Lin Martin

August 24, 2021

IT FDN 110 A

Assignment 07

<https://linmartinn7.github.io/IntroToProg-Python-Mod07/>

# Exception Handling and Pickling in Python

## Introduction

For Assignment 07, our class was tasked with researching two features within Python, exception handling and pickling. We were also tasked with creating a custom script of our own creation to demonstrate our understanding and the functionality of both features. In the document below, I will discuss what I learned about both features and how I created the script for this week’s assignment.

## What is Exception Handling?

In Python, exception handling is a process that allows a script to bypass an error. This is done using the *try* statement. When an error occurs within the script, the *try* statement will force the script to process a different command. The *try* statement will be followed up with an *except* statement, which will allow the user to create an error message should the *try* condition fail to resolve the error. Exceptions can also be used when specific conditions are met using the *raise* function in Python. This allows programmers to allow certain errors to occur in their code without stopping it from running, which is a useful feature when trying to overcome errors that could be produced but don’t necessarily need to break the entire script.

### Exception Handling Research Links

While researching exception handling in Python, there were two websites I found useful.

The first site I found helpful with understanding exception handling was W3 School’s tutorial (Python Try Except, <https://www.w3schools.com/python/python_try_except.asp>, 2021)(External Link). This site provided a high-level overview of exception handling, which I was appreciative of. When first trying to learn about this process, I appreciated W3 School’s simple presentation of the concept so I could understand it more easily. The site also included multiple examples of how to use the *try* and *except* commands to handle errors and included a section on using *finally* and the *raise* exception commands as well.

The second site I used to learn more about exception handling was Tutorials Point’s article on exception handling (Python – Exception Handling, <https://www.tutorialspoint.com/python/python_exceptions.htm>, 2021)(External Site). This site provided a much more thorough explanation on exception handling and provided multiple examples of the types of exceptions that can be used in a script. There were also examples of how to write these exceptions in Python that would and would not produce errors, which was very informative.

## What is Pickling?

In Python, the pickle function is used to serialize/de-serialize objects within the code. In laments terms, this process converts text stored within the Python script into binary code. This functionality is incredibly useful for storing and transferring data. By pickling the data being manipulated by the Python script, it can be stored more concisely without losing any data. While pickled data may look unusable, it still maintains all the original information that existed before being pickled. This allows the data to be fully utilized fully in the future.

This process is different than compressing data for two major reasons; pickling data does not result in data loss and is reversable. When data is compressed to reduce the file size, it loses some of the additional data stored within that file. Also, once a file is compressed, the process often cannot be reversed. By pickling data in Python, the integrity of the data is maintained while reducing the file’s size. This allows larger sets of data to be transferred between locations more easily. Pickling is fully reversable as well, which allows the data to be accessed in the future.

### Pickling Research Links

While researching pickling in Python, there were two websites I found very useful.

The first was a tutorial site hosted by DataCamp which was very thorough in explaining what pickling is and how it can be used in a Python script (Pickle in Python: Object Serialization, <https://www.datacamp.com/community/tutorials/pickle-python-tutorial>, 2021)(External Site). The article also included examples of what kind of data could be pickled by Python and provided broken down code samples explaining each step on pickling data.

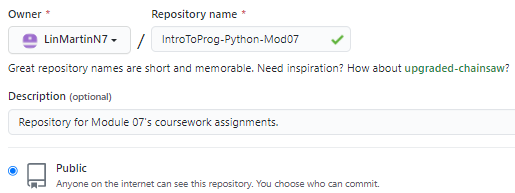
The second site I found very useful was an article by Lokesh Sharma on Medium.com (What is Pickle in python?, <https://medium.com/@lokeshsharma596/what-is-pickle-in-python-3d9f261498b4>, 2021)(External Site). This article discussed pickling from a high-level overview that was less in-depth than the first article. I appreciated this because it helped me understand pickling in a plain talk manner. The article also had an example provided demonstrating how to pickle and unpickle data. These completed script examples of how to pickle and unpickle data made the concept easier to understand for me.

|  |  |
| --- | --- |
|  |  |

***Figure 01: Sample of pickling and unpickling data from Lokesh Sharma’s article on pickling. (What is Pickle in python?,*** [***https://medium.com/@lokeshsharma596/what-is-pickle-in-python-3d9f261498b4***](https://medium.com/@lokeshsharma596/what-is-pickle-in-python-3d9f261498b4)***, 2021)(External Site)***

