Denise Albano
3/1/22
Foundations of Programming: Python
Assignment07
https://github.com/dla425/IntroToProg-Python-Mod07

# Pickling & Exception Handling Python Script

## Introduction

This paper will document the steps I took to create a new script that demonstrates how pickling and structured error handling work in Python. In order to complete the assignment, I watched the course video in module 7 by Randall Root and read chapter 7 in text book, as well as researching pickling and exception handling in Python on the web.

## Websites

The following are the additional websites I reviewed in order to complete the assignment:

https://www.datacamp.com/community/tutorials/pickle-python-tutorial
https://wiki.python.org/moin/UsingPickle
https://www.geeksforgeeks.org/understanding-python-pickling-example/
https://pythontic.com/modules/pickle/dump
https://pythontic.com/modules/pickle/load

https://docs.python.org/3/tutorial/errors.html
https://www.tutorialspoint.com/python/python_exceptions.htm
https://www.geeksforgeeks.org/multiple-exception-handling-in-python/
https://learnbyexample.github.io/100_page_python_intro/exception-handling.html
https://www.w3schools.com/python/ref_string_isnumberic.asp

## Creating the Python Script:

For this assignment I used PyCharm to create the python script. To start, I created a new project in PyCharm that uses the folder _PythonClass\Assignment07. Within this project, I created a new python script file named Assignment07.py (see figure 1).
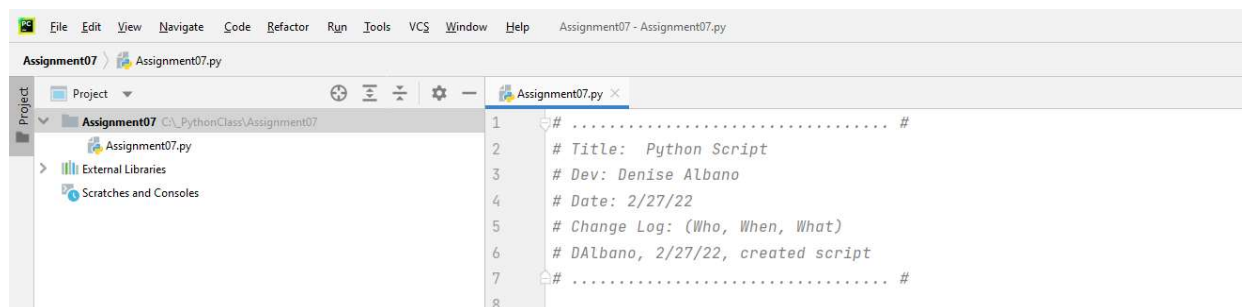


*Figure 1: New project and python script in PyCharm*

I started my script with a header that included information such as title, developer, date, and a change log. I started the script by importing pickle in Python. I then set up three sections, Data, Processing, and Presentation, to divide my code for this program.

## Data Section:

In the data section, I declared my variables and constants.

## Processing Section:

In the processing section, I created two functions to save and read the data to a binary file through the use of pickling.

To save data to the "ChildData.dat" file, I used the open() function with the 'ab' access mode to append the data to the binary file. I then used the pickle.dump() method using list_of_data (data to be pickled) and file_name (where to write the pickled data) as arguments in this method. I then used the close() function to close the file.

To read data from the "ChildData.dat" file, I used the open() function with the 'rb' access mode to read the data from the binary file. I then used the pickle.load() method to read the pickled objects from the file using file_name as the argument in this method and assigned it to list_of_data. I then used the close() function to close the file and return statement to return list_of_data.

## Presentation Section:

In order to have data to pickle, I needed to get some information from the user. To do this, I used a while loop to create a menu of choices for the user as a triple-quoted string. Using the input function, I created a string variable, strChoice, to capture the menu option from the user. I then used an if … elif statement to perform each choice the user entered.

