

## Pickling and Error Handling

### Introduction

In week 7, we covered the topics of creating scripts using custom functions, working with binary files, and structured error handling. We also worked on building out advanced GitHub web pages by using markdown files. Pickling and error handling was briefly presented in the course materials, but we were instructed to learn more about these topics on our own as a part of the assignment. Following our independent research, we were tasked with developing our own script to demonstrate our learning.

### Pickling and Error Handling Research

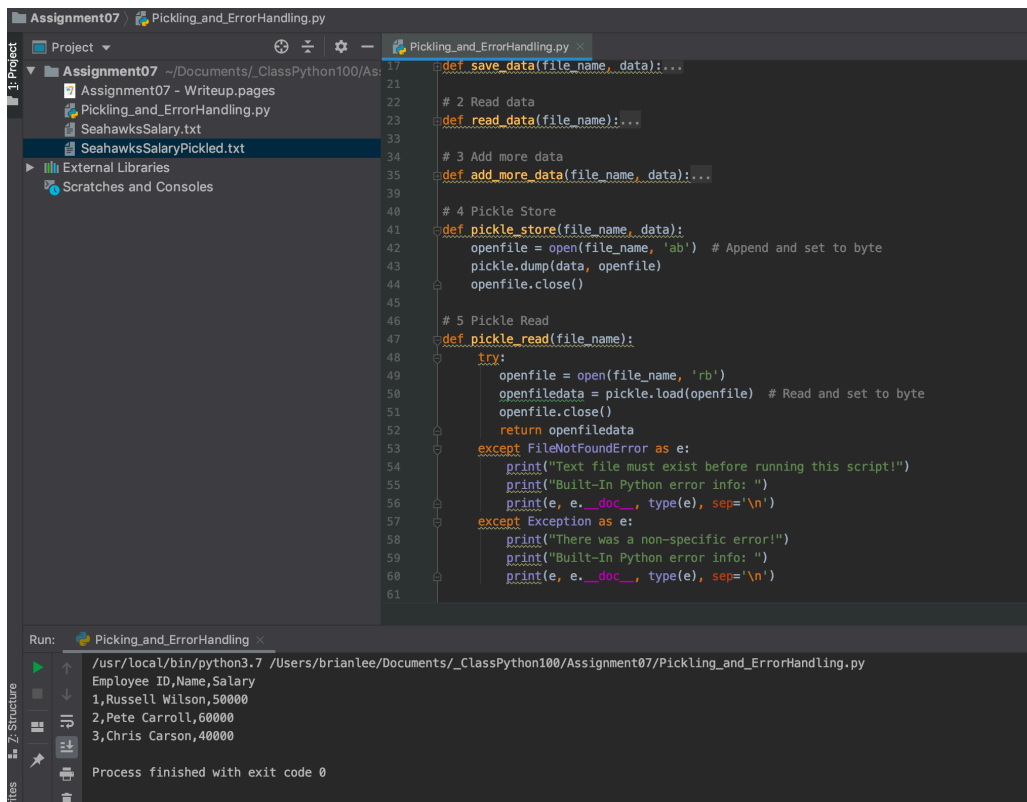
I found both the resources below to be helpful as they are thorough in explaining each step of the process and providing example problem/solutions along the way (similar to the class notes). I appreciate the detail step by step notes to successfully complete the concepts that way we are able to make necessary adjustments from foundation knowledge.

Pickling URL: <https://www.datacamp.com/community/tutorials/pickle-python-tutorial>

Error Handling URL: <https://realpython.com/python-exceptions/>

### Creating and Running My To Do List Script

After reviewing the given videos/course materials provided in Module 7, our seventh assignment was to independently research how pickling and error handling works and then demonstrate by writing our own script. From the information learned from the course materials and external sources, I wrote a script to capture fake salary data for employees, save, read, amend, pickle, and unpickle data files. To be able to use pickle functions, I had to import the pickle module at the beginning of my script. Next, I worked in error handling by using the try-except block within my Pickle Read function as there is a possibility the file does not exist in the directory.



```
def save_data(file_name, data):...
# 2 Read data
def read_data(file_name):...
# 3 Add more data
def add_more_data(file_name, data):...
# 4 Pickle Store
def pickle_store(file_name, data):
    openfile = open(file_name, 'ab') # Append and set to byte
    pickle.dump(data, openfile)
    openfile.close()
# 5 Pickle Read
def pickle_read(file_name):
    try:
        openfile = open(file_name, 'rb')
        openfiledata = pickle.load(openfile) # Read and set to byte
        openfile.close()
        return openfiledata
    except FileNotFoundError as e:
        print("Text file must exist before running this script!")
        print("Built-In Python error info: ")
        print(e, e.__doc__, type(e), sep='\n')
    except Exception as e:
        print("There was a non-specific error!")
        print("Built-In Python error info: ")
        print(e, e.__doc__, type(e), sep='\n')
```

