###############################################################

Student: Aditya Singh Sandhu

Course: CSC 500 – Principles of Programming

Module: 6 – Portfolio Assignment

###############################################################

Python3 Code ~

# COPY THIS LINE OF CODE - BEGINNING

def main():

room\_numbers = {

'CSC101': '3004',

'CSC102': '4501',

'CSC103': '6755',

'NET110': '1244',

'COM241': '1411'

}

instructors = {

'CSC101': 'Haynes',

'CSC102': 'Alvarado',

'CSC103': 'Rich',

'NET110': 'Burke',

'COM241': 'Lee'

}

meeting\_times = {

'CSC101': '8:00 a.m.',

'CSC102': '9:00 a.m.',

'CSC103': '10:00 a.m.',

'NET110': '11:00 a.m.',

'COM241': '1:00 p.m.'

}

course\_number = input("Enter a course number (e.g., CSC101): ")

if course\_number in room\_numbers:

print(f"Course Number: {course\_number}")

print(f"Room Number: {room\_numbers[course\_number]}")

print(f"Instructor: {instructors[course\_number]}")

print(f"Meeting Time: {meeting\_times[course\_number]}")

else:

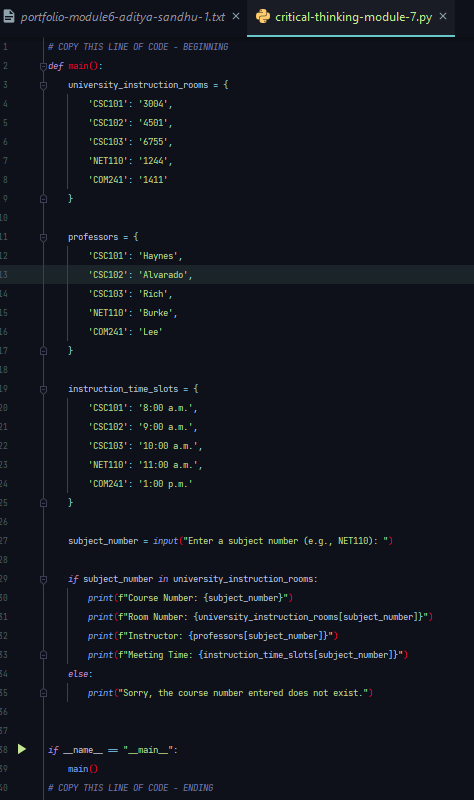
print("Sorry, the course number entered does not exist.")

if \_\_name\_\_ == "\_\_main\_\_":

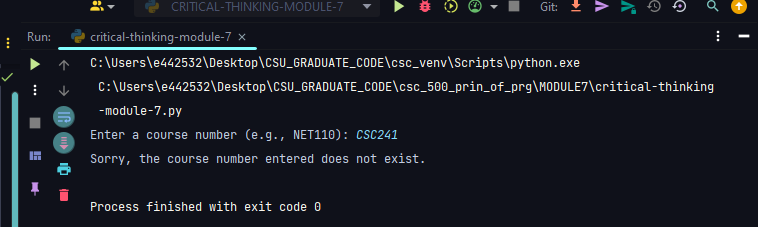
main()

# COPY THIS LINE OF CODE – ENDING

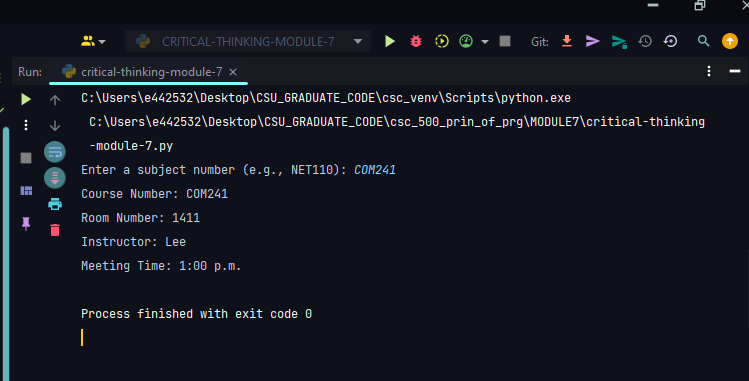
"Screenshot 1" "Main Function and within it, in subject\_number built-in input call and dictionary variables university\_instruction\_rooms professors and, instruction\_time\_slots.”



"Screenshot 2" "output of main function via the if ­­­­\_\_name\_\_ == “\_\_main\_\_“: function call of main() with Incorrect Input CSC241”



"Screenshot 3" "output of main function via the if ­­­­\_\_name\_\_ == “\_\_main\_\_“: function call of main() with correct Entry COM241”



The provided Python program is designed to manage and display information about university courses, including the room numbers where the courses are held, the instructors teaching them, and their meeting times. The program utilizes dictionaries to store and efficiently retrieve this information based on user input.

Dictionaries are a data structure in Python that store data as key-value pairs. In this program, dictionaries are used to map course numbers (keys) to their corresponding details like room numbers, instructors, and meeting times (values). university\_instruction\_rooms maps course numbers to the room numbers where each course is conducted. Professors maps course numbers to the names of the instructors teaching each course. instruction\_time\_slots maps course numbers to the times when each course is scheduled.

The program begins by prompting the user to enter a course number. This input is stored in the variable subject\_number. The program then checks whether the entered course number exists in the university\_instruction\_rooms dictionary. This is achieved using the in keyword, which checks for the presence of a key in a dictionary.

If the entered course number is found in the university\_instruction\_rooms dictionary, the program proceeds to retrieve and display the relevant information from all three dictionaries:

Next room number is retrieved from university\_instruction\_rooms using the entered course number as the key. Then the instructor is retrieved from the professors dictionary. Meeting time is retrieved from the instruction\_time\_slots dictionary. The use of formatted strings (f"...") allows the program to seamlessly insert variable values into output messages, making the output more readable and informative.

The Program has an if else statement to handle invalid input, screenshot 2 depicts that behavior. If the user enters a course number that does not exist in the university\_instruction\_rooms dictionary, the program displays a message indicating that the entered course number does not exist. This provides a simple form of error handling, ensuring that the program responds gracefully to invalid input.

Finally we have a modular Structure and main() function. The program is organized into a main() function, which is a common practice in Python programming to encapsulate the main logic of the script. This makes the code easier to read and maintain. The if \_\_name\_\_ == "\_\_main\_\_": statement at the end of the script ensures that the main() function is only executed when the script is run directly, not when it is imported as a module in another script. This is a standard Python idiom that enhances the reusability of the code.

Git hub link:

<https://github.com/65AR645ASAN/csc_500_prin_of_prg/blob/main/MODULE7/critical-thinking-aditya-sandhu-module-7.docx>

Work Cited

Python Software Foundation. (n.d.). Built-in functions. In Python 3.12.0 documentation. Retrieved August 26, 2024, from https://docs.python.org/3/library/functions.html

LearnPython.com. (n.d.). If-else Python practice problems. Retrieved August 26, 2024, from https://learnpython.com/blog/if-else-python-practice-problems/

Tutorialsteacher.com. (n.d.). Python if...elif...else statement. Retrieved August 26, 2024, from https://www.tutorialsteacher.com/python/python-if-elif

Programiz. (n.d.). Python if...elif...else statement. Retrieved August 26, 2024, from https://www.programiz.com/python-programming/if-elif-else