```python
def foo(name, n):

    name = 'Edie'
    n = n + 2
    return name, n

name = 'Jack'
n = 6
print(foo(name, n))
print(name, n)

"""it's printing the current value, function will not
interfere with these constant"""

#
import functools
list = [1,2,3,4]
func = lambda x,y:x*y
print(functools.reduce(func, list))

"""Reduce takes two first elements of a sequence, apply
function to them and returns
a single value. Next step the returned value becomes the
first and the third element
is the second item.  It continues until it returns a single
value as a final result."""

#
def foo(a,b):
    if a+b in range(10,15):
        return 20
    else:
        return a+b
x = foo(9,5)
y = foo(9,6)
print(foo(x,y))
"""Here foo(9,5) = 14 that means it return 20 because it's
in range. And y = foo(9,6)=15
that means it's not in range. So x = 20 and y = 15 and the
```

```python
result is = 35"""

#
def foo(a, b):
    if a == b:
        return (a+b)*2
    else:
        return a+b
x = foo(1,3)
y = foo(2,2)
print(x,y)

#
"""userInput = input("Who is your Girl ? ")
while userInput != "Mou":
    userInput = input("No! It's not. ")
    if userInput == "Mou":
        print("Congo! You find her.")
        break
        make it reverse"""

#
id = [3,100,5,0,4]
print(id[False])
print(id[True])

a = (True == 1)
print(a)
b = (False == 0)
print(b)

"""Because True is treated as 1 in python so var[True] =
var[1], I mean index 1 as
Python counts starting from 0."""

#
(s,t) = (259,259)
print(type((259,259)))
```

```python
for i in range(1,11):
    print([f"{i} odd", f"{i} even"][ i % 2 == 0])

""" f is to format the string and make the variables appear
in the string."""

def foo(var = 5):
    #Default argument
    var += 1.1
    return var
print(foo(), foo(1))
#X,Y
"""6.1 -- as no argument is passed so takes default
argument.
2.1 -- argument is passed, so will give according to that
value. print X and Y are
like placeholders  to the actual answer.just fill it up with
you answer. You can use
keyword args in the same way you use positional args."""
"""Correct answer is nothing. You did not initialized X or
Y. But you could wrote
X, Y = (foo(), foo(1)). Then you end up with 6.1 and 2.1"""

#
#Print(i if i%2==0 else i%2!=0
"""for i in range(1,11):
#    print(i if i%2 == 0 else i%2 != 0)
    if i%2 == 0:
        print(True)
    elif i%2 != 0:
        print(False)
"""
#inheritance python OOPS concept
class Base():
    def hi(self):
        return 5
    #extending
class Child(Base):
    def hello(self):
```

```python
        return 6
c = Child()
#access child class object
print(c.hello())
#You can access parent class objects because of inheritance
print(c.hi())

name = 'Jhon'
print(name.center(25))
print(name.center(25, '*'))
print(name.rjust(25, '-'))
print(name.ljust(30))
print(name.ljust(25))

#
n = 5
for level in range(1, n+1):
    print(level * '*')
for level in range(1, n+1):
    print(' ' * (n-level) + level * '*')

#
character = 'Iron Man'
movie = 'Avengers: Endgame'
print('Will ' + character + ' alive in ' + movie + ' ?')
print('Will %s alive in %s ?'%(character, movie))
print('Will {} alive in {} ?'.format(character, movie))
print('Will {0} alive in {1} ?'.format(character, movie))
print(f'Will {character} alive in {movie} ?')

#
a = [4,4,3,5,7,]
print(type(a))
print(set(a))
print(a)

#
def foo(func_name):
    a = [1,5,3,7,4]
```

```python
        return func_name(a)
print(foo(sum))

"""import os
fileType = '.mp4'
anyFile = os.listdir('.')
for file in anyFile:
    if file.endswith(fileType):
        print(file)
"""
def foo(n):
    if n == 1:
        return None
print(foo(1))

def foo(n):
    if n == 1:
        return
print(foo(1))

#
def foo(func_name):
    a = [1,5,3,7,4]
    return func_name(a)
print(foo(sum))
print(foo(len))
print(foo(min))

import dis
def func(a,b):
    return a+b
print(dis.dis(func))

a = [1,2,3,4]
for item in a:
    if item == 2:
        a.pop(2)
        a.remove(2)
print(a)
```

```python
#
"""n = int(input("Calculate salary for months: "))
def recursion(n):
    if n == 1:
        return 500
    else:
        return recursion(n-1)+500
print(recursion(n))"""

import re
text = "Dummy python text"
print(re.sub("m", "-", text))
print(re.sub("m|t", "-", text))
print(re.sub("m|t|o", "-", text))
"""for multiple string replacement at once"""
#
import re
text = "python programming language is for Devs"
print(re.findall("p\w+",text))
text = "shakil loves python but shoshi loveshakil. "
print(re.findall("s\w+",text))
message = "hei, hello, how was that"
print(re.findall("h\w",message))

#
my_items = ['a', 'b', 'c']
i = 0
while i<len(my_items):
    print(my_items[i])
    i += 1
print(range(len(my_items)))
for i in range(len(my_items)):
    print(my_items[i])
for item in my_items:
    print(item)
for i,item in enumerate(my_items):
    print(f'{i}: {item}')
```

```python
#
emails = {
    'Bob': 'bob@example.com',
    'ALice': 'alice@example.com',
}
for name, email in emails.items():
    print(f'{name} - {email}')
```