Exercise 1:
Write a Python program to filter out the vowels from a given list of characters using the filter() function.

```
def list1(n):
    vovels = ['a', 'e', 'i', 'o', 'u']
    if n in vovels:
        return True
    else:
        return False
list2=['a','b','c','d','e','f','i','o','u']
result=list(filter(list1,list2))
print(result)
vovels1=list(filter(lambda i:i in ['a', 'e', 'i', 'o', 'u'],list2))
print(vovels1)


#print(list(even1))
```

```
nbda (i)
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\python.exe
['a', 'e', 'i', 'o', 'u']
['a', 'e', 'i', 'o', 'u']

Process finished with exit code 0
```

Exercise 2:
Write a Python program to filter out the prime numbers from a given list of integers using the filter() function

```
def prime(n):
    if n <2:
        return False
    for i in range(2, int(n**0.5)+1):
        if n%i==0:
            return False
    return True
numbers=[1,2,3,4,5,6,7,8,9,10,11,12,13,15]
primenumber=list(filter(prime,numbers))
print(primenumber)
prime_lamda=list(filter(lambda x: prime(x),numbers))
print(prime_lamda)
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scri
[2, 3, 5, 7, 11, 13]
[2, 3, 5, 7, 11, 13]
```

Exercise 3:
Write a Python program to filter out the strings with length greater than 5 from a given list of strings using the filter() function.

```
def str(str_list):
    return list(filter(str_length,str_list))
def str_length(string):
    return len(string) <=5
str_list=['apple','orange','banana','kivi','jack fruit','ladu','mala']
filterlist=str(str_list)
print(filterlist)
#Lambda
filterlist=list(filter(lambda x:len(x) <=5, str_list))
print(filterlist)
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\python.exe" "
['apple', 'kivi', 'ladu', 'mala']
['apple', 'kivi', 'ladu', 'mala']

Process finished with exit code 0
```

Exercise 4:
Write a Python program to convert a list of strings to uppercase using the map() function.

```
def convert_upper(str_list):
    return list(map(str.upper,str_list))
my_list=['apple','orange','banana','kivi','jack fruit','ladu','mala']
upper_list = convert_upper(my_list)
print(upper_list)
```

```
nvert_upper()
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\python.exe"
['APPLE', 'ORANGE', 'BANANA', 'KIVI', 'JACK FRUIT', 'LADU', 'MALA']

Process finished with exit code 0
```

Exercise 5:
Write a Python program to square each number in a given list using the map() function.

```
def squire(num):
    return num ** 2

my_numbers=[1,2,3,4,5,6,7,8]
squire_num=list(map(squire,my_numbers))
print(squire_num)
#Lambda
squired_numbers = list(map(lambda x:x**2, my_numbers))
print(squired_numbers)
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scrip
[1, 4, 9, 16, 25, 36, 49, 64]
[1, 4, 9, 16, 25, 36, 49, 64]

Process finished with exit code 0
```

Exercise 6:
Write a Python program to calculate the length of each string in a given list of strings using the map() function.

```
my_list=['apple','orange','banana','kivi','jack fruit','ladu','mala']
length = list(map(len,my_list))
print(length)
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\python.exe"
[5, 6, 6, 4, 10, 4, 4]

Process finished with exit code 0
```

Exercise 7:
Write a Python program to evaluate a given mathematical expression using the eval() function.
expression = "3 * 5 + 2"

```
expression = "3 * 5 + 2"
result=eval(expression)
print(result)
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\ve
17

Process finished with exit code 0
```

Exercise 8:
Write a Python program to evaluate a given logical expression using the eval() function.
expression = "5 > 2 and 3 < 4"

```
expression = "5 > 2 and 3 < 4"
result=eval(expression)
print(result)
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\v
True

Process finished with exit code 0
```

Exercise 9:
Write a Python program to calculate the sum of elements in a list using the reduce() function.

```
from functools import reduce
def add_list(x,y):
    return x + y
my_list = [1,2,3,4,5,6,7,8,9,10]
sum_= reduce(add_list,my_list)
print(f'Sum of list is = {sum_}')
sum_list=reduce(lambda x,y:x+y,my_list)
print(f'sum of list is = {sum_list}')
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProje
Sum of list is = 55
sum of list is = 55

Process finished with exit code 0
```

Exercise 10:
Write a Python program to calculate the product of elements in a list using the reduce() function.

```python
from functools import reduce

def reduce_list(x,y):
    return x * y
my_list = [1,2,3,4,5,6,7,8,9,10]
result = reduce(reduce_list,my_list)
print(f'products of the elements in the list = {result}')
#Lambda
result_lambda=reduce(lambda x,y:x*y,my_list)
print(f'products of the elements in the list = {result_lambda}')
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\python.
products of the elements in the list = 3628800
products of the elements in the list = 3628800

Process finished with exit code 0
```

Exercise 11:
Write a Python program to find the maximum element in a list using the reduce() function.

```python
from functools import reduce
def reduce_list(a,b):
    if a > b:
        return a
    else:
        return b

my_list = [1,2,3,4,5,6,7,8,9,10]
result1= reduce(reduce_list,my_list)
print(result1)
#Lambda
result = reduce(lambda a,b: a if a > b else b, my_list)
print(result)
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\
10
10

Process finished with exit code 0
```

Exercise 12:
Write a generator function that takes a string and yields each word in the string

```
def word_generator(string):
    for word in string.split():
        yield word
my_str= "Hellow ! This is only a word generator function"
for word in word_generator(my_str):
    print(word)
```

```
ord_generator() › for word in string.split()
Python_Class ×
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\pyt
Hellow
!
This
is
only
a
word
generator
function

Process finished with exit code 0
```

Exercise 13:
Write a generator function that takes a list of integers and yields only the even numbers

```
def even_numbers(lst):
    for num in lst:
        if num % 2 == 0:
            yield num

my_list = [1,2,3,4,5,6,7,8,9,10]
for even_num in even_numbers(my_list):
    print(even_num)
```

```
r even_num in even_numbers(my...
Python_Class ×
"C:\Users\VINOD VM\PycharmProjects\pythonProject\v
2
4
6
8
10

Process finished with exit code 0
```

Exercise 14:
Write a generator function that takes a list of strings and yields only the strings that start with a vowel.

```
def vovels_start(str):
    vovels=['a','e','i','o','u']
    for string in str:
        if string[0].lower() in vovels:
            yield string
my_list=['apple','orange','banana','kivi','jack fruit','Egipt','India']
for vovels_str in vovels_start(my_list):
    print(f'Vovels start in list = {vovels_str}')
```

```
Python_Class
"C:\Users\VINOD VM\PycharmProjects\pythonProject\venv\Scripts\python.exe" "C:\U
Vovels start in list = apple
Vovels start in list = orange
Vovels start in list = Egipt
Vovels start in list = India

Process finished with exit code 0
```