

# Knowledge Document

## GitHub Link

[https://github.com/Abracollabra/Assignment\\_07.git](https://github.com/Abracollabra/Assignment_07.git)

## Introduction

Module 07 is focused on error handling, as well as the introduction of working with binary files. Students are asked to begin with last week's script and modify it to include various types of structured error handling.

A second major task for this week is to modify the permanent data store to binary data.

## Exception Handling Resource Samples from Web Searching

I'm comfortable with both text and videos, but if I'm trying to learn a concept, I often first go to videos.

This is an example of an instructional video that works well for me, followed by some of the highlights of what's especially helpful:

[Python tricks: All there is to know about Exceptions](#) - YouTube

Sebastiaan Mathôt

- The instructor speaks slowly and clearly enough for me to understand him.
- I like seeing the big picture at the outset, so, though it's minor, I appreciate seeing the template for what he's going to do on the screen from the beginning. (See *Figure 1* below)
- Honestly, the fact that his background is very light colored (approaching white), with primary text in blue and green is huge for me.
- The scope and sequence of what he covers are manageable and logical.

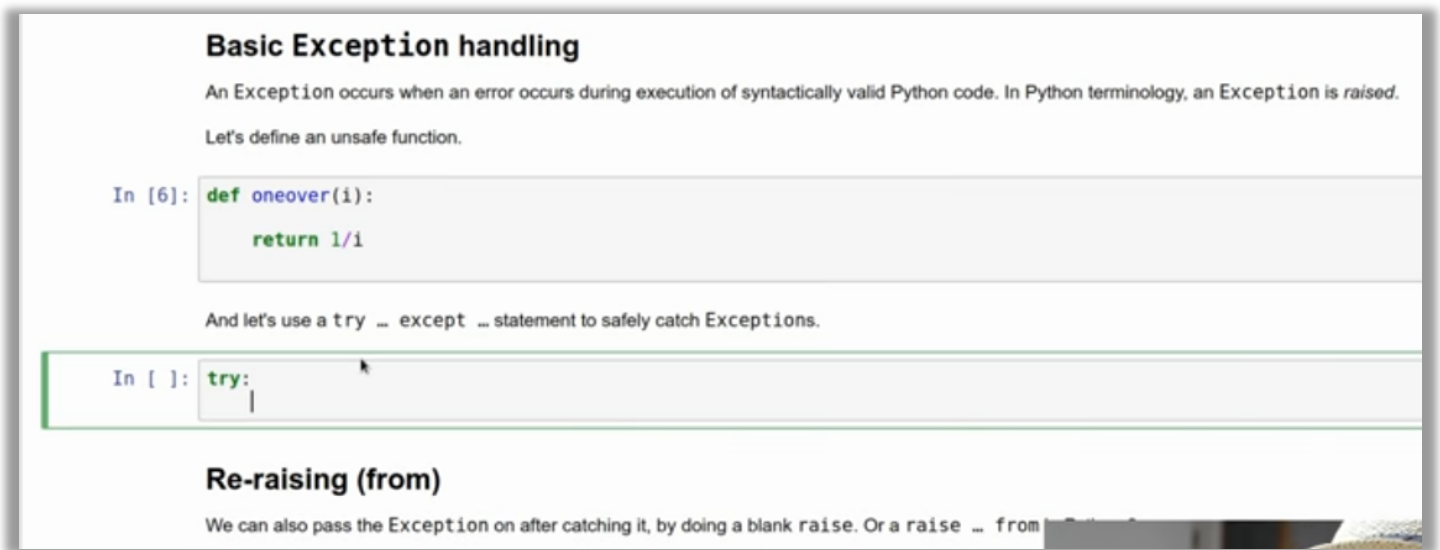


Figure 1 - Screenshot of "Python tricks: All there is to know about Exceptions" video

An example of a video that isn't as useful for me at this stage is:

[Try Except Exception handling in python | Python tutorials for Beginners](#) - YouTube  
Smart Programmer

I can see its potential utility for someone who just wants a quick overview of the concept in action. But at my current knowledge level, because it doesn't have any narration, I've found myself spending more time wondering what's going on than the lack of narration saves me.

## Pickling in Python Samples from Web Searching

[Python: How to Save and Load Data for Beginners](#) - YouTube

TokyoEdTech

- Super simple
- Nice and slow
- Every line he types has a comment that he leaves on screen
- He highlights blocks of code that he references after the fact
- I like the color theme he uses
- Huge bonus - Everything he does in the entire tutorial fits on the screen at the end, without having to shrink it into oblivion

[Python Pickle Module for saving Objects by serialization](#) - YouTube video embedded in webpage pythonprogramming.net

- This is great because it has a clearly formatted page of text on the website to go into more depth about what the video covers. Specifically, the horizontal bars on the page that separate various steps and sections are very helpful
- Another light-colored background
- I appreciate that at several points along the way he rhetorically asks what something is used for, why it matters, etc, then answers the question.
- Very conversational speaking style

## Script Creation Highlights

I got stuck trying to continue with modifications, since I got a ValueError from the jump. Since I hadn't changed anything yet with how the program ran, I tried to go back to the clean code from Assignment 06 but kept getting the same error. Dirk helped me figure out what wasn't working.

When trying to figure out how to incorporate pickling, I ended up accidentally deleting the entire script and replacing it with a weird series of characters. Unable to 'undo' the move beyond replacing the characters with the line of code I was trying to move, after a variety of StackOverflow searches, I gave up for the night.

This morning when I started over, I was surprised to discover that same series of characters, this time in the .dat data file I'd established. See Figure 4 below, which includes the original caption I created. And a search using those characters didn't turn up an explanation but did lead to a bunch of web pages that had been moved. (See Figure 5 below.)