Run: Pickling\_and\_ErrorHandling.py

/usr/local/bin/python3.7 /Users/brianlee/Documents/\_ClassPython100/Assignment07/Pickling\_and\_ErrorHandling.py

Employee ID,Name,Salary

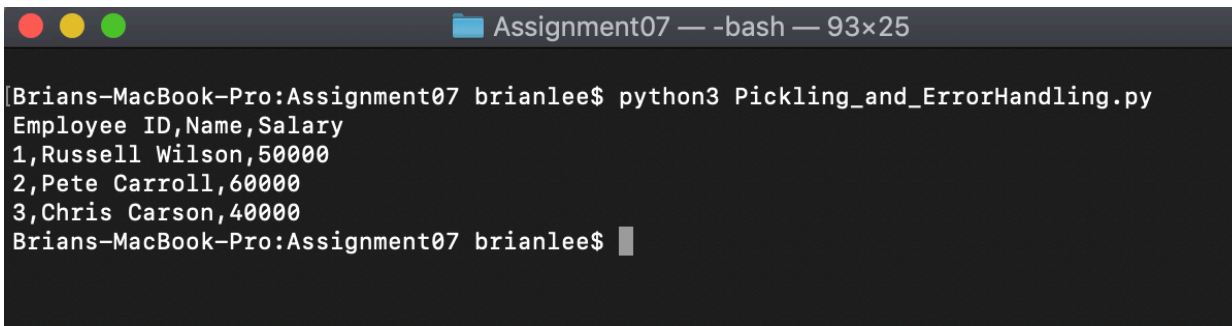
1,Russell Wilson,50000

2,Pete Carroll,60000

3,Chris Carson,40000

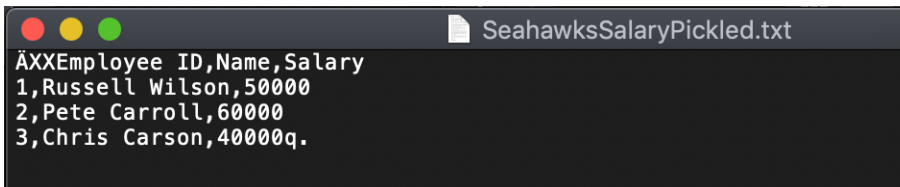
Process finished with exit code 0

Figure 1. Code in PyCharm

A terminal window titled "Assignment07 — -bash — 93x25" on a macOS system. The prompt is "Brians-MacBook-Pro:Assignment07 brianlee\$". The user has run the command "python3 Pickling\_and\_ErrorHandling.py". The script outputs a table with three columns: "Employee ID", "Name", and "Salary". The data rows are: "1, Russell Wilson, 50000", "2, Pete Carroll, 60000", and "3, Chris Carson, 40000". The prompt returns to "Brians-MacBook-Pro:Assignment07 brianlee\$".

```
Brians-MacBook-Pro:Assignment07 brianlee$ python3 Pickling_and_ErrorHandling.py
Employee ID,Name,Salary
1,Russell Wilson,50000
2,Pete Carroll,60000
3,Chris Carson,40000
Brians-MacBook-Pro:Assignment07 brianlee$
```

Figure 2. Running the Python Script in Terminal

A text file named "SeahawksSalaryPickled.txt" is open in a text editor. It contains the same output as the terminal window, including the header "Employee ID,Name,Salary" and the three data rows: "1,Russell Wilson,50000", "2,Pete Carroll,60000", and "3,Chris Carson,40000".

```
Employee ID,Name,Salary
1,Russell Wilson,50000
2,Pete Carroll,60000
3,Chris Carson,40000
```

Figure 3. Output file from the Pickling\_and\_ErrorHandling.py Script

### Summary

I enjoyed drilling into moving data in and out of data files as I believe that will be a huge part of how I will be utilizing Python. Pickling seems useful when transferring datasets from one environment to another, but from my research it sounded like pickling is Python specific and may cause some trouble even with different versions. Although this assignment required me to stretch my thinking as we had to write our own script, it would have also been nice to see how these concepts work within a real use case. Finally, I also found developing a GitHub web page to be extremely useful if you are looking to build out a portfolio or just document findings within your work - either way, extremely helpful to learn where I would have otherwise wouldn't have thought of doing.