Functions and Methods Homework

Complete the following questions:

Write a function that computes the volume of a sphere given its radius.

In [13]:

```
import math
def vol(rad):
    if rad < 0:
        return "Radius cannot be negative."
    else:
        volume = (4/3) * math.pi * rad**3
        return volume
```

In [14]:

```
radius = 6.0
volume = vol(radius)
print(f"The volume of the sphere with radius {radius} is {volume:.2f}")
```

```
The volume of the sphere with radius 6.0 is 904.78
```

Write a function that checks whether a number is in a given range (Inclusive of high and low)

In [15]:

```
def ran_check(num, low, high):
    if low <= num <= high:
        return True
    else:
        return False
```

In [16]:

```
num_to_check = 5
low_range = 2
high_range = 8

if ran_check(num_to_check, low_range, high_range):
    print(f"{num_to_check} is within the range [{low_range}, {high_range}]")
else:
    print(f"{num_to_check} is not within the range [{low_range}, {high_range}]
```

```
5 is within the range [2, 8]
```

If you only wanted to return a boolean:

In [17]:

```
def ran_bool(num, low, high):
    return low <= num <= high
```

In [19]:

```
num_to_check = 9
low_range = 2
high_range = 8

is_within_range = ran_bool(num_to_check, low_range, high_range)

print(is_within_range)
```

```
False
```

In [9]:

```
ran_bool(3,1,10)
```

Out[9]:

```
True
```

Write a Python function that accepts a string and calculate the number of upper case letters and lower case letters.

```
Sample String : 'Hello Mr. Rogers, how are you this fine Tuesday?'
Expected Output :
No. of Upper case characters : 4
No. of Lower case Characters : 33
```

If you feel ambitious, explore the Collections module to solve this problem!

In [20]:

```
def up_low(s):
    upper_count = 0
    lower_count = 0

    for char in s:
        if char.isupper():
            upper_count += 1
        elif char.islower():
            lower_count += 1

    return upper_count, lower_count
```

In [21]:

```
sample_string = 'Hello Mr. Rogers, how are you this fine Tuesday?'
upper_count, lower_count = up_low(sample_string)

print("No. of Upper case characters:", upper_count)
print("No. of Lower case Characters:", lower_count)
```

```
No. of Upper case characters: 4
No. of Lower case Characters: 33
```

Write a Python function that takes a list and returns a new list with unique elements of the first list.

```
Sample List : [1,1,1,1,2,2,3,3,3,3,4,5]
Unique List : [1, 2, 3, 4, 5]
```

In [22]:

```
def unique_list(l):
    unique_elements = []
    for item in l:
        if item not in unique_elements:
            unique_elements.append(item)
    return unique_elements

sample_list = [1, 1, 1, 1, 2, 2, 3, 3, 3, 3, 4, 5]
unique_result = unique_list(sample_list)

print("Unique List:", unique_result)
```

```
Unique List: [1, 2, 3, 4, 5]
```

In [14]:

```
unique_list([1,1,1,1,2,2,3,3,3,3,4,5])
```

Out[14]:

```
[1, 2, 3, 4, 5]
```

Write a Python function to multiply all the numbers in a list.

```
Sample List : [1, 2, 3, -4]
Expected Output : -24
```

In [24]:

```
def multiply(numbers):
    product = 1

    for number in numbers:
        product *= number

    return product
```

In [25]:

```
sample_list = [1, 2, 3, -4]
result = multiply(sample_list)

print("Product:", result)
```

```
Product: -24
```

In [18]:

```
multiply([1,2,3,-4])
```

Out[18]:

```
-24
```

Write a Python function that checks whether a passed string is palindrome or not.

Note: A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run.

In [26]:

```
def palindrome(s):
    s = s.replace(" ", "").lower()

    return s == s[::-1]
```

In [27]:

```
sample_string = "nurses run"
if palindrome(sample_string):
    print(f"'{sample_string}' is a palindrome.")
else:
    print(f"'{sample_string}' is not a palindrome.")
```

```
'nurses run' is a palindrome.
```

In [20]:

```
palindrome('helleh')
```

Out[20]:

```
True
```

Hard:

Write a Python function to check whether a string is pangram or not.

```
Note : Pangrams are words or sentences containing every letter of the
alphabet at least once.
For example : "The quick brown fox jumps over the lazy dog"
```

Hint: Look at the string module

In [28]:

```
import string

def ispangram(str1, alphabet=string.ascii_lowercase):

    str1 = str1.lower().replace(" ", "")

    unique_letters = set(str1)

    return set(alphabet) == unique_letters
```

In [29]:
```
sample_string = "The quick brown fox jumps over the lazy dog"
if ispangram(sample_string):
    print(f"'{sample_string}' is a pangram.")
else:
    print(f"'{sample_string}' is not a pangram.")
```

```
'The quick brown fox jumps over the lazy dog' is a pangram.
```

In [30]:
```
ispangram("The quick brown fox jumps over the lazy dog")
result = ispangram("The quick brown fox jumps over the lazy dog")
print(result)
```

```
True
```

In [33]:
```
string.ascii_lowercase
import string
lowercase_alphabet = string.ascii_lowercase

print(lowercase_alphabet)

```

```
abcdefghijklmnopqrstuvwxyz
```

####**Great Job!**