And though I suspected that I was on the right track last night because I wasn't getting any errors, later experience showed that my three seconds of optimism before The Event, was, in this case, misplaced.

As I continued to try to fix the UnicodeDecodeError of 'invalid byte start', I realize that every time I got that error, that combination of characters was repeated in the binary file.

I learned about Spyder's [History Log](#) and History Pane, but since I'd already started over by then, and it's not simple to access, but it's good to know about for the future.

By the way, the screenshots of the code running in Spyder and the terminal are from last night, and they're limited because I hadn't gotten the programs to perform as directed. (Figures 2 and 3 below)

I spent a lot of time today trying to get the pickling to work, but in the end, I decided to admit reality.

In the interest of keeping it real, especially since I didn't manage to get the pickling to work, below is a hit parade of the errors I encountered along the way. I managed to make most of them go away at one point or another, but it was ultimately a game of Whack-o-Mole.

- UnicodeDecodeError: 'utf-8' codec can't decode byte 0x80 in position 0: invalid start byte
- TypeError: a bytes-like object is required, not 'str'
- TypeError: file must have a 'write' attribute
- AttributeError: '\_io.BufferedRandom' object has no attribute 'append'
- AttributeError: '\_io.BufferedReader' object has no attribute 'clear'

Now the code doesn't run, but even if I restored it to its original state, it wouldn't have the attributes for this week's assignment. So I guess what it shows indirectly is my journey to the school of hard knocks.

When I went to upload to GitHub, I realized that somehow the binary file I'd created had become a VLC video file on my computer. So, I adjusted the capitalization of the file name, opened it through Spyder again, and deleted its contents. Hopefully that'll at least make it a viable file on its own.

And yes, had I gotten past the pickling, which I couldn't seem to back away from, I would have gone on to error handling.

```
In [53]: runfile('/Users/d\Documents/Coding!/UW_PCE/Mod_07/
CDInventory.py', wdir='/Users/d\Documents/Coding!/UW_PCE/
Mod_07')

Menu

[l] load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] delete CD from Inventory
[s] Save Inventory to file
[x] exit

Which operation would you like to perform?
[l, a, i, d, s or x]:
```

Figure 2 - CDInventory.py Running in Spyder

```
[(base) deborahc @Deborahs-MacBook-Pro Mod_07 % python CDInventory.py

Menu

[l] load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] delete CD from Inventory
[s] Save Inventory to file
[x] exit

Which operation would you like to perform?
[l, a, i, d, s or x]:
```

Figure 3 - CDInventory.py Running in the Terminal Window

Figure 4 - Code I Accidentally Replaced My Script With, and That Printed to the Binary File with UnicodeDecodeErrors

 <https://www.gamejolt.com/games/teenagewarriors/667215>

## Game Jolt - Games for the love of it

301 Moved Permanently. nginx

 <https://www.amazon.com/9000W-Turbine-Blades-Controller-Efficiency/dp/B09DD7SSDM>

[www.amazon.com](https://www.amazon.com)

[www.amazon.com](https://www.amazon.com)

 <https://hubtel.com/pharmacy>

## Pharmacy - hubtel.com

We provide over the counter pharmaceuticals

 <https://www.hqmc.marines.mil/ppo/Leaders/Leaders-View/Article/553902/mr-randy-r-sm...>


## Headquarters Marine Corps

Object moved to here.

 <https://www.cannondale.com/Account/Login/en-us/bikes/electric/e-fitness/treadwell-neo>

[www.cannondale.com](https://www.cannondale.com)

Today · Object Moved This document may be found here

 <https://www.habitat.org/in/evansville>

[www.habitat.org](https://www.habitat.org)

Moved Permanently. The document has moved here.

Figure 5 - Results from a Search Using the Combination of Characters That Replaced the Original Version of My Script (See Figure 4 above)

```

In [55]: runfile('/Users/deborah /Documents/Coding!/UW_PCE/Mod_07/
CDInventory.py', wdir='/Users/deborah /Documents/Coding!/UW_PCE/
Mod_07')
Traceback (most recent call last):

  File "/Users/deborah /opt/anaconda3/lib/python3.9/site-packages/
spyder_kernels/py3compat.py", line 356, in compat_exec
    exec(code, globals, locals)

  File "/Users/deborah /Documents/Coding!/UW_PCE/Mod_07/
CDInventory.py", line 12, in <module>
    pickle.dump(lstTbl, open('CDInventory.py', 'wb'))

NameError: name 'lstTbl' is not defined

```

```

In [56]: runfile('/Users/deborah /Documents/Coding!/UW_PCE/Mod_07/
CDInventory.py', wdir='/Users/deborah /Documents/Coding!/UW_PCE/
Mod_07')

```

Menu

```

[l] load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] delete CD from Inventory
[s] Save Inventory to file
[x] exit

```

Which operation would you like to perform?

```
[l, a, i, d, s or x]: x
```

*Figure 6 - Screenshot of the Console Showing That the Segment of Code Mentioned Had Been on Line 12 Before Being Moved. When Moved, the Program Ran*

## Notable Learning

- Time isn't magic.
- Things that look simple and straightforward aren't necessarily so.
- When there are complex, interlocking actions, unless one truly understands each of them, even 'small' changes to the code might not be possible without breaking it.
- If one is continually struggling with the same issue, it's usually faster to go back to basics to get solid with those, rather than to repeatedly try tweaking code and/or processes that aren't fully understood.

I found some interesting videos and websites.

## Notable Challenges

See Notable Learning above.

Time.

Tried to start with the pickling but couldn't figure out where to put the pickling code within the script, or how to write it.

## Summary

I look forward to seeing my own error handling pickling code actually working. Till then, I've got a good start on learning the concepts, and some nice memories of the time I spent with family before starting in on the assignment.