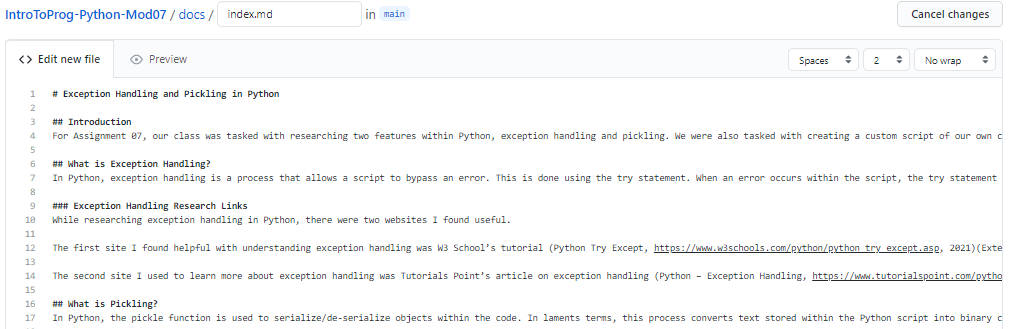
## Creating a GitHub Site

For Assignment 07, our class was tasked with creating a new GitHub repository to store the files for this weeks’ coursework, as well as creating a GitHub website to display our written paper. To get started, I created a new repository on my GitHub account with the IntroToProg-Python-Mod07 name.



***Figure 02: Creating new GitHub repository for Module 07’s coursework.***

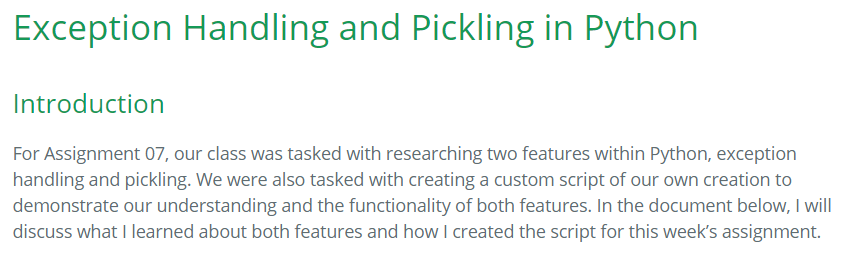
Next, I created a docs folder and index.md page to serve as my GitHub site for displaying my written paper. I added some of my work in progress for the Assignment 07 write up document so the page could populate with some information. Due to having used GitHub in a previous course to create webpages, formatting the text felt fairly familiar.



***Figure 03: Creating a GitHub site to post Assignment 07’s paper.***

I made a note to save screenshots of the images used in my written document so they could be uploaded onto the GitHub site once those sections of the paper were completed. The drag and drop feature within GitHub for uploading images stored on PCs is very useful.

Once I had enough sample text on the site, I saved the file and went into the settings for my repository. I accessed the page menu and changed the theme of the page. After waiting several minutes, I verified that the theme had taken on my GitHub site and that my document was shown correctly on the page.

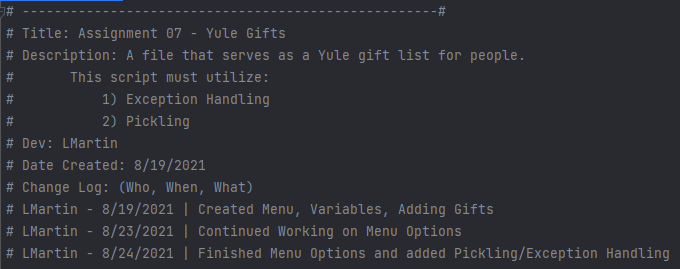
****

***Figure 04: Creating a GitHub site to post Assignment 07’s paper.***

## Creating the Python Script

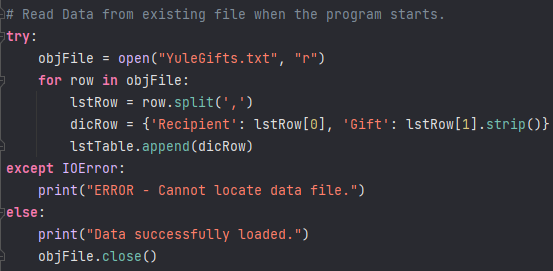
For this assignment, we were tasked with creating a script in Python that would use pickling and exception handling. I decided to create a script that would capture gift ideas for Yule, given that we are drawing closer to winter and I like to pre-plan my holiday gift giving.

Like previous assignments, I created a header for the document and updated it as I worked on this script.



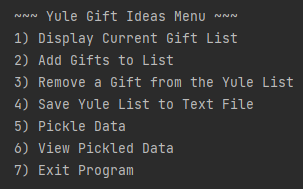
***Figure 05: Script header in PyCharm.***

Next, I declared the variables used in the file and added a section to read the data from an existing file called *YuleGifts.txt*. This file serves as the base file for the script and gift list so the user can use it repeatedly. At this point in the script, I decided to add an exception handling clause in the event that the *YuleGifts.txt* file could not be located.



