In [1]:
```
print("Hello World")
```

```
Hello World
```

In [2]:
```
numbers = [111, 7, 2, 1]
print(len(numbers))
print(numbers)
```

```
4
[111, 7, 2, 1]
```

In [3]:
```
numbers.append(4)
```

In [4]:
```
print(len(numbers))
```

```
5
```

In [5]:
```
print(numbers)
```

```
[111, 7, 2, 1, 4]
```

In [6]:
```
numbers.append(10)
```

In [7]:
```
print(numbers)
```

```
[111, 7, 2, 1, 4, 10]
```

In [8]:
```
numbers.insert(0, 222)
```

In [9]:
```
print(numbers)
```

```
[222, 111, 7, 2, 1, 4, 10]
```

In [10]:
```
numbers.insert(3, 5)
```

In [11]:
```
print(numbers)
```

```
[222, 111, 7, 5, 2, 1, 4, 10]
```

In [12]:
```
my_list = []  # Creating an empty list.

for i in range(5):
    my_list.append(i + 1)

print(my_list)
```

```
[1, 2, 3, 4, 5]
```

In [13]:
```
my_list = []

for i in range(5):
    my_list.insert(0, i + 1)
```

```
print(my_list)
```

```
[5, 4, 3, 2, 1]
```

In [14]:
```
my_list = []

for i in range(5):
    my_list.insert(0, i + 1)
    print(my_list)
```

```
[1]
[2, 1]
[3, 2, 1]
[4, 3, 2, 1]
[5, 4, 3, 2, 1]
```

In [15]:
```
lst = [1, 2, 3, 4, 5]
lst_2 = []
add = 0

for number in lst:
    add += number # add = add + number -> iterasi pertama, number = 1; add = add + 1
    lst_2.append(add) # [1]

print(lst_2)
```

```
[1, 3, 6, 10, 15]
```

In [16]:
```
lst = [5, 3, 1, 2, 4]
print(lst)
```

```
[5, 3, 1, 2, 4]
```

In [17]:
```
lst.sort()
```

In [18]:
```
print(lst)
```

```
[1, 2, 3, 4, 5]
```

In [19]:
```
list_1 = [1]
list_2 = list_1
```

In [20]:
```
print(list_1)
```

```
[1]
```

In [21]:
```
print(list_2)
```

```
[1]
```

In [22]:
```
list_1[0] = 2
```

In [23]:
```
print(list_1)
```

```
[2]
```

In [24]:
```
print(list_2)
```

```
[2]
```

In [25]:
```
list_1 = ["A"]
list_2 = list_1[:]
```

In [26]:
```
print(list_1)
```

```
['A']
```

In [27]:
```
print(list_2)
```

```
['A']
```

In [28]:
```
list_1[0] = "a"
```

In [29]:
```
print(list_1)
```

```
['a']
```

In [30]:
```
print(list_2)
```

```
['A']
```

In [31]:
```
my_list = [10, 8, 6, 4, 2]
new_list = my_list[1:3] # Memotong my_list dari index ke-1 sampai index ke-2 -> 2 it
print(new_list)
```

```
[8, 6]
```

In [32]:
```
my_list[1]
```

Out[32]:
```
8
```

In [34]:
```
my_list[0:4]
```

Out[34]:
```
[10, 8, 6, 4]
```

In [35]:
```
my_list = [10, 8, 6, 4, 2]
del my_list[:]
```

In [36]:
```
print(my_list)
```

```
[]
```

In [37]:
```
my_list = [10, 8, 6, 4, 2]
del my_list
```

In [38]:
```
print(my_list)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/2684720215.py in <module>
----> 1 print(my_list)

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

In [39]:
```
my_list = [0, 3, 12, 8, 2]
```

In [40]:
```
print(5 in my_list) # Apakah 5 ada di my_list?
```

```
False
```

In [41]:
```
print(12 in my_list)
```

```
True
```

In [42]:
```
print(4 not in my_list)
```

```
True
```

In [43]:
```
list_1 = ["A", "B", "C"]
list_2 = list_1 # list_2 bukan copy-an dari list_1, tetapi mengacu ke objek/kotak ya
list_3 = list_2 # list_3 bukan copy-an dari list_2, tetapi mengacu ke objek/kotak ya

del list_1[0] # Menghapus "A" -> berdampak ke list_1, list_2, dan list_3 => Setelah
del list_2[0] # Menghapus "B" -> berdampak ke list_1, list_2, dan list_3 => Setelah

print(list_3)
```

```
['C']
```

In [44]:
```
list_1 = ["A", "B", "C"]
list_2 = list_1
list_3 = list_2

del list_1[0]
del list_2

print(list_3)
```

```
['B', 'C']
```

In [45]:
```
print(list_2)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/152081140.py in <module>
----> 1 print(list_2)

