TASK 4

Exercise 1: Create a function with a default argument

Write a program to create a function `show_employee()` using the following conditions.

- It should accept the employee's name and salary and display both.
- If the salary is missing in the function call then assign default value 9000 to salary

Given:

```
showEmployee("Ben", 12000)
showEmployee("Jessa")
```

Expected output:

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

In [4]:
```
# Excercice 1
def showEmployee(a,b=9000):
        print("name: "+a,"salary: ",b)
showEmployee("Ben", 12000)
showEmployee("Jessa")
```

```
name: Ben salary:  12000
name: Jessa salary:  9000
```

Exercise 2: Create an inner function to calculate the addition in the following way

- Create an outer function that will accept two parameters, `a` and `b`
- Create an inner function inside an outer function that will calculate the addition of `a` and `b`
- At last, an outer function will add 5 into addition and return it

In [21]:
```
# Exercise 2
a=int(input("Enter a number: "))
b=int(input("Enter a number: "))
def add5(a,b):
    def addition(a,b):
        print(a+b)
        addition(a,b)
    print(a+b+5)
add5(a,b)
```

```
Enter a number: 8
Enter a number: 7
20
```

Exercise 3: Generate a Python list of all the even numbers between 4 to 30

In [23]:
```
# Exercise 3
Evennum=[i for i in range(4,31) if i%2==0]
print(Evennum)
```

```
[4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30]
```

Exercise 4: Lambda Function to Check if value is in a List

Given a list, the task is to write a Python program to check if the value exists in the list or not using the lambda function.

Input : L = [1, 2, 3, 4, 5]
element = 4
Output : Element is Present in the list

Input : L = [1, 2, 3, 4, 5]
element = 8
Output : Element is NOT Present in the list

In [24]:
```
# Exercise 4
l=[1,2,3,4,5]
num=int(input("Enter an element in the list: "))
x= lambda l,num:True if num in l else False
if(x(l,num)):
    print(num,"Present in the list")
else:
    print(num,"Not present in the list")
```

```
Enter an element in the list: 4
4 Present in the list
```

Exercise 5: Sort list of tuples with their sum

Sort the points based on their sum of elements in the tuples

points = [(1, 2), (5, 3), (0, 7), (3, 1)]

In [25]:
```
# Exercise 5
points = [(1, 2), (5, 3), (0, 7), (3, 1)]
sorted_points=sorted(points, key=lambda x:x[0]+x[1])
print(sorted_points)
```

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

In [ ]:

Exercise 6 :

Write a python function, which will find all such numbers between 1000 and 3000 (both included) such that each digit of the number is an even number. Return the results as a list

```
def even_numbers(start,end):
    even_digit_numbers=[]
    for number in  range(start,end+1):
        even_digits = all(int(digit)%2 == 0 for digit in str(number))
        if even_digits:
            even_digit_numbers.append(number)
    return even_digit_numbers
result=even_numbers(1000,3000)
print(result)
```

```
[2000, 2002, 2004, 2006, 2008, 2020, 2022, 2024, 2026, 2028, 2040, 2042, 2044,
2046, 2048, 2060, 2062, 2064, 2066, 2068, 2080, 2082, 2084, 2086, 2088, 2200, 2
202, 2204, 2206, 2208, 2220, 2222, 2224, 2226, 2228, 2240, 2242, 2244, 2246, 22
48, 2260, 2262, 2264, 2266, 2268, 2280, 2282, 2284, 2286, 2288, 2400, 2402, 240
4, 2406, 2408, 2420, 2422, 2424, 2426, 2428, 2440, 2442, 2444, 2446, 2448, 246
0, 2462, 2464, 2466, 2468, 2480, 2482, 2484, 2486, 2488, 2600, 2602, 2604, 260
6, 2608, 2620, 2622, 2624, 2626, 2628, 2640, 2642, 2644, 2646, 2648, 2660, 266
2, 2664, 2666, 2668, 2680, 2682, 2684, 2686, 2688, 2800, 2802, 2804, 2806, 280
8, 2820, 2822, 2824, 2826, 2828, 2840, 2842, 2844, 2846, 2848, 2860, 2862, 286
```

Exercise 7 :

Write a python function that accepts a sentence and calculate and return the number of letters and digits.
Suppose the following input is supplied to the program:
hello world! 123
Then, the output should be:
LETTERS 10
DIGITS 3

In [29]:
```
str=input("input a string: " )
d=l=0
for x in str:
    if x.isdigit():
        d+=1
    elif x.isalpha():
        l+=1
    else:
        pass
print("Letters",l)
print("Digits",d)
```

```
input a string: Hello World! 123
Letters 10
Digits 3
```

Exercise 8 MAP:

