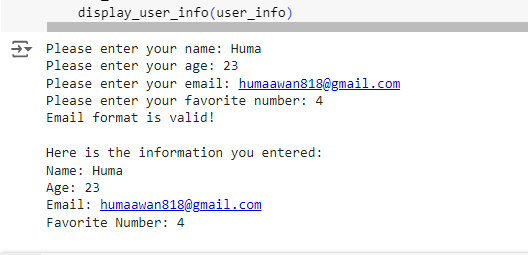
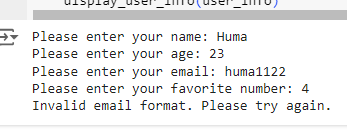
**Assignment#2**

I used Google Colab for coding to solve the given problems in the task.

**SOLUTION:1**

1. The **‘get\_user\_info() ’** function prompts the user to enter their name, age, email, and favorite number using the ‘input()’ function.
2. The inputs are stored in a dictionary with appropriate keys using the **‘{}’** syntax.
3. The email format is validated by checking if the **‘@’** and **‘.’** characters are present in the email string using the in operator. If the format is invalid, the function returns None.
4. The **‘display\_user\_info()’** function takes the **‘user\_info’** dictionary as an argument and uses f-strings to format a message using the variables.
5. The main program calls **‘get\_user\_info()’** and checks if the returned value is not None. If it's not None, it calls **‘display\_user\_info()’** with the **‘user\_info’** dictionary as an argument.

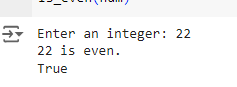


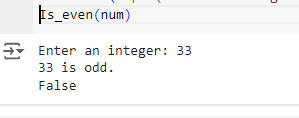


==================================================================

**Solution:2**

1. The function uses the modulo operator **(%)** to check if the number is even. If the remainder of the division of the number by **2** is **0**, then the number is even.
2. The function uses an **if statement** to check the condition. **If** the number is **even**, it prints a message indicating that the number is even and **returns** True.
3. **If** the number is **not even**, the **else** clause is executed, printing a message indicating that the number is **odd** and returning False.

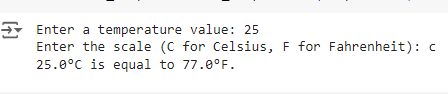


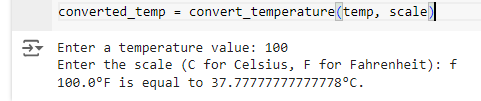


=========================================================

**Solution:3**

1. The function uses an if statement to check the scale. If the scale is **"C"** (Celsius), it converts the temperature to Fahrenheit using the formula **(temp \* 9/5) + 32**.
2. If the scale is **"F"** (Fahrenheit), it converts the temperature to Celsius using the formula **(temp - 32) \* 5/9**.
3. If the input scale is invalid, the function prints an error message and returns **None**.

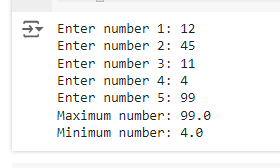
****

****

=============================================================

**Solution:4**

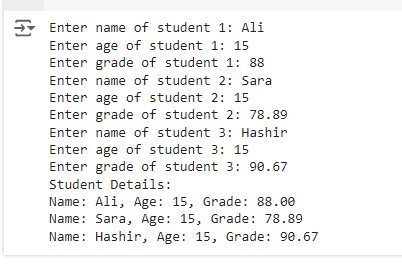
1. The **find\_max\_min** function takes a list of numbers as input.
2. The function uses the built-in **max** and **min** functions to find the maximum and minimum numbers in the list, respectively.
3. The function **returns** a tuple containing both the maximum and minimum numbers.
4. The code prompts the user to enter **5** numbers, storing them in a list called **numbers**.
5. The **find\_max\_min** function is called with the **numbers** list as input, and the returned maximum and minimum numbers are stored in the **max\_num** and **min\_num variables**, respectively.
6. The code displays the maximum and minimum numbers using **f-strings**.



===================================================================

**Solution:5**

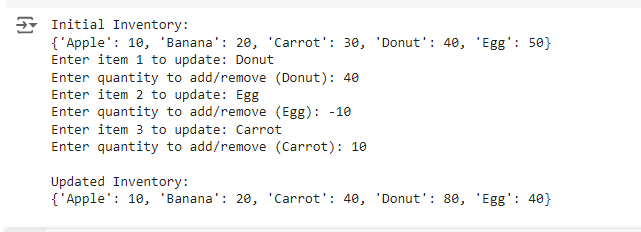
1. The code prompts the user to enter the details of **3** students: **name**, **age**, and **grade**.
2. The details are stored in a **list** of tuples, with each tuple containing the name, age, and grade of a student.
3. The list of tuples is converted into a dictionary using a dictionary comprehension. The student name is used as the key, and the tuple (age, grade) is used as the value.



========================================================================

**Solution:6**

1. The **update\_inventory** function takes a dictionary **inventory\_dict**, an item name **item**, and a quantity, **quantity** to add or remove.
2. The function checks if the item is in the inventory dictionary. If it is, it updates the quantity by adding or removing the specified quantity.
3. The function ensures that the quantity does not go below zero by setting it to zero if it would otherwise be negative.

****

========================================================================