```python
# 1 write a python program to access a function inside a function.

def greater(a,b):
    if a > b:
        return a
    return b


def greatest(a,b,c):
    if a > b and a > c:
        return a
    elif b >a and b > c:
        return c
    else:
        return c


#function inside function

def new_greatest(a,b,c):
    bigger = greater(a,b)
    return greater(bigger, c)

print(new_greatest(100,200,300))
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\1 func inside func.py"
300
PS C:\Users\popso\Desktop\javascript>
```

```python
1   # 2. a python function to create and print a list where the values are square of numbers between 1 and 30
2
3   def values():
4       l = list()
5
6       for i in range (1,30):
7           l.append(i**2)
8
9       print(l)
10
11  values()
12
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   **TERMINAL**   ▶ Code + ∨

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\2 square of numbers.py"
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 256, 289, 324, 361, 400, 441, 484, 529, 576, 625, 676, 729, 784, 841]
PS C:\Users\popso\Desktop\javascript> ▊
```

```python
# 3. assign a different name to function and call it through the new name.

def student1(name, age):
    print(name,age)

student1('john cena', 25)

#assign new name

student2 = student1
#calling using a new name
student2('rock', 25)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\3 new name func.py"
john cena 25
rock 25
PS C:\Users\popso\Desktop\javascript>
```

```python
# 4. a python list of all the even numbers between 4 to 30.

def even():
    l = []
    a = range(4,30)
    for i in a:
        if i % 2 == 0:
            print(i)
            l.append(i)
print(even())
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   **TERMINAL**   ≥ Code ＋∨ ⬚ 🗑

```
4
6
8
10
12
14
16
18
20
22
24
26
28
None
PS C:\Users\popso\Desktop\javascript>
```

num 5 num as arg and return sum.py ×

assignments func part2 > num 5 num as arg and return sum.py > ...

```python
#5 . func that accepts two num as arguments and return the sum.

def add (x,y):
    result = x + y
    print (x, "+", y, "=", result)

n1 = int(input("enter the first num: "))
n2 = int (input("enter the second num: "))

add(n1,n2)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\num  5 num as arg and return sum.py"
enter the first num: 10
enter the second num: 10
10 + 10 = 20
PS C:\Users\popso\Desktop\javascript>
```

```python
# 6. python function that accepts different values as parameters and returns a list.

def multiple():
    operation = "sum"
    total = 4+6
    operation2 = "substract"
    after_sub = 6-6
    return [operation, total,operation2,after_sub]

values = multiple()

print(values)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\6 values as para.py"
['sum', 10, 'substract', 0]
PS C:\Users\popso\Desktop\javascript>
```

```python
# 7. python function that returns a tuple.

def person():
    return "popson", 25, "wayanad" #you can return a tuple by separating each item with a comma

 #"It is actually the comma which makes a tuple, not the parentheses,"


print(person())
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\7 func return tuple.py"
('popson', 25, 'wayanad')
PS C:\Users\popso\Desktop\javascript>
```

```python
#8 define a function that accepts 2 values and return its sum , substraction and multiplication.

def products(a,b):
    return a+b, a-b, a*b

x =int( input('enter the first num: '))
y =int( input('enter the second num: '))

sum, subt,multiply = products(x,y)
print("sum of two num is ", sum)
print("subt result of two num ", subt)
print("after multiplication ", multiply)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   **TERMINAL**   ⟩ Code + ∨ ▯ ▥ ∧ ✕

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\8  .2value sum sub mult.py"

enter the first num: 8
enter the second num: 10
sum of two num is  18
subt result of two num  -2
after multiplication  80
PS C:\Users\popso\Desktop\javascript> 
```

```python
# 9. define a function that accepts roll number and returns whether the student is present or not.

def student(roll):
    x = [1,3,5,7,9,22] #Use a list to store sample roll num
    if roll in x:
        print(f"roll num {roll} is present")
    else:
        print(f"roll num {roll} is absent")
roll = int(input("enter roll no: "))
student(roll)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\9.roll number.py"
enter roll no: 12
roll num 12 is absent
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\9.roll number.py"
enter roll no: 5
roll num 5 is present
PS C:\Users\popso\Desktop\javascript>
```

```python
# 10. define a func which counts vowels and consonant in a word.


def count(val):
    vov = 0
    con = 0
    for i in range(len(val)):
        if val[i] in ['a','e','i','o','u']:
            vov = vov+1
        else:
            con = con+1


    print('count of vovels is ',vov)
    print('count of consonant is', con)


x = input("enter the value: ")
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   ▶ Code + ∨ ⬚ 🗑 ∧ ✕

PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\10.func vowels and consonan
t.py"
enter the value: popson
count of vovels is 2
count of consonant is 4
PS C:\Users\popso\Desktop\javascript> ▎

```python
#11 . define a function that accepts a number and returns whether the number is even or odd.

def even_odd(num):
    if num % 2 == 0:
        print("entered num is even")
    else:
        print("entered num is odd")
x = int(input("enter your num "))
even_odd(x)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\11.return even or odd .py"
enter your num 10
entered num is even
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\11.return even or odd .py"
enter your num 7
entered num is odd
PS C:\Users\popso\Desktop\javascript>
```

🐍 12. accepts 3 values and return .py ✕

▷ ∨ ▥ ⋯

assignments func part2 > 🐍 12. accepts 3 values and return .py > ...

```python
1    # 12. define a function in python that accepts 3 values and returns the maximum of three numbers.
2
3    def max(a,b,c):
4        if a>b and a>c:
5            print(f"{a} is max num among 3")
6        elif b>a and b>c:
7            print(f"{b} is max among 3")
8        else:
9            print(f"{c} is max among 3")
10
11   max(10,20,30)
12
13
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   **TERMINAL**

⟩ Code ＋∨ ▥ 🗑 ∧ ✕

```
PS C:\Users\popso\Desktop\javascript> python -u "c:\Users\popso\Desktop\javascript\assignments func part2\12. accepts 3 values and re
turn .py"
30 is max among 3
PS C:\Users\popso\Desktop\javascript>
```