GJohnson

7/27/2024

Foundations Of Programming: Python

Assignment05

Create a Program

# Introduction

The goal of this document will be to demonstrate the use of python to create a program using while loops, programming menus, conditional logic, adds data processing using lists and files and add the use of data processing using dictionaries and exception handling all while using PyCharm as your IDE.

# Creating the script

## Acceptance Criteria

Understanding the acceptance criteria to ensure successful development is crucial to the success of any python project. The acceptance criteria for this script requires the following:

1. File Name: “Assignment05.py”
2. Script Header: “Name” and “Current Date”
3. Constants:
   1. **MENU: str**

---- Course Registration Program ----

Select from the following menu:

1. Register a Student for a Course

2. Show current data

3. Save data to a file

4. Exit the program

-----------------------------------------

* 1. **FILE\_NAME: str** is set to the value "Enrollments.json"

1. Variables:
   1. student\_first\_name: str & “”
   2. student\_last\_name: str & “”
   3. course\_name: str & “”
   4. json\_data: str & “”
   5. file: None
   6. menu\_choice: str & “”
   7. student\_data : dict & []
   8. students: list & []
2. Input & Output
   1. **Choice 1**: Prompts user to enter students first and last name and course\_name using input() and stores the inputs in the respective variables
   2. **Choice 2**: Outputs string using print() function
   3. Data collected from menu choice 1 is added to a two-dimensional list table (list of dictionaries)
   4. All data from the list is displayed when menu choice 2 is selected
3. File Processing
   1. When the program starts, the two-dimensional list is displayed
   2. **Choice 3**: program opens a file names “Enrollments.json” in write mode using the open() function. It writes the content of the students variables to the file using the dump() function then closes file using the close() method. It displays what was stored in the file.
   3. **Choice 4**: ends program
4. Error Handling
   1. The program provides structured error handling when the file is read into the list of dictionary rows.
   2. The program provides structured error handling when the user enters a first name.
   3. The program provides structured error handling when the user enters a last name.
   4. The program provides structured error handling when the dictionary rows are written to the file.
5. Output expectation
   1. Accepts user input for student’s first, last and course name
   2. Displays user input for student’s first, last and course name
   3. Saves user input for student’s first, last and course name as a comma-separated string file
   4. Program allows for user to input multiple registrations
   5. Program allows users to display multiple registrations
   6. Program allows users to save multiple registrations
   7. Program runs correctly in PyCharm and from console or terminal
6. Source Control
   1. The script file and the knowledge document are hosted on a GitHub repository.
   2. A link to the repository is included in the knowledge document.
   3. A link to the repository is included in the GitHub links forum.

### File Name and Header

In the first steps of creating this script, a [header](#_How_to_Download) was completed, detailing the title of the assignment, name and date (Figure 1). The file was then saved as “Assignment05\_Name”.

A screen shot of a computer

Description automatically generated

***Figure 1: Example of Python Header***

### Constants and Variables

To meet 3 and 4 of the acceptance criteria, [constants](#_What_is_a) and [variables](#_What_is_a_1) were written into the code with their correlating data types- to meet acceptance criteria, including the use of two dimensional list table (Figure 2).

A screenshot of a computer program

Description automatically generated

***Figure 2: Constants & Variables Development***

### Input and Outputs

For the input and output acceptance criteria, the input() function and output() functions were used. Each data type was labeled while also assigning values to our constants. Data collected from option 1 was stored in a two-dimensional list table. Error handling was also added to include errors for when characters other than alpha’s were used, as well as adding the ability to restart the program instead of ending due to an error (Figure 3).

For option number 2, the code recalls all inputs and provides the list to users (Figure 3).

A screenshot of a computer program

Description automatically generated

***Figure 3: Input, Output & Values***

### File Processing

Lastly, to complete the script and meet all acceptance criteria, methods to open, write and close were created in the script- using the open(), dump() and close() functions. Error handling was added to ensure if there were specific JSON and or other exceptions errors, that helpful information to users would be provided, while also preventing the program from crashing. Additional error help was added when starting the program. Code was added to ensure if the file was empty, the program was not ended. Details were added in case the file name was not found, and was built to provide clear feedback and create a new file (Figure 4.1).

With all elements present (Figure 4) use of command prompt was used to ensure that our code meets the output expectations (Figure 5) and the file was created with the correct values (Figure 6).

A screen shot of a computer program

Description automatically generated

***Figure 4: Example of Python File Processing***

A screenshot of a computer program code

Description automatically generated

***Figure 4.1: Example of Python File Processing***

A screenshot of a computer program

Description automatically generated

***Figure 5: Example of Command Prompt Use***

A close-up of a computer screen

Description automatically generated

***Figure 6: Enrollments JSON***

### Source Control

Source control for code is an important part of development and a way to ensure revisions are tracked while protecting in case a roll back to previous code is needed. For this project, GitHub was used, storing the repository [here](https://github.com/gjohnson0707/IntroToProg-Python-Mod05).

# Summary

The use of python to create programs with loops and conditional logic is an example of how python can serve in any business or individual use case. By using error coding and source control, you make your program more robust and user friendly to the end user.

# Appendix

## What is a “Header”

A python header can be used in various ways but serves the main purpose of providing details to a user related to the code. In above use (Figure 1), the header is used to document the title of the project, a description of what the code does and a log to track the changes to the project and code.

## What is a “Constant”

A constant is a method to store data or information in python. A constant is documented in all upper-case and is generally meant to be unchanged throughout a program’s lifecycle.

## What is a “Variable”

A variable is a method to store data or information in python. A variable is documented in all lower-case characters and is generally okay to update throughout a program’s lifecycle.