Question 1:

Explanation

Prompt the User:

The get_user_input function prompts the user to enter their name, age, email, and favorite number. These inputs are then stored in a dictionary named user_info.

Store Inputs in a Dictionary:

The user_info dictionary contains the user inputs with keys: 'name', 'age', 'email', and 'favorite_number'.

Validate Email Format:

The validate_email function uses a regular expression to check if the email contains "@" and "." in the correct format. It returns True if the email is valid, otherwise False.

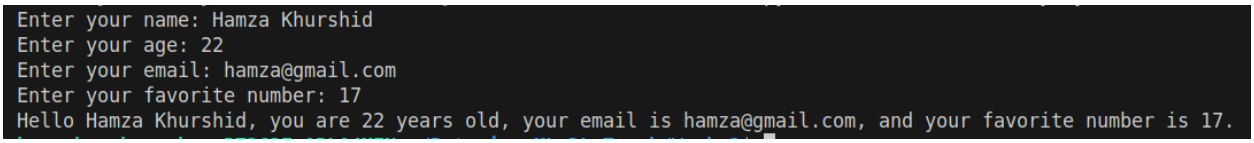
Display Formatted Message:

The display_message function formats and prints the message using the information stored in the user_info dictionary.

Main Function:

The main function coordinates the process. It calls get_user_input to get the user's input, validates the email, and if valid, displays the formatted message. If the email is invalid, it prints an error message.

This code ensures that the user is prompted for their details, the details are validated, and a message is displayed, all while adhering to the specified requirements.



Question 2:

Explanation

Function Definition:

is_even(number) is defined to take one parameter, number, which is expected to be an integer.

Check if the Number is Even:

The condition if number % 2 == 0: checks if the remainder when number is divided by 2 is 0, which indicates that the number is even.

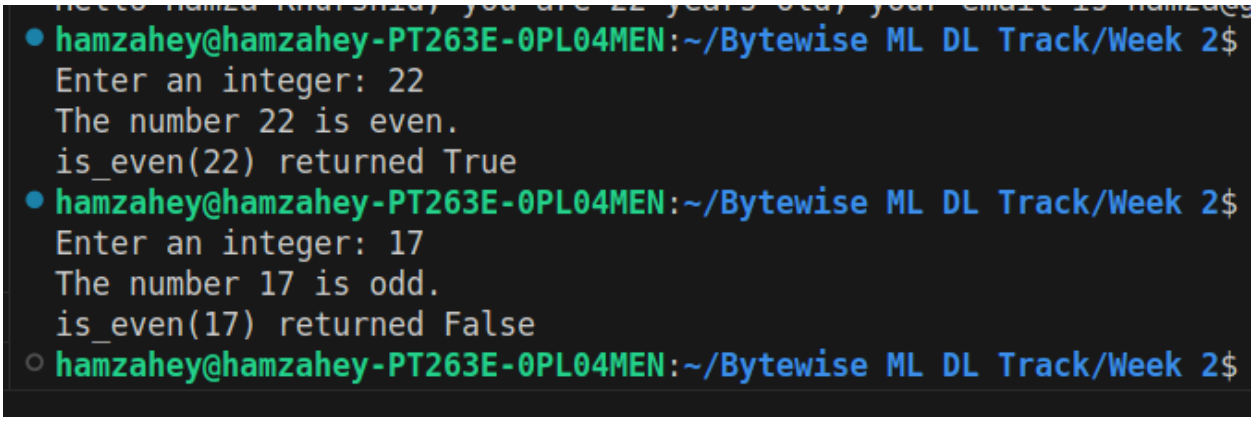
- If the number is even, it prints a message stating the number is even and returns True.
- If the number is not even (i.e., it is odd), it prints a message stating the number is odd and returns False.

Example Usage:

The user is prompted to enter an integer.

The is_even function is called with the entered integer, and the result is printed.

This code will correctly determine whether a number is even or odd, print the appropriate message, and return True or False based on the condition.



Question 3:

Function Definition:

- convert_temperature(temp, scale) is defined to take two parameters: temp (temperature value) and scale (a string representing the scale, either "C" for Celsius or "F" for Fahrenheit).