For the first option, add data, I used input functions to capture the child's name (as a string) and age (as an integer) from the user. This data was used to create the list, lstChild. To deal with any data entered incorrectly by the user, I created a while True loop around input function. Using the try-except error handling method, if a number was entered for the child's name (therefore the isnumeric() method return true), then a error message was displayed to the user and the loop started over so that a correct name could be enter. If a negative number or a non-integer was entered for the child's age, then a ValueError was raised and a message to enter a positive number was displayed. Again the loop started over so that a correct age could be entered.

For the second option, save data, I called the save_data_to_file function using the variables, strFileName and lstChild, as the arguments. I then used a print() function to display to the user that the data had been saved.

For the third option, read data, in order to read the data from the file into the list object, I called the read_data_from_file function (using strFileName as the argument) within a print() function in order to display the contents of the file.

For the fourth option, exit program, if the user choose to exit the program a Goodbye statement is displaying using the print function.

To end the while loop, I used the break statement. The final code for this assignment is shown below:

```
# ............................................................. #
# Title:   Assignment07
# Description:   Script that includes pickling and exception handling
# Dev: Denise Albano
# Date: 2/27/22
# Change Log: (Who, When, What)
```

```python
# DAlbano, 2/27/22, created pickling script
# DAlbano, 3/1/22, added exception handling script
# ................................................................ #

import pickle  # This imports code from another code file!

# Data ---------------------------------------------------------- #
strFileName = 'ChildData.dat'
lstChild = []

# Processing ---------------------------------------------------- #
def save_data_to_file(file_name, list_of_data):
    objFile = open(file_name, "ab")
    pickle.dump(list_of_data, objFile)
    objFile.close()

def read_data_from_file(file_name):
    objFile = open(file_name, "rb")
    list_of_data = pickle.load(objFile)
    objFile.close()
    return list_of_data

# Presentation -------------------------------------------------- #
# Get name and age from user, then store it in a list object
while (True):
    print("""
    Menu of Options
    1) Add data
    2) Save data to file
    3) Read data from file
    4) Exit Program
    """)
    strChoice = str(input("Which option would you like to perform? [1 to 4] - "))
    print()

    if (strChoice.strip() == '1'):
        while True:
            try:
                strName = str(input("Enter the child's name: "))
                if strName.isnumeric():
                    raise Exception("Do not use numbers for child's name")
                break
            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')
                print()

        while True:
            try:
                intAge = int(input("Enter the child's age: "))
                if intAge < 0:
                    raise ValueError
                break
            except ValueError:
                print("Please use a postivie number")

        lstChild = [strName,intAge]

    # Store the list object into a binary file
```

```python
    elif (strChoice.strip() == '2'):      # Store the list object into a binary file
        save_data_to_file(strFileName, lstChild)
        print("Data saved")

    # Read the data from the file into a new list object and display the contents
    elif (strChoice.strip() == '3'):
        print(read_data_from_file(strFileName))

    # Exit the program
    elif (strChoice.strip() == '4'):

        print("Goodbye!")
        break
```