NameError: name 'list_2' is not defined
```

In [46]:
```
print(list_1)
```

```
['B', 'C']
```

In [47]:
```
del list_1
```

In [48]:
```
print(list_3)
```

```
['B', 'C']
```

In [49]:
```
del list_3
```

In [50]:
```
print(list_3)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/3604676404.py in <module>
----> 1 print(list_3)

NameError: name 'list_3' is not defined
```

In [52]:
```
list_1 = ["A", "B", "C"]
list_2 = list_1
```

In [53]:
```
var_1 = "A"
var_2 = var_1
```

In [54]:
```
print(var_1)
```

```
A
```

In [55]:
```
print(var_2)
```

```
A
```

In [56]:
```
var_1 = "B"
```

In [57]:
```
print(var_2)
```

```
A
```

In [58]:
```
var_1 = "A"
```

In [59]:
```
var_2 = var_1
var_3 = var_2
```

In [60]:
```
tabel = ["Nama", "Kelas", "Nilai"]
```

In [62]:
```
del tabel[:]
```

In [63]:
```
tabel
```

Out[63]:
```
[]
```

In [64]:
```
tabel = ["Nama", "Kelas", "Nilai"]
tabel_backup = tabel[:]
```

In [65]:
```
del tabel[:]
```

In [66]:
```
tabel_backup
```

Out[66]:
```
['Nama', 'Kelas', 'Nilai']
```

In [67]:
```
my_list = [1, 2, "in", True, "ABC"]

print(1 in my_list)  # outputs True
print("A" not in my_list)  # outputs True
print(3 not in my_list)  # outputs True
print(False in my_list)  # outputs False
```

```
True
True
True
False
```

In [68]:
```
def message():
    print("Enter a value: ")
```

In [69]:
```
message()
```

```
Enter a value:
```

In [70]:
```
def contoh_function():
    print(15 * 2)
```

In [71]:
```
contoh_function()
```

```
30
```

In [73]:
```
contoh_function()
```

```
30
```

In [74]:
```
def message():
    print("Enter a value: ")

print("We start here.")
message()
print("We end here.")
```

```
We start here.
Enter a value:
We end here.
```

In [75]:
```
def message(number):
```

```
    print("Enter a number:", number)
```

In [76]:
```
message(5)
```

```
Enter a number: 5
```

In [77]:
```
message(10)
```

```
Enter a number: 10
```

In [78]:
```
def contoh_function(x):
    print(15 * x)
```

In [79]:
```
contoh_function(1)
```

```
15
```

In [80]:
```
contoh_function(3)
```

```
45
```

In [81]:
```
contoh_function(10)
```

```
150
```

In [82]:
```
contoh_function(15)
```

```
225
```

In [86]:
```
def contoh_function(x, y):
    print(x * y + 2 * x - y)
```

In [84]:
```
contoh_function(3, 1)
```

```
8
```

In [ ]:
```

```

In [85]:
```
contoh_function(2, 5)
```

```
9
```

In [87]:
```
def introduction(first_name, last_name):
    print("Hello, my name is", first_name, last_name)

introduction("Luke", "Skywalker")
```

```
Hello, my name is Luke Skywalker
```

In [88]:
```
introduction("Skywalker", "Luke")
```

```
Hello, my name is Skywalker Luke
```

In [89]:
```
def contoh_function(x, y):
    print(x / y)
```

In [90]:
```
contoh_function(5, 2)
```

```
2.5
```

In [91]:
```
contoh_function(2, 5)
```

```
0.4
```

In [92]:
```
contoh_function(x=5, y=2)
```

```
2.5
```

In [93]:
```
contoh_function(y=2, x=5)
```

```
2.5
```

In [94]:
```
contoh_function(y=2, 5)
```

```
  File "C:\Users\ASUS\AppData\Local\Temp/ipykernel_12372/2557490819.py", line 1
    contoh_function(y=2, 5)
                         ^
SyntaxError: positional argument follows keyword argument
```

In [95]:
```
contoh_function(2, y=5)
```

