In [1]:
```
def show_employee(name, salary=9000):
    print("Name:", name, "salary:", salary)
```

In [2]:
```
show_employee("Ben", 12000)
show_employee("Jessa")
```

```
Name: Ben salary: 12000
Name: Jessa salary: 9000
```

In [6]:
```
evenNums=[]
for i in range(4, 31):
    if i%2==0:
        evenNums.append(i)
print(f"Even numbers between 4&41 are {evenNums}")
```

```
Even numbers between 4&41 are [4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30]
```

In [7]:
```
def outer_add(a, b):
    def inner_add(x, y):
        return x + y
    totalsum = inner_add(a, b) +5
    return totalsum
```

In [8]:
```
result=outer_add(15,20)
print(result)
```

```
40
```

In [9]:
```
L = [1, 2, 3, 4, 5]
check_element = lambda lst, element: "Element is Present in the list" if any(i == element for i in lst) else "Element is NOT Pres
```

In [11]:
```
element = 7
result = check_element(L, element)
print(result)
```

```
Element is NOT Present in the list
```

In [14]:
```
sentence=input("Enter your sentence: ")
letters=0
digits=0
for i in sentence:
    if i.isalpha():
        letters+=1
    if i.isdigit():
        digits+=1
print(f"Digits: {digits}\nLetters: {letters}")
```

```
Enter your sentence: jfdgsdiawherf123565fgc
Digits: 6
Letters: 16
```

In [16]:
```
inputCharacter = input("Enter a sequence of characters: ")
uppercase=map(lambda x: x.upper(), inputCharacter)
lowercase=map(lambda x: x.lower(), inputCharacter)
x=uppercase
y=lowercase
print(x,"\n",y)
```

```
Enter a sequence of characters: HBDIGhvhtdfhd
<map object at 0x000001A8CB76D6C0>
 <map object at 0x000001A8CB76C700>
```

In [17]:
```
print(x)
```

```
<map object at 0x000001A8CB76D6C0>
```

In [18]:
```
listA=['one', 'two' , 'three']
listB=['apple','cherry','watermelon']
list(map(lambda x, y: x+ ' ' +y, listA, listB))
```

Out[18]:
```
['one apple', 'two cherry', 'three watermelon']
```

In [20]:
```
student_data = {
    'Cierra Vega': (6.2, 71),
    'Alden Cantrell': (5.9, 65),
    'Kierra Gentry': (6.0, 68),
    'Pierre Cox': (5.8, 66)
}

filtered_students = dict(filter(lambda item: item[1][0] > 6.0 and item[1][1] > 70, student_data.items()))
```

```
print("Original Dictionary:")
print(student_data)

print("Height > 6ft and Weight > 70kg:")
print(filtered_students)
```

```
Original Dictionary:
{'Cierra Vega': (6.2, 71), 'Alden Cantrell': (5.9, 65), 'Kierra Gentry': (6.0, 68), 'Pierre Cox': (5.8, 66)}
Height > 6ft and Weight > 70kg:
{'Cierra Vega': (6.2, 71)}
```

In [21]:
```
original_list = [12, 0, None, 23, None, -55, 234, 89, None, 0, 6, -12]
filtered_list = list(filter(lambda x: x is not None, original_list))
print(f"Remove None value from the said list: \n {filtered_list}")
```

```
Remove None value from the said list:
 [12, 0, 23, -55, 234, 89, 0, 6, -12]
```

In [1]:
```
from functools import reduce
list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
product = reduce(lambda x, y: x * y, list1)
print(f"Product of the said list numbers: {product}")
```

```
Product of the said list numbers: 3628800
```

In [2]:
```
from functools import reduce
original_list = [4, 3, 2, 2, -1, 18]
result = reduce(lambda x, y: x * y, original_list)
print("Multiply all the numbers of the said list:", result)
```

```
Multiply all the numbers of the said list: -864
```

In [3]:
```
mixed_list = [19, 'red', 12, 'green', 'blue', 10, 'white', 'green', 1]

sorted_list = sorted(mixed_list, key=lambda x: (isinstance(x, int), x))
print(f"Sort the said mixed list of integers and strings: {sorted_list}")
```

```
Sort the said mixed list of integers and strings: ['blue', 'green', 'green', 'red', 'white', 1, 10, 12, 19]
```

In [4]:
```
points = [(1, 2), (5, 3), (0, 7), (3, 1)]

sorted_points = sorted(points, key=lambda point: sum(point))
print(sorted_points)
```

```
[(1, 2), (3, 1), (0, 7), (5, 3)]
```

In [5]:
```
list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
list2 = [2, 4, 6, 8]
list1 = list(filter(lambda x: x not in list2, list1))
print("Updated list1:", list1)
```

```
Updated list1: [1, 3, 5, 7, 9, 10]
```

In [7]:
```
from collections import Counter
original_list = [3, 4, 5, 8, 0, 3, 8, 5, 0, 3, 1, 5, 2, 3, 4, 2]
counted_items = dict(Counter(original_list))
sorted_counted_items = dict(sorted(counted_items.items()))
print("Count the occurrences of the items in the said list:")
print(sorted_counted_items)
```

```
Count the occurrences of the items in the said list:
{0: 2, 1: 1, 2: 2, 3: 4, 4: 2, 5: 3, 8: 2}
```

In [ ]:
```
```