**Creating Python Programs**

Write a program that creates a dictionary containing course numbers and the room numbers of the rooms where the courses meet. The dictionary should have the following key-value pairs:

| **Key-Value Pairs: Room Number** | |
| --- | --- |
| **Course Number (key)** | **Room Number (value)** |
| CSC101 | 3004 |
| CSC102 | 4501 |
| CSC103 | 6755 |
| NET110 | 1244 |
| COM241 | 1411 |

The program should also create a dictionary containing course numbers and the names of the instructors that teach each course. The dictionary should have the following key-value pairs:

| **Key-Value Pairs: Instructors** | |
| --- | --- |
| **Course Number (key)** | **Instructor (value)** |
| CSC101 | Haynes |
| CSC102 | Alvarado |
| CSC103 | Rich |
| NET110 | Burke |
| COM241 | Lee |

The program should also create a dictionary containing course numbers and the meeting times of each course. The dictionary should have the following key-value pairs:

|  |  |
| --- | --- |
| **Key-Value Pairs: Meeting Time** | |
| **Course Number (key)** | **Meeting Time (value)** |
| 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. |

The program should let the user enter a course number and then it should display the course's room number, instructor, and meeting time.

**Submission:**

Compile and submit your pseudocode, source code, screenshots of the application executing the code, the results and GIT repository in a single document (Word is preferred).

**Pseudocode:**

1. Create three dictionaries:

* course\_rooms: Maps course numbers to room numbers.
* course\_instructors: Maps course numbers to instructor names.
* course\_times: Maps course numbers to meeting times.

1. Prompt the user to enter a course number.
2. Look up the entered course number in the three dictionaries:

* Get the room number from course\_rooms.
* Get the instructor from course\_instructors.
* Get the meeting time from course\_times.

1. Display the room number, instructor, and meeting time for the entered course number.
2. If the course number is not found, display an error message.

**Source Code:**

def main():

# Step 1: Create the dictionaries

course\_rooms = {

"CSC101": 3004,

"CSC102": 4501,

"CSC103": 6755,

"NET110": 1244,

"COM241": 1411

}

course\_instructors = {

"CSC101": "Haynes",

"CSC102": "Alvarado",

"CSC103": "Rich",

"NET110": "Burke",

"COM241": "Lee"

}

course\_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."

}

# Step 2: Get user input for course number

course\_number = input("Enter the course number (e.g., CSC101, CSC102): ").strip()

# Step 3: Look up the course number in the dictionaries and display the information

if course\_number in course\_rooms:

room\_number = course\_rooms[course\_number]

instructor = course\_instructors[course\_number]

meeting\_time = course\_times[course\_number]

# Step 4: Display the information

print(f"\nCourse: {course\_number}")

print(f"Room Number: {room\_number}")

print(f"Instructor: {instructor}")

print(f"Meeting Time: {meeting\_time}")

else:

# If the course number is not found

print("Course not found. Please check the course number and try again.")

# Call the main function to run the program

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

main()

**Screenshots:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated