Nikita Daharia

June 1, 2022

Foundations of Programming: Python

Assignment 07

<https://github.com/nikitadaharia/IntroToProg-Python-Mod06>

**Pickling and Exception Handling (Try-Except):**

**Introduction:**

This paper aims to demonstrates pickling and exception handling feature in python. I explained these features by creating a python script that displays the birth year of a user when the user inputs their current age. The latest version of PyCharm Community version is used for this assignment on a Mac OS.

**Pickling:**

Python pickle is used for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk. What pickle does is that it “serializes” the object first before writing it to file. Pickling is a way to convert a python object (list, dict, etc.) into a character stream. The idea is that this character stream contains all the information necessary to reconstruct the object in another python script.

Text

Description automatically generated

***Figure 1. Python program to illustrate pickle.dump() (Source: geeksforgeeks.org)***

**Handling Exceptions in python (Try-Except):**

Error in Python can be of two types i.e. Syntax errors and Exceptions. Errors are the problems in a program due to which the program will stop the execution. On the other hand, exceptions are raised when some internal events occur which changes the normal flow of the program.

**Exceptions:** Exceptions are raised when the program is syntactically correct, but the code resulted in an error. This error does not stop the execution of the program, however, it changes the normal flow of the program.

**Try and Except Statement – Catching Exceptions:**

Try and except statements are used to catch and handle exceptions in Python. Statements that can raise exceptions are kept inside the try clause and the statements that handle the exception are written inside except clause.

In figure 2, the statements that can cause the error are placed inside the try statement (second print statement in our case). The second print statement tries to access the fourth element of the list which is not there and this throws an exception. This exception is then caught by the except statement.

Text

Description automatically generated

***Figure 2. Python program to handle simple runtime error (Source: geeksforgeeks.org)***

**Catching Specific Exception:**

A try statement can have more than one except clause, to specify handlers for different exceptions. Please note that at most one handler will be executed.

**Try with Else Clause:**

In python, you can also use the else clause on the try-except block which must be present after all the except clauses. The code enters the else block only if the try clause does not raise an exception.

**Finally Keyword in Python:**

Python provides a keyword finally, which is always executed after the try and except blocks. The final block always executes after normal termination of try block or after try block terminates due to some exception.

**Raising Exception:**

The raise statement allows the programmer to force a specific exception to occur. The sole argument in raise indicates the exception to be raised. This must be either an exception instance or an exception class (a class that derives from Exception).

This information about pickling and handling exceptions was gathered from <https://www.geeksforgeeks.org/pickle-python-object-serialization/>. The author successfully explained the fundamentals with numerous examples on this webpage. Examples are easier are understand as compared to plain text and this webpage was extremely insightful.

**Python script displaying the birth year of an individual based on their current age:**

My script asks the user to input their current age and based on that, it displays the birth year of the user.

I started by creating a new sub-folder called Assignment07 inside of the \_PythonClass folder (you created in Module 01) in the Documents folder on a Mac OS. Then I created a new project in PyCharm that uses the \_PythonClass\Assignment07 folder as its location.

Then global variables used throughout the code were defined as seen in Figure 1.

Text

Description automatically generated

***Figure 1. Snippet of the python script defining the global variables***

To keep my data processing distinct from my display, I employ custom functions. I have functions for saving and reading data. These not only pickle or unpickle data, but also contain a try/except block to capture failures. This can happen when there is no file with that name, like when reading a file, or if there is some subtle error with the file name or data (basically the arguments) for saving the data.

Text

Description automatically generated

***Figure 2. Pickling data using Try and Except***

Enter the age to list in memory option first asks the user to enter their current age, then asks user which operations they want to conduct on them, and then finally, it displays their year of birth as output.

Try/except blocks were used to catch if the user does not enter their age between 0 and 100 as shown in figure 3.

Text

Description automatically generated

***Figure 3. Exception handling in python using Try and Except***

The script is the class consists of functions that process the data to read, add, remove and write data to the file. Code was added after explaining the parameters to add, remove and write data to the list.

Text

Description automatically generated

***Figure 4. Snippet of the python script to add data to the file***

Text

Description automatically generated

***Figure 5. Snippet of the python script to display a menu of choices to the user as output***

Run the Script by right clicking on the file and choosing Run.

Text

Description automatically generated

***Figure 6. Snippet of output displayed in PyCharm Shell after running the python script.***

I chose the 1st option to enter my age to list in memory. On entering my age as 25, the program displayed my birth year as 1997 as shown in Figure 6.

Run the script on the Terminal window

***Text

Description automatically generated***

***Figure 7.*** ***Output displayed in Terminal window after running the python script.***

Lastly, the binary file age.dat was located and opened in the text editor to verify that it worked. The text editor could not render the data in a text format because the data is a serialized python object.

**Summary:**

Python is a simple yet powerful language programming language that runs on Windows, Linus/Unix, and Mac OS. I used PyCharm to create a python script that asks the user to input their current age and based on that, it displays the birth year of the user.

The script was run both in PyCharm and Terminal . Finally, the code was verified by locating the binary file age.dat and opening it in a text editor.