Practice Questions - Strings & Data Structures

Question 1:

Given an original list consiting of values: 22, 41, 33, 66, 8, 12 and 90. Use **append(), extend(), insert(), and concatenation** to add the values 32, 45, 16 and 11 to the list.

In [ ]:

Question 2:

Write a Python program that tests whether a list is empty or not. Repeat the task for testing an empty string.

In [ ]:

Question 3:

- How many times does "dream" appear in the text?
- Where is this text from?
- Create a python code to print the text.

Dream on\ Dream on\ I dream on\ Dream a little, I'll dream on\ Dream on\ I dream on\ I dream on

Dream a little, I'll dream on\ Dream on\ Dream on\ Dream on\ I'll dream on\ Dream on\ Dream on\ I dream on

In [ ]:

Question 4:

Create a dictionary, with keys being the name of employees: Oliver, Myers, Noah and Fallon.

Add the values age (random - between 40 and 60), income (random between 100,000 and 150,000) and rating (random between 3.0 and 5.0)

In [ ]:

Question 5:

Create the following student data into 2 sub dictionaries based on the scores. (score <= 10 & score > 10)

In [ ]:
```
students = {"student_1" : 13 , "student_2" : 17 , "student_3" : 9 , "student_4" : 15 ,
```

```
            "student_7" : 16 , "student_8" : 12 , "student_9" : 13 , "student_10" : 15
            "student_13" : 10 , "student_14" : 12 , "student_15" : 13 , "student_16" :
            "student_19" : 9 , "student_20" : 17 ,}
```

Question 6:

Write a python program that asks the user to enter an integer n and return a dictionary whose keys are integers 1, 2, 3, ... n and whose values are 1! , 2! , 3! , ... , n!

In [ ]:

Question 7:

Write a program to identify the largest word in a string.

In [ ]:

Question 8:

Given an integer array, write a program to identify all the rows with values greater than the mean of the entire data.

In [ ]:

Question 9:

Given an array of strings, write a program to create a new array which contains the lengths of the strings in ascending order

In [ ]:

Question 10:

Write a program to calculate the sum of the diagonal elements of a **n X n** matrix.

In [ ]: