In [7]:
```
def vol(rad):

    return (4/3)*(22/7)*(rad**3)

```

In [8]:
```
vol(2)
```

Out[8]:
```
33.52380952380952
```

In [22]:
```
def ran_check(num,low,high):
    if num in range(low,high+1):
        print(f'{num} in range of {low} and {high}')

    else:
        print(False)
```

In [23]:
```
ran_check(5,2,7)
```

```
5 in range of 2 and 7
```

In [36]:
```
def up_low(s):
    low_n1 = 0
    high_n1=0
    for char in s:
        if char.islower():
            low_n1 += 1
        elif char.isupper():
            high_n1 += 1
        else:
            pass
    print(f'Number of lower charecters : {low_n1}')
    print(f'Number of higher charecters : {high_n1}')

```

In [37]:
```
s = 'Hello Mr. Rogers, how are you this fine Tuesday?'
up_low(s)
```

```
Number of lower charecters : 33
Number of higher charecters : 4
```

In [40]:
```
def unique_list(lst):
    return list(set(lst))
```

In [41]:
```
unique_list([1,1,1,1,2,2,3,3,3,3,4,5])
```

Out[41]:
```
[1, 2, 3, 4, 5]
```

In [46]:
```
def multiply(numbers):
    total = 1
    for num in numbers:
        total = total * num
    return total
```

In [47]:
```
multiply([1,2,3,-4])
```

Out[47]: -24

In [56]:
```
def palindrome(s):
    for num in s:
        if num == num[::-1]:
            return True
        else:
            return False
```

In [57]:
```
palindrome('helleh')
```

Out[57]: True

In [83]:
```
import string

def ispangram(str1, alphabet=string.ascii_lowercase):
    bet = set(alphabet)
    str1 = str1.replace(" ","")
    str1 =str1.lower()
    str1 = set(str1)
    return  str1 == bet
```

In [84]:
```
ispangram("The quick brown fox jumps over the lazy dog")
```

Out[84]: True

In [81]:
```
import string

def ispangram(str1, alphabet=string.ascii_lowercase):
    # Create a set of the alphabet
    alphaset = set(alphabet)

    # Remove spaces from str1
    str1 = str1.replace(" ",'')

    # Lowercase all strings in the passed in string
    # Recall we assume no punctuation
    str1 = str1.lower()

    # Grab all unique letters in the string as a set
    str1 = set(str1)

    # Now check that the alpahbet set is same as string set
    return str1 == alphaset
```

In [82]:
```
1 ispangram("The quick brown fox jumps over the lazy dog")
```

Out[82]: True

In [ ]:
```
1
```