Let's create a function(with docstring)

```
def is_even(num):
  """
  This function returns if a given number is odd or even
  input - any valid integer
  output - odd/even
  created on - 16th Nov 2022
  """
  if type(num) == int:
    if num % 2 == 0:
      return 'even'
    else:
      return 'odd'
  else:
    return 'pagal hai kya?'
```

```
# function
# function_name(input)
for i in range(1,11):
  x = is_even(i)
  print(x)
```

```
odd
even
odd
even
odd
even
odd
even
odd
even
```

```
print(type.__doc__)
```

```
type(object_or_name, bases, dict)
type(object) -> the object's type
type(name, bases, dict) -> a new type
```

2 Point of views

```
is_even('hello')
```

```
'pagal hai kya?'
```

Parameters Vs Arguments

Types of Arguments

- Default Argument
- Positional Argument
- Keyword Argument

```
def power(a=1,b=1):
  return a**b
```

```
power()
```

```
    1
```

```
# positional argument
power(2,3)
```

```
    8
```

```
# keyword argument
power(b=3,a=2)
```

```
    8
```

`*args and **kwargs`

`*args and **kwargs` are special Python keywords that are used to pass the variable length of arguments to a function

```
# *args
# allows us to pass a variable number of non-keyword arguments to a function.

def multiply(*kwargs):
  product = 1

  for i in kwargs:
    product = product * i

  print(kwargs)
  return product
```

```
multiply(1,2,3,4,5,6,7,8,9,10,12)
```

```
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12)
43545600
```

```
# **kwargs
# **kwargs allows us to pass any number of keyword arguments.
# Keyword arguments mean that they contain a key-value pair, like a Python dictionary.

def display(**salman):

  for (key,value) in salman.items():
    print(key,'->',value)



display(india='delhi',srilanka='colombo',nepal='kathmandu',pakistan='islamabad')
```

```
india -> delhi
srilanka -> colombo
nepal -> kathmandu
pakistan -> islamabad
```

Points to remember while using `*args and **kwargs`

- order of the arguments matter(normal -> `*args` -> `**kwargs`)
- The words “args” and “kwargs” are only a convention, you can use any name of your choice

How Functions are executed in memory?

Without return statement

```
L = [1,2,3]
print(L.append(4))
print(L)
```

```
None
[1, 2, 3, 4]
```

Variable Scope

```
def g(y):
    print(x)
    print(x+1)
x = 5
g(x)
print(x)
```

```
def f(y):
    x = 1
    x += 1
    print(x)
x = 5
f(x)
print(x)
```

```
def h(y):
    x += 1
x = 5
h(x)
print(x)
```

```
def f(x):
   x = x + 1
   print('in f(x): x =', x)
   return x

x = 3
z = f(x)
print('in main program scope: z =', z)
print('in main program scope: x =', x)
```

Nested Functions

```
def f():
  def g():
    print('inside function g')
    f()
  g()
  print('inside function f')
```

```
f()
```

```
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
inside function g
```

```
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
    inside function g
```