Write a Python program to convert all the characters into uppercase and lowercase and eliminate duplicate letters from a given sequence. Use the

map() function

```
In [46]: # Exercise 8
a=input("Enter the sentence: ")
result_1=map(lambda x:x.upper(),a)
result_2=map(lambda x:x.lower(),a)
result_3=set(a)

for i in result_3:
    print(i)
```

```
Enter the sentence: Banana
a
n
B
```

```
In [47]: for i in result_1:
    print(i)
```

```
B
A
N
A
N
A
```

```
In [48]: for i in result_2:
    print(i)
```

```
b
a
n
a
n
a
```

Exercise 9 MAP:

Write a Python program to add two given lists and find the difference between them. Use the map() function

```
In [49]: list_1=[1,2,3,4,5,6]
list_2=[1,5,8,7,4,9]
result_1=map(lambda x,y:x+y,list_1,list_2)
list(result_1)
```

```
Out[49]: [2, 7, 11, 11, 9, 15]
```

```
In [50]: result_2=map(lambda x,y:x-y,list_1,list_2)
list(result_2)
```

```
Out[50]: [0, -3, -5, -3, 1, -3]
```

Exercise 10 Filter:

Write a Python program to filter the height and weight of students, which are stored in a dictionary using lambda.
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 [51]:
```
dict_1={'Cierra Vega': (6.2, 71), 'Alden Cantrell': (5.9, 65), 'Kierra Gentry': (
dict_2=filter(lambda i:dict_1[i][0]>6 and dict_1[i][1]>70,dict_1)
for i in dict_2:
    print({i:dict_1[i]})
```

```
{'Cierra Vega': (6.2, 71)}
```

Exercise 11 Filter:

Write a Python program to remove all elements from a given list present in another list using lambda.
Original lists:
list1: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
list2: [2, 4, 6, 8]
Remove all elements from 'list1' present in 'list2:
[1, 3, 5, 7, 9, 10]

In [53]:
```
list1=[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
list2= [2, 4, 6, 8]
result=filter(lambda x:x not in list2,list1)
list(result)
```

Out[53]:
```
[1, 3, 5, 7, 9, 10]
```

Exercise 12 Reduce:

Write a Python program to calculate the product of a given list of numbers using lambda.
list1: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Product of the said list numbers:
3628800

In [54]:
```
# Exercise 12
from functools import reduce
list1= [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
result=reduce(lambda a,b:a*b,list1)
result
```

Out[54]:
```
3628800
```

Exercise 13 Reduce:

Write a Python program to multiply all the numbers in a given list using lambda.
Original list:
[4, 3, 2, 2, -1, 18]

Multiply all the numbers of the said list: -864

```
In [55]: from functools import reduce
         Original_list=[4, 3, 2, 2, -1, 18]
         result=reduce(lambda a,b:a*b, Original_list)
         result

Out[55]: -864
```

Exercise 14 Reduce:

Write a Python program to calculate the average value of the numbers in a given tuple of tuples using lambda.
Original Tuple:
((10, 10, 10), (30, 45, 56), (81, 80, 39), (1, 2, 3))
Average value of the numbers of the said tuple of tuples:
(30.5, 34.25, 27.0)

```
In [64]: # Exercise 14
         from functools import reduce
         tup=((10, 10, 10), (30, 45, 56), (81, 80, 39), (1, 2, 3))
         avg=map(lambda i:sum(i)/(len(i)),zip(*tup))
         print(tuple(avg))

         (30.5, 34.25, 27.0)
```

Exercise 15:

Write a Python program to sort a given mixed list of integers and strings using lambda. Numbers must be sorted before strings.
Original list:
[19, 'red', 12, 'green', 'blue', 10, 'white', 'green', 1]
Sort the said mixed list of integers and strings:
[1, 10, 12, 19, 'blue', 'green', 'green', 'red', 'white']

Exercise 16:

Write a Python program to count the occurrences of items in a given list using lambda.
Original list:
[3, 4, 5, 8, 0, 3, 8, 5, 0, 3, 1, 5, 2, 3, 4, 2]
Count the occurrences of the items in the said list:
{3: 4, 4: 2, 5: 3, 8: 2, 0: 2, 1: 1, 2: 2}

```
In [100]: list1=[3, 4, 5, 8, 0, 3, 8, 5, 0, 3, 1, 5, 2, 3, 4, 2]
          dic=dict(map(lambda x: (x,list1.count(x)),set(list1)))
          print(dic)

          {0: 2, 1: 1, 2: 2, 3: 4, 4: 2, 5: 3, 8: 2}
```

Exercise 17:

Write a Python program to remove None values from a given list using the lambda function.
Original list:
[12, 0, None, 23, None, -55, 234, 89, None, 0, 6, -12]
Remove None value from the said list:
[12, 0, 23, -55, 234, 89, 0, 6, -12]

```
In [2]: l1=[12, 0, None, 23, None, -55, 234, 89, None, 0, 6, -12]
        no_none=list(filter(lambda x:x is not None,l1))
        print(no_none)

        [12, 0, 23, -55, 234, 89, 0, 6, -12]
```