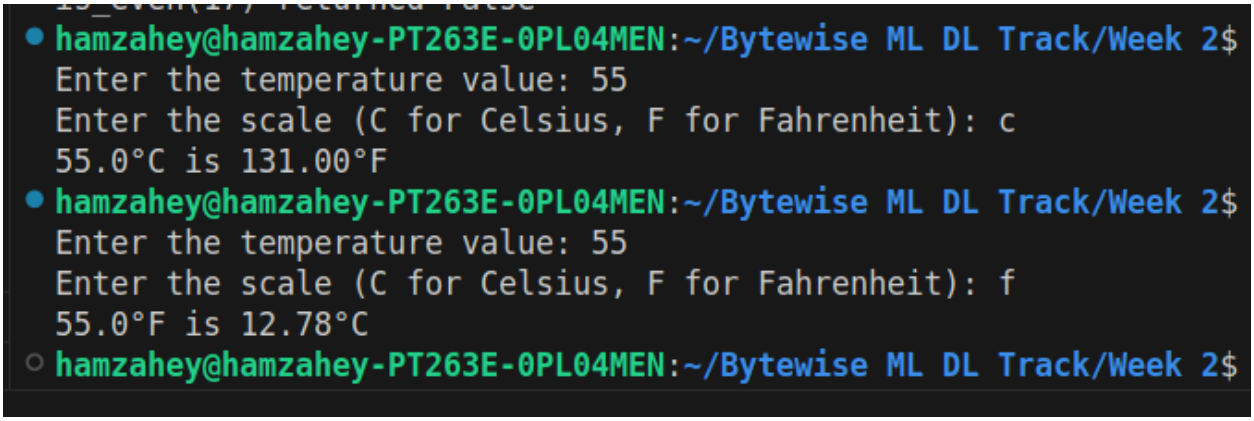
Check the Scale and Convert the Temperature:

- The if statement checks if the scale is "C" (Celsius). If true, it converts the temperature to Fahrenheit using the formula F = (C * 9/5) + 32, prints the converted temperature, and returns it.
- The elif statement checks if the scale is "F" (Fahrenheit). If true, it converts the temperature to Celsius using the formula C = (F - 32) * 5/9, prints the converted temperature, and returns it.
- If the scale is neither "C" nor "F", it prints an error message indicating an invalid scale and returns None.

Example Usage:

- The user is prompted to enter a temperature value and a scale.
- The convert_temperature function is called with the entered temperature and scale, and the converted temperature is displayed.

This code will correctly convert a given temperature from Celsius to Fahrenheit or vice versa and display the converted temperature.



Question 4:

Function Definition:

- find_max_min(numbers_list) is defined to take one parameter, numbers_list, which is a list of numbers.
- The function uses the built-in max() and min() functions to find the maximum and minimum numbers in the list, respectively.
- It returns a tuple containing the maximum and minimum numbers.

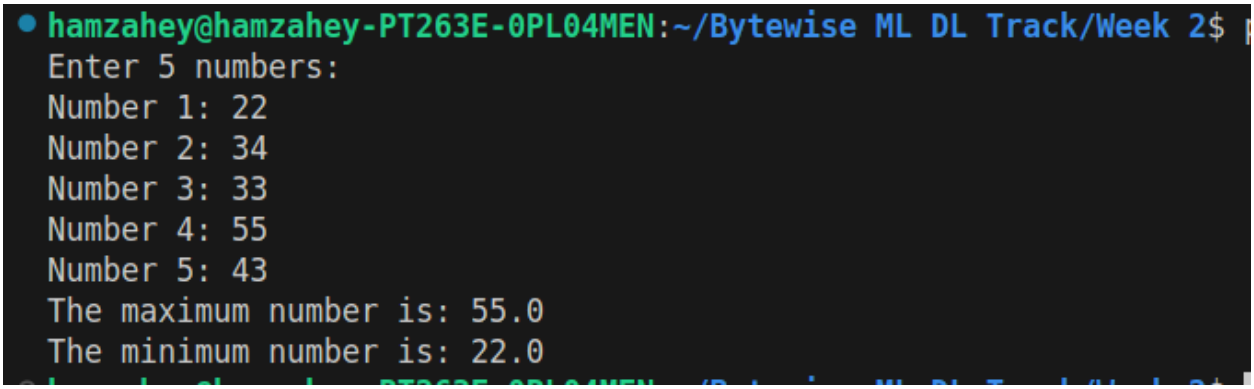
Main Function:

- The main function prompts the user to enter 5 numbers.
- It initializes an empty list called numbers.
- A for loop is used to prompt the user for 5 numbers, which are converted to floats and appended to the numbers list.
- The find_max_min function is called with the numbers list to get the maximum and minimum numbers.
- The maximum and minimum numbers are then printed.

Entry Point:

- The if __name__ == "__main__": block ensures that the main function is called only when the script is executed directly, not when it is imported as a module.

This code will prompt the user to enter 5 numbers, find the maximum and minimum numbers in the list using the find_max_min function, and display the results.



