Reece Wonio

May 30th, 2022

IT FDN 110 A

Assignment 07

Pickling and Error Handling

Introduction

In this paper I will discuss the concepts of pickling data and error handling in Python. These concepts fit right into existing code we have worked with throughout this class, but their purpose can save users space on their machines and make the overall user experience more friendly and efficient. I hope after reading this, the reader will have a basic understanding of how pickling and error handling work within Python.

What is Pickling?

When a user is serializing and de-serializing data in Python, it is called pickling, though it goes by many other names in other languages. Basically, when you are pickling in program, you are storing objects in the program. This is particularly useful when working with large datasets because when the data is pickled, it is stored as a binary file. This saves space on memory and the program can recall the data much faster. For a great video tutorial on how to pickle your own program as well as some further text on the concept, click [HERE](https://pythonprogramming.net/python-pickle-module-save-objects-serialization/)

What is Error Handling?

Human error is an inevitable aspect when a user is running a program. When a n error occurs, Python may return a syntax error or exception based the type of error that occurred. Sometimes these errors can be confusing and may not direct the user to a clear solution. Thankfully there is error handling in Python. Through the try and except statement, the individual writing the program can place “try” above a section of code that a user may be inclined to write an error in. Below that, they can write “except” and call out the specific error they want to the program to catch. For example, the writes could place try above an input function that involves division. Then in the except statement, they could write something that looks for the division of zero and catch that with a prewritten message if that error occurs. For more info on error handling, checkout the webpage [HERE](https://docs.python.org/3/tutorial/errors.html#handling-exceptions)

The Code for the Assignment, Pickling

When pickling a data set or program, the first step is to import the pickle module, which is done by the import pickle program. From there, the process was very similar to the functions assignment last week with some small adjustments. The first thing to note is that pickled data cannot be stored in a text file, it must be saved as a binary data file. Instead of saving our file with .txt like we have been doing, it must be saved as .dat to change the file type.

Another change is when we are writing or reading the file, we cannot just put “W” or “R” in the program. There must be a “b” at the end. This tells the program that it is a binary file access mode and allows the program to write or read the pickled file.

Lastly, you must enter the .dump function. This function requires two arguments, the data to pickle and which file to store it. Then when reading the file, you must enter the .load function which tells the program to load the pickled file you have called out. This can all be seen in figure one below.

Text

Description automatically generated

***Figure 1: Pickling a data set with functions***

Error Handling the Assignment

For this assignment I was asking the user to divide two numbers through an input function. Knowing that two possible errors that could come from this are dividing by zero or entering a non-number, I called out those errors through the try and except functions and told the program to write a message to the user to check their work. This is explained through figure 2 below.

Text

Description automatically generated

***Figure 2: Error Handling the input functions***

Summary

This was a fun assignment that tied in a lot of concepts from previous lessons with minor changes. I am excited to explore pickling more especially with reading how it can help machine learning and safe time and space with storing data. The error handling was also great to learn so that I can create more user-friendly programs in the future. Below are the remaining images of the program running.

Text

Description automatically generated

***Figure 3: Program running without error***

***Text

Description automatically generated***

***Figure 4: Program with Error***

***A screenshot of a computer

Description automatically generated with medium confidence***

***Figure 5: Pickle File***

***Graphical user interface, text, application

Description automatically generatedFigure 6: Program Running in IDLE***