Nick Greco
Oct 21, 2024
Foundations of Programming: Python
Assignment02

Python Script for Enrollment Data

Introduction

Python is a high-level, versatile programming language renowned for its readability and simplicity, making it popular for web development, data analysis, artificial intelligence, and more. In this assignment, I will explain the steps involved in creating a script that prompts the user to enter a student's first and last name, outputs enrollment information, and stores this data within a .CSV file. This script demonstrates knowledge of input and output functions, data types, the utilization of variables and constants, string formatting, creation of and writing to .CSV files, and coding best practices.

Creating the Script

Overview

This Python script is designed to collect and record enrollment information for a course. Here's a summary of its main functionalities:

1. **Constants Definition:** The script defines several constants (denoted in capital letters), including the course name, price, state tax rate, total price (including tax), and the filename for storing enrollment data.
2. **User Input:** The script prompts the user to input the student's first and last names.
3. **Data Formatting:** The script formats the collected data into a CSV string, which includes the student's names, course name, course price, and total price.
4. **File Writing:** The script writes this formatted data to a CSV file named 'Enrollments.csv'.

Figure 1 below shows the completed script for this assignment.

```
# ------------------------------------------------------------------------------ #
# Title: Assignment02
# Desc:  This assignment demonstrates using constants, variables, operators,
#            formatting, and files.
# Change Log: (Who, What, When)
#   N.Greco, 10/16/2024, Created script.
#   N.Greco, 10/19/2024, Updated filename to match title.
# ------------------------------------------------------------------------------ #

# Define the Data Constants
COURSE_NAME: str = "Python 100"
COURSE_PRICE: float = 999.98
STATE_TAX: float = .09
TOTAL_PRICE: float = COURSE_PRICE + (COURSE_PRICE * STATE_TAX)
FILE_NAME: str = "Enrollments.csv"

# Define the Data Variables
student_first_name: str = ""
student_last_name: str = ""
course_name: str = ""
csv_data: str = ""
file_obj: object = None

# Get data from the user
student_first_name = input("Student's First Name: ")
student_last_name = input("Student's Last Name:  ")

# Present the data to the user
csv_data = f"{student_first_name},{student_last_name},{COURSE_NAME},"\
    f"{float(COURSE_PRICE):.2f},{float(TOTAL_PRICE):.2f}\n"
print(csv_data)

# Process the data to a file
file_obj = open(FILE_NAME, "w")
file_obj.write(csv_data)
file_obj.close()
# print("Data Recorded\n")  # Used for debugging.
```

Figure 1: 'Assignment02' Python Script

Concepts Demonstrated

This code demonstrates several key programming concepts:

1. **Constants:** Immutable values ("COURSE_NAME", "COURSE_PRICE", etc.) are defined and utilized.
2. **Variables:** It uses variables ("student_first_name", "student_last_name") to store user input.
3. **Data Types:** The script employs different data types, such as strings and floats.
4. **Arithmetic Operations:** It calculates the total price including tax using basic arithmetic.
5. **String Formatting:** The code formats strings for CSV output, ensuring proper formatting of prices.
6. **User Input:** It gathers data from the user using the input() function.
7. **File Handling:** It demonstrates how to create and write to a file using open(), write(), and close().

8. **Line Continuation**: Using the backslash (“\”) for line continuation in the second print statement to improve code readability, following PEP 8 recommendations.
9. **Comments and Documentation:** The code includes comments and a header for clarity and maintenance.
10. **Debugging Strategies:** A print() statement was used for debugging and was commented out in the final release.

These concepts form a foundation for understanding basic programming and user interaction in Python.

Running the Script

Using Python IDLE

Python IDLE (Integrated Development and Learning Environment) is a simple IDE that is installed with Python. It provides a user-friendly interface for writing, testing, and debugging code. Figure 2 shows the script run in this environment.