Question 5:

Explanation

Prompt the User for Details:

The main function uses a loop to prompt the user to enter the name, age, and grade for 3 students.
These details are stored in a tuple (name, age, grade) and appended to the students_list.

Convert List of Tuples to Dictionary:

A dictionary comprehension is used to convert students_list into students_dict.
The comprehension iterates over each tuple in the list, using the student's name as the key and the tuple (age, grade) as the value.

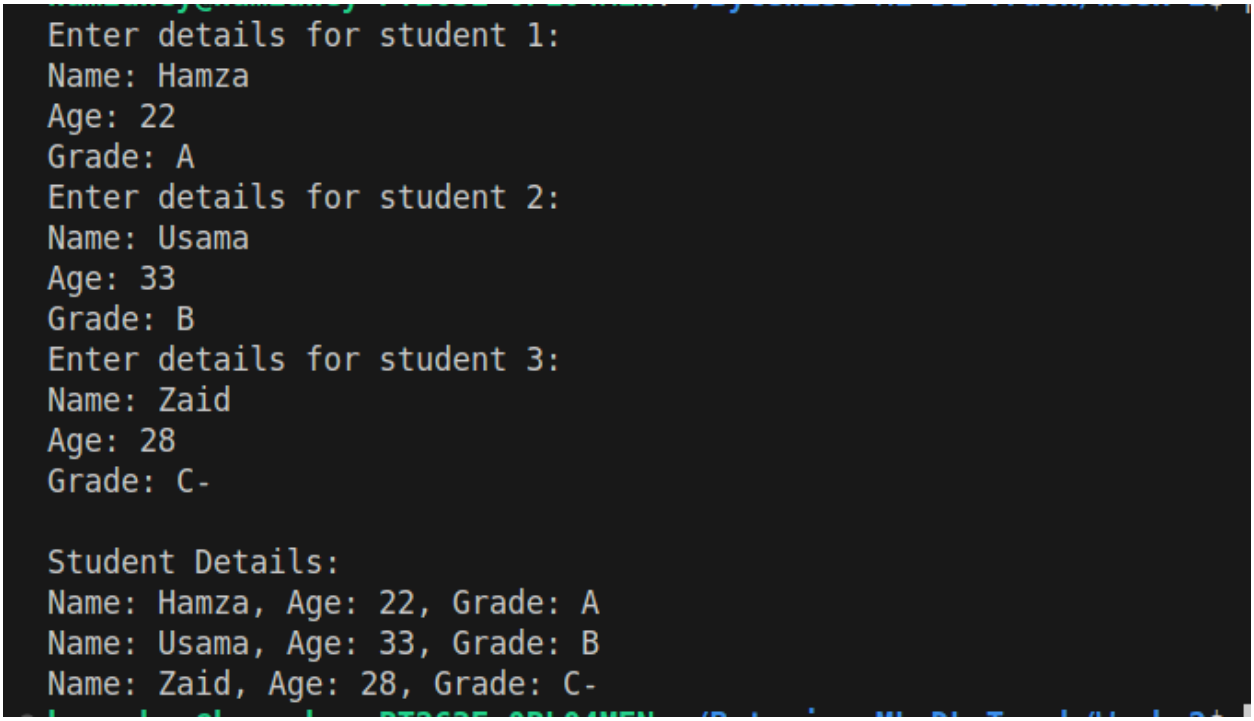
Display the Output:

The program prints the details of each student stored in the dictionary in a formatted manner.

Entry Point:

The if __name__ == "__main__": block ensures that the main function is called only when the script is executed directly.

This code will prompt the user to enter details for 3 students, store these details in a list of tuples, convert the list into a dictionary, and display the details in a user-friendly format.



Question 6:

Explanation

Function Definition:

The update_inventory function takes three parameters: inventory_dict, item, and quantity.
It updates the inventory by adding or removing the specified quantity. If the item does not exist in the inventory and the quantity is positive, it adds the item with the

specified quantity. If the quantity is negative and the item does not exist, it sets the quantity to 0.
It ensures that the quantity of any item does not go below zero.
It returns the updated inventory dictionary.

Main Function:

The main function initializes an inventory dictionary with at least 5 items.
It prompts the user to update the inventory by adding or removing quantities of 3 items. It uses a loop to get the item name and quantity to update from the user, then calls update_inventory to update the inventory.
It displays the initial and updated inventory.

Entry Point:

The if __name__ == "__main__": block ensures that the main function is called only when the script is executed directly.

This code will initialize an inventory with 5 items, allow the user to update the inventory by adding or removing quantities of 3 items, and display the updated inventory.