Dominic Ferenczy  
May 18, 2021  
Foundations of Programming: Python  
Assignment 07  
[uwferenczy/IntroToProg-Python-Mod07 (github.com)](https://github.com/uwferenczy/IntroToProg-Python-Mod07)

Pickling and Exception Handling

# Introduction

This document details how I applied what I have learned in module 7 of the Introduction to Programming class to Pickle and Unpickle a status dictionary into a .dat file specified by the user and use error handling if the .dat data type was not specified. Part of the assignment was to research Pickling and Exception Handling online. I used DataCamp.com for both. Topics covered in this document include how to use pickling in a Python script, how to implement exception handling and how to test the code.

# Pickling

Pickling can be used to save objects to a file. This is useful when you do not need to process the objects into a readable text file, but still need to be able to store the data for use at later run times. The pickle library needs to be imported to be used in the script. Libraries can be imported using the import function. Once imported, the file needs to be opened in a way that specifies binary. This can be done by specifying ‘wb’ in the open function. Once opened, the dictionary object can be pickled into the file using pickle.dump(). The resulting code can be seen in figure 1.

  
Figure 1: PyCharm showing a dictionary being pickled into a file

The file can then be unpickled using a similar process. The file needs to be opened, this time specifying ‘rb’ to read the binary file that was pickled. This file can then be unpickled using pickle.load(). The resulting code can be seen in figure 2.

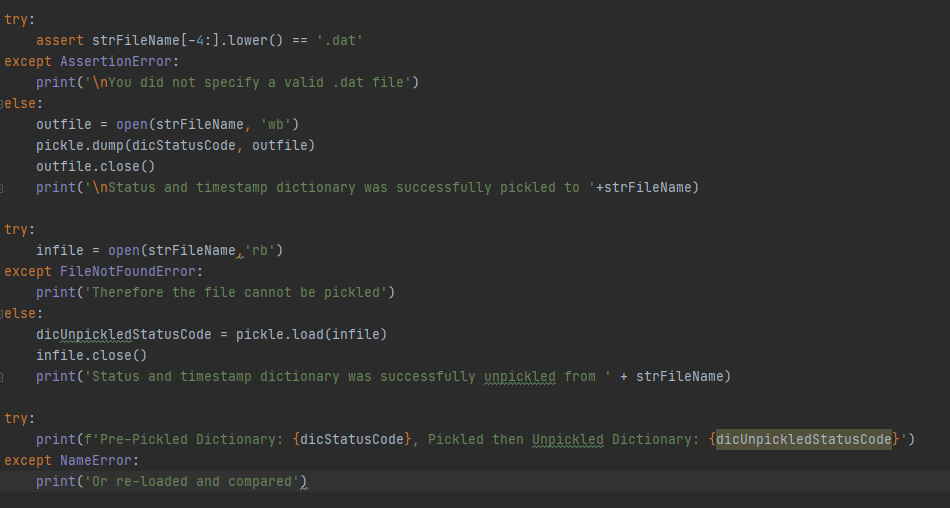
  
Figure 2: PyCharm showing a dictionary being unpickled into a variable

I commented out the infile line because I ended up integrating it into an exception.

For more information on Pickling, please visit [Python Pickle Tutorial - DataCamp](https://www.datacamp.com/community/tutorials/pickle-python-tutorial) (external site).

# Exception Handling

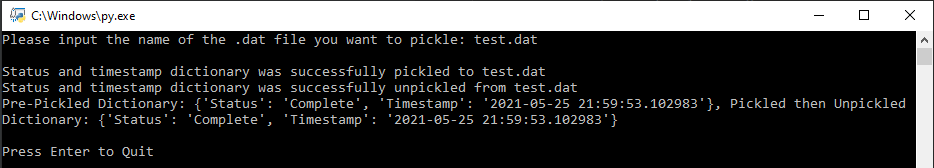
Exception handling can be useful to force the correct use of an application and prevent crashes. In the case of my code, I wanted to ensure that users specify a .dat file type. To do this, I needed to import the sys library. Once imported, I was able to use the try, accept function. I checked to see if the right 4 characters of the file name are .dat using the assert function. If they were not, I raised an exception using the except function and specified that it was an Assertion Error. Instead of exiting the application, I printed a message to notify the user that they did not specify a valid .dat file. The try, except functions can also use an else and finally operations. Because one error can cascade out to other downstream functions, I created try except statements for pickling the file, unpickling the file and displaying the comparison between the original dictionary and unpickled dictionary. During the testing, I tried to cause an error to occur, then copied the error name and used it as the exception in the try except function for each step. The resulting code can be seen in Figure 3.

   
Figure 3: PyCharm showing multiple try except functions

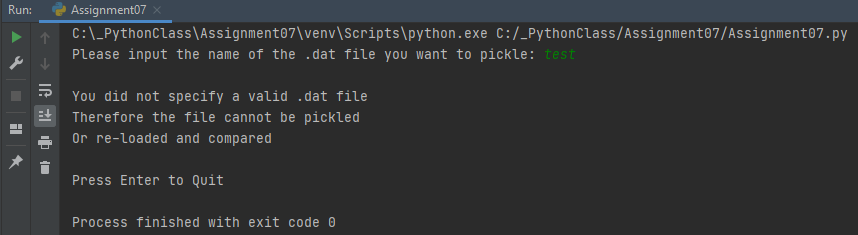
For more information on exception handling please visit [(Tutorial) Exception and Error Handling in Python - DataCamp](https://www.datacamp.com/community/tutorials/exception-handling-python) (external site).

# Testing Application

I tested the application in command prompt as shown in figure 4. It ran correctly in command prompt and matched the assignment requirements.

  
Figure 4: Command Prompt showing program running correctly

I then tested the application in PyCharm as shown in figure 6. It ran correctly in PyCharm and matched the assignment requirements.

  
Figure 5: PyCharm showing program running correctly

I then opened the .dat file to verify that the dictionary was pickled to the file. The items were pickled correctly.

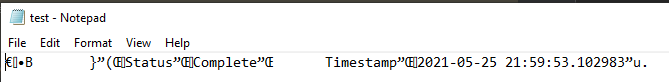


Figure 6: Notepad showing items added correctly

# Summary

This document captured what I learned in module 7 of the class. I documented the steps I took to use classes to use pickling in a Python script, how to implement exception handling and how to test the code.