure 6: The output of the try-except example script in PyCharm***

A screenshot of a cell phone

Description automatically generated

***Figure 7: The output of the try- except example script in the Terminal***

The following resources also provide a good introduction to exception handling in Python because they chunk out the different ways one could handle exceptions:

* <https://www.programiz.com/python-programming/exception-handling>
* <https://www.w3schools.com/python/python_try_except.asp>

# **Summary**

In this assignment I researched pickling and exception handling in Python and demonstrated how these work with new scripts.