**Use Cases:**

1. Consider a scenario where you're developing a system to manage training sessions for a software development company. The JSON snippet provided represents one such training session named 'Python Training.' In this scenario, you have multiple training sessions with different instructors, dates, and participants. For this you need to handle booleans, integers, strings, floats a lists and dictionaries.

**JSON-**

{

"name": "Python Training",

"date": "April 19, 2024",

"completed": True,

"instructor": {

"name": "XYZ",

"website": "http://pqr.com/"

},

"participants": [

{

"name": "Participant 1",

"email": "email1@example.com"

},

{

"name": "Participant 2",

"email": "email2@example.com"

}

]

}

2. Submit your answers as a Python script file (.py) named greek\_gods\_exam.py.

CSV Tables:

greek\_gods.csv:

God,Domain,Symbol,Age

Zeus,King of the Gods,Thunderbolt,10000

Poseidon,God of the Sea,Trident,9000

Hera,Queen of the Gods,Peacock,8500

Athena,Goddess of Wisdom,Owl,8000

Hades,God of the Underworld,Helmet,9500

greek\_goddesses.csv:

Goddess,Domain,Symbol,Age

Aphrodite,Goddess of Love,Dove,7000

Artemis,Goddess of the Hunt,Bow and Arrow,7500

Demeter,Goddess of Agriculture,Wheat,8000

Hestia,Goddess of the Hearth,Fire,8200

Persephone,Queen of the Underworld,Pomegranate,7800

Questions:

1.Merge the data from greek\_gods.csv and greek\_goddesses.csv based on a common field and create a new table that includes information about both gods and goddesses.

2.Filter the merged table to only include gods and goddesses who are older than 8000 years, then sort them based on their ages in descending order.

3.Join the two tables based on the "Domain" field and calculate the average age of gods and goddesses in each domain.

4.Determine which god/goddess has the highest age, and then find out if they are a god or a goddess.

5.Create a new column in each table called "Age\_Group" and categorize the gods/goddesses into groups such as "Young" (age < 5000), "Middle-aged" (age between 5000 and 8000), and "Old" (age > 8000).

6.Compare the average ages of gods and goddesses. Is there a significant age difference between them? If yes, which group tends to be older?

7.Write a Python program using for loop to iterate over the "Age" column of the merged table (after merging the gods and goddesses tables) and print out the names of gods/goddesses who are older than 8000 years.

8.Write a Python program to find the oldest god/goddess from the merged table (after merging the gods and goddesses tables) by iterating through the "Age" column using a while loop. Print out the name of the oldest god/goddess and their age.