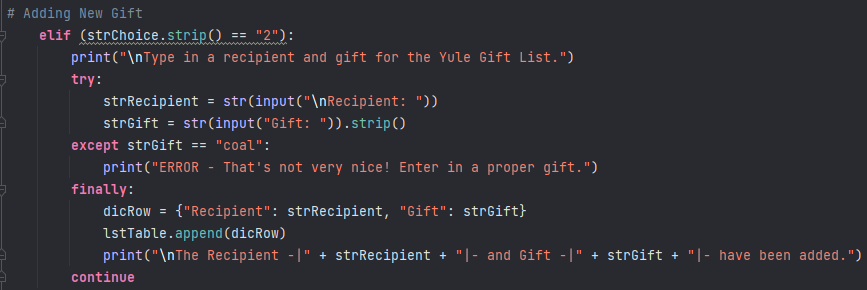
***Figure 06: Loading data from YuleGifts.txt with exception handling clause.***

The rest of the file worked similarly to some of the previous assignments we were tasked with. I created a menu with multiple prompts for the user to select from and added code that performed each function in the list.



***Figure 07: Yule Gift Idea Menu being tested in PyCharm.***

I attempted to add another exception handle parameter in the script for adding a new gift. I was unsuccessful to get it to function correctly but wanted to keep it included in the script. I would like to go back and get this once I have a better understanding of where I went wrong with the exception handling function.



***Figure 08: Adding New Gift script in PyCharm where exception handling failed to work correctly.***

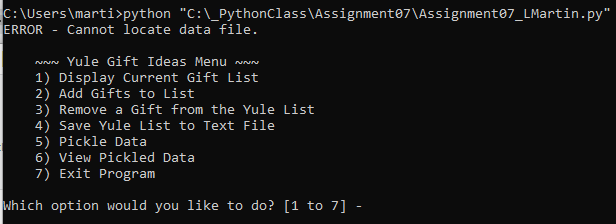
I continued creating the script without any other exception handling statements, instead redirecting my focus to the pickling menu options I had created in the list. I had two different options referring to pickling, *#5 Pickle Data* and *#6 View Pickled Data* (Figure\_\_). I finished the script with an option to close the file.

|  |  |
| --- | --- |
|  |  |

***Figure 09: Python script for pickling data and viewing pickled data.***

## Testing the Python Script

Once my script was created, I knew I needed to test it in PyCharm and in Windows Command Prompt. I had been testing the script repeatedly in PyCharm while I had been writing statements, so I was confident in its performance there. I tested the script in Command Prompt and immediately noticed that my first exception handling clause was working because I got an error that the file could not be located. This felt like a huge success for me due to my inability to get my second exception handling clause to work in the new gift creation option.



***Figure 10: Exception handling working in Windows Command Prompt.***

I proceeded with testing the file, adding multiple recipients and gifts to the Yule Gift Ideas List. I confirmed that a new YuleGifts.txt file was created in my profile on the PC when I selected the prompt to save the list.

|  |  |
| --- | --- |
|  |  |

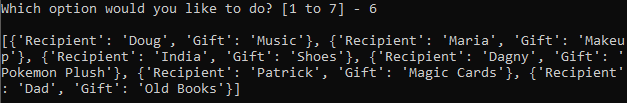
***Figure 11: Successful creation of recipients in Python script, generating a text file once the prompt to save was selected.***

Next, I wanted to verify that a pickled file of data would be created by the script. I selected *#5 Pickle Data* and immediately got an error that I had not received while testing in PyCharm. I went back to the code and realized I forgot to add import pickle to the beginning of the code for the *#5 Pickle* Data section. I immediately fixed the error and re-ran the code, getting a successful pickling confirmation. I verified that the file had been created on my C Drive as well.

|  |  |
| --- | --- |
| Windows Command Prompt |  |
| C Drive |  |

***Figure 12: Successful pickling of the Yule Gift List script.***

When selecting the option to view the pickled data in command prompt, all entries appeared correctly (Figure 13). With this, I was ready to turn the assignment in.



***Figure 13: Successful reading of pickled data in the Yule Gift List.***

## Summary

I enjoyed the change of pace this week’s assignment provided our class, especially after my struggles with Assignment 06. Researching items online for work is second nature to me, so having to research both pickling and exception handling was fun. I think it’s good that we were expected to share what websites were beneficial to us in our learning. There are multiple learning styles, so seeing what methods make sense to students must be helpful for professors.

The flexibility we had with the assignment this week was a lot of fun. I felt more passionate about my script because it was something that held personal meaning to me. I know it’s not perfect or the most advanced thing I have created, but it is a cool tool I can use. Also, even though pickling isn’t really intended for this purpose, I can use it to securely store notes and lists such as these were no one will accidentally find them. The amount of

‘junk code’ in the pickled files will detour prying eyes, which can be very helpful come this holiday season.