

# functions

```
In [1]: 1 # definition of function
        2
        3 def sayhi():
        4     print("hiiii hello everyone welcome to my session ")
```

```
In [3]: 1 # calling a function
        2 sayhi()

hiiii hello everyone welcome to my session
```

```
In [13]: 1 # generalised function with user input
         2 def addition(n1,n2):
         3     a = n1+n2
         4     return a
```

```
In [14]: 1 # calling a function
         2
         3 result = addition(20,250)
         4 print(result)

270
```

```
In [16]: 1 # function parameter order
         2
         3 def areaofcircle(radius,pi):
         4     return (pi*radius**2)
```

```
In [17]: 1 # calling a function using proper order
         2 areaofcircle(10,3.14)
```

Out[17]: 314.0

```
In [18]: 1 # calling a function using wrong order
         2 areaofcircle(3.14,10)
```

Out[18]: 98.596

```
In [19]: 1 # using default arguments
         2
         3 def areacircle(radius,pi = 3.14):
         4     return (pi*radius**2)
```

```
In [20]: 1 # calling
         2 areacircle(10)
```

Out[20]: 314.0

```
In [32]: 1 # parameters vs arguments :  
2 # parameters are used while defining function  
3 # arguments are used while calling function
```

```
In [30]: 1 # odd_even  
2  
3 def iseven(num):  
4     if num%2==0:  
5         return True  
6     else:  
7         return False
```

```
In [31]: 1 iseven(10)
```

Out[31]: True

```
In [ ]: 1 # num : parameter  
2 # 10 : argument
```

```
In [33]: 1 # docstring : add comments / documentations to your function  
2 # it is written in triple quotes  
3  
4 def iseven(num):  
5     """Return true if number is even else return false."""  
6     if num%2==0:  
7         return True  
8     else:  
9         return False
```

```
In [34]: 1 # calling doc string from defined function  
2  
3 iseven.__doc__
```

Out[34]: 'Return true if number is even else return false.'

```
In [ ]: 1
```

```
In [35]: 1 # variable lenght of arguments
```

```
In [36]: 1 def add(*args):  
2     return sum(args)
```

```
In [38]: 1 add(1,3,5,10)
```

Out[38]: 19

```
In [50]: 1 def getmax(*args):  
2     return max(args)
```

```
In [52]: 1 getmax(1,7,89,45)
```

```
Out[52]: 89
```

```
In [53]: 1 # global Vs local variable
```

```
In [54]: 1 var = 1 # global varibale
2
3 def ad(*nums):
4     # total = local variable
5     total = 0
6     for i in nums:
7         total = total + i
8     return total
```

```
In [58]: 1 ad(1,2,3,4,5,6)
```

```
Out[58]: 21
```

```
In [56]: 1 print(var) # global
```

```
1
```

```
In [57]: 1 print(total) # local
```

```
-----
-----
NameError                                Traceback (most recent c
all last)
<ipython-input-57-517c68ee957b> in <module>
----> 1 print(total)

NameError: name 'total' is not defined
```

```
In [ ]: 1
```

## lambda

```
In [59]: 1 # find cube of number using normal def(function)
2
3 def cube(n):
4     return (n*n*n)
```

```
In [60]: 1 cube(2)
```

```
Out[60]: 8
```

In [65]:

```
1 # find cube of number using lamda
2 g = lambda x:x*x*x
```

In [68]:

```
1 g(3)
```

Out[68]: 27

In [70]:

```
1 # square rot of number using lambda
2 (lambda a:a**0.5)(9)
```

Out[70]: 3.0

In [ ]:

```
1
```

In [71]:

```
1 def agecal(yob):
2     return (2022-yob)
```

In [72]:

```
1 agecal(1990)
```

Out[72]: 32

In [ ]:

```
1
```

In [ ]:

```
1
```

In [ ]:

```
1
```

In [ ]:

```
1
```