## Running the program:

I first ran the program entering in each menu choice to confirm that the pickling was working.  A screenshot of the output for each menu item is shown in Figure 2 and the binary file 'ChildData.bat" that was created in Figure 3.
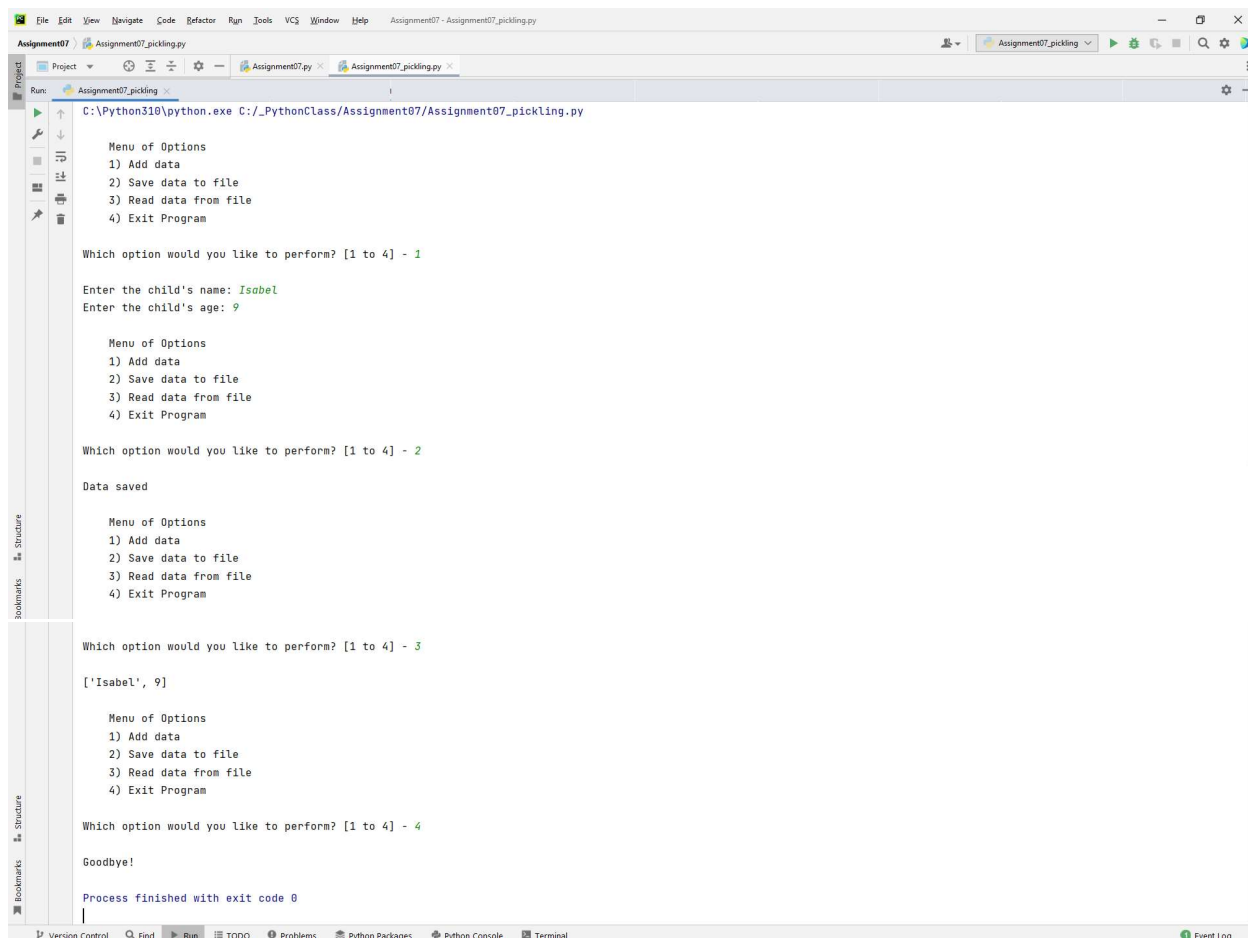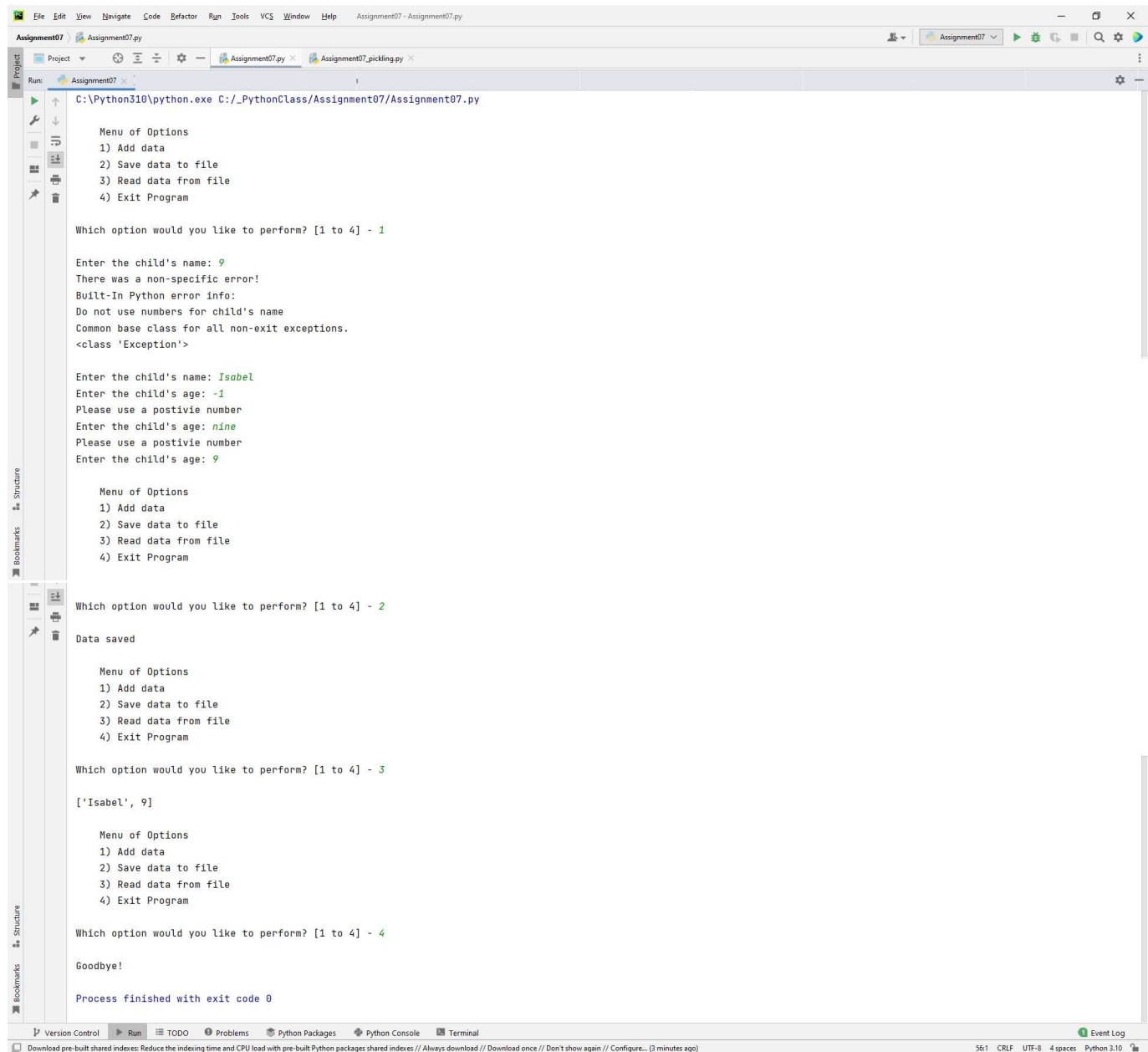


Figure 2:  Screenshot of the pickling portion of the script running PyCharm



Figure 3: A screenshot of the binary file created

Next I ran the program and specifically enter wrong information in order to trigger the exception handling portion of the script. A screenshot of that exception handling output is shown in figure 4.



*Figure 4: Screenshot of the pickling portion of the script running PyCharm*

## Post Files to GitHub

After creating a GitHub repository named "IntoToProg-Python-Mod07" in my account, I uploaded the Assignement07 files and committed the changes. As a last step, I shared the link to my GitHub repository on the canvas discussion board for peer review.

*Figure 5: Screenshot of files loaded into my GitHub account*

# Add a GitHub Webpage

## Summary

The information I found within these websites helped me to write the script to complete this