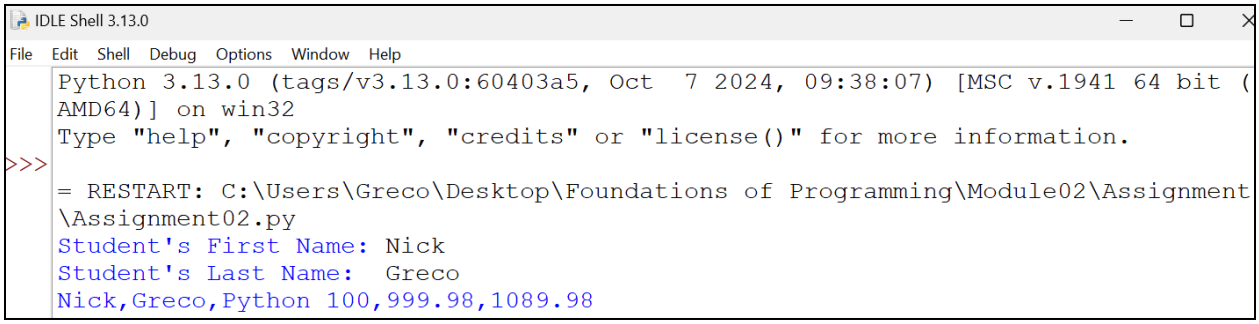

Figure 2: Script Run using IDLE

Using Windows Command Prompt

The Command Prompt is a command-line interpreter available in Windows operating systems. It allows users to execute commands, run scripts, and perform various system tasks without a graphical user interface. Figure 3 shows the process of navigating to and running the script in this environment.

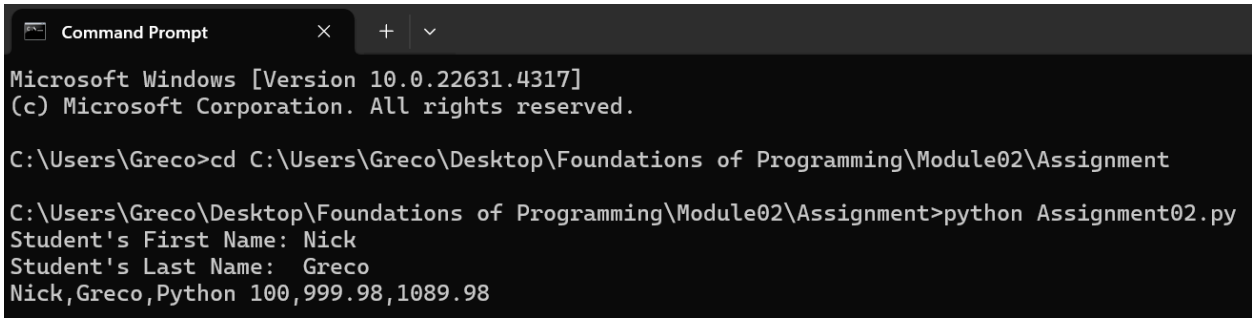

Figure 3: Script Run using Windows Command Prompt

Verifying Output Data File

Since the location of the data output file was not specified in the script, the file is stored relative to where the script was saved (as requested). Figure 4 shows 'Enrollments.csv' stored alongside 'Assignment02.py' after execution.

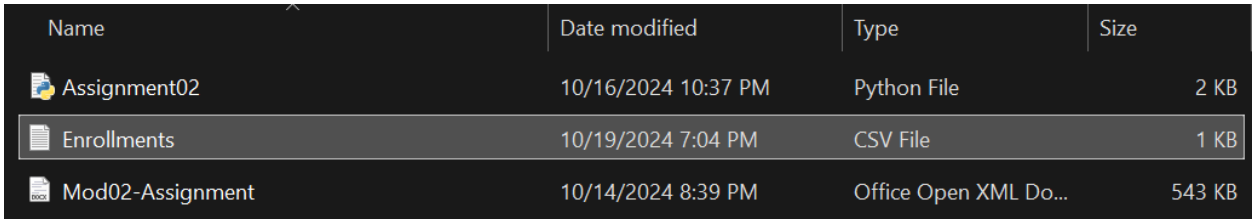

Figure 4: CSV File Generated in the Same Folder

Figure 5 shows the contents of the .CSV file when opened in Microsoft Notepad. The contents match the output within IDLE and command prompt.

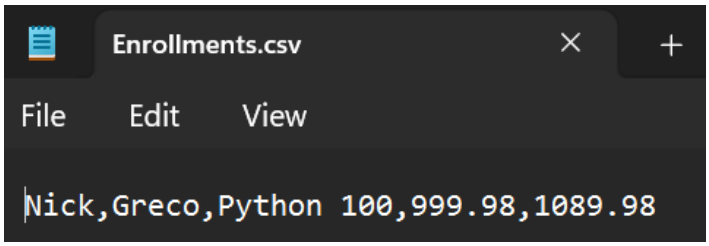

Figure 5: 'Enrollments.csv' Data Contents

Summary

In this assignment, I successfully created a Python script that collects and records student enrollment information, showcasing various programming concepts such as constants, variables, and file handling. By implementing user input and formatting data for a .CSV file, the script demonstrates the practical application of fundamental coding principles. Overall, this experience reinforced my understanding of Python and highlighted the importance of coding best practices for future projects.

Citations

OpenAI ChatGPT. (October 2024). https://chatgpt.com/: Aspects of this assignment were informed and created by queries I submitted to ChatGPT.