```
0.4
```

In [96]:
```
def contoh_function(x, y, z):
    print(x + y - z)
```

In [97]:
```
contoh_function(3, z=2, y=1)
```

```
2
```

In [99]:
```
contoh_function(3, 2, z=1)
```

```
4
```

In [100…
```
def introduction(first_name, last_name="Smith"):
    print("Hello, my name is", first_name, last_name)
```

In [101…
```
introduction(first_name="Carlo", last_name="Abimanyu")
```

```
Hello, my name is Carlo Abimanyu
```

In [102…
```
introduction(first_name="Carlo")
```

```
Hello, my name is Carlo Smith
```

In [103…
```
introduction(last_name="Abimanyu")
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/388368048.py in <module>
----> 1 introduction(last_name="Abimanyu")

TypeError: introduction() missing 1 required positional argument: 'first_name'
```

In [104…
```
def contoh_function(x=1, y=2):
    print(x + y)
```

In [105…
```
contoh_function()
```

```
3
```

In [106…
```
contoh_function(x=5)
```

```
7
```

In [107…
```
contoh_function(y=10)
```

```
11
```

In [109…
```
contoh_function(x=5, y=10)
```

```
15
```

In [ ]:
```
1 + 1 + 2 + 3 + 5
```

In [110…
```
def perkalian(x, y):
    print(x * y)
```

In [111…
```
perkalian(5, 2)
```

```
10
```

In [112…
```
perkalian(4, 2)
```

```
8
```

In [113…
```
print(10 + perkalian(4, 2))
```

```
8
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/254533457.py in <module>
----> 1 print(10 + perkalian(4, 2))

TypeError: unsupported operand type(s) for +: 'int' and 'NoneType'
```

In [114…
```
type(perkalian(4, 2))
```

```
8
```

Out[114…
```
NoneType
```

In [115…
```
def perkalian(x, y):
    print(x * y)
```

In [116…
```
perkalian(3, 2)
```

```
6
```

In [117…
```
type(perkalian(3, 2))
```

```
6
```

Out[117…
```
NoneType
```

In [118…
```
def perkalian1(x, y):
    return x * y
```

In [119…
```
perkalian1(3, 2)
```

Out[119…
```
6
```

In [120…
```
type(perkalian1(3, 2))
```

Out[120…
```
int
```

In [121…
```
print(10 + perkalian1(4, 2))
```

```
18
```

In [126…
```
def meter_to_cm(length_in_meter):
    return length_in_meter * 100

def cm_to_meter(length_in_cm):
    return length_in_cm / 100
```

In [127…
```
meter_to_cm(10)
```

Out[127…
```
1000
```

In [128…
```
cm_to_meter(1000)
```

Out[128…
```
10.0
```

In [133…
```
def is_int(data):
    if type(data) == int:
        return True
    elif type(data) == float:
        return False
```

In [130…
```
print(is_int(5)) # True
```

```
True
```

In [131…
```
print(is_int(5.0)) # False
```

```
False
```

In [132…
```
print(is_int("5"))
```

```
None
```

In [135…
```
def is_int(data):
    if type(data) == int:
        return True
    elif type(data) == float:
        return False
    else:
        print(False)
```

In [136…
```
print(False == is_int("5"))
```

```
False
False
```

In [138…
```
type(is_int("5"))
```

```
False
```

Out[138…
```
NoneType
```

In [139…
```
def is_int(data):
    if type(data) == int:
        return True
    elif type(data) == float:
        return False
    else:
        print(5)
```

In [140…
```
print(1 + is_int("5")) # 1 + None
```

```
5
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/906725988.py in <module>
----> 1 print(1 + is_int("5"))

TypeError: unsupported operand type(s) for +: 'int' and 'NoneType'
```

In [141…
```
def is_int(data):
    if type(data) == int:
        return True
    elif type(data) == float:
        return False
    else:
        return 5
```

In [142…
```
print(1 + is_int("5"))
```

```
6
```

In [143…
```python
def is_int(data):
    if type(data) == int:
        return True
    elif type(data) == float:
        return False
    else:
        return "5"
```

In [144…
```python
print("1" + is_int("5"))
```

```
15
```

In [145…
```python
print(1 + is_int("5"))
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/906725988.py in <module>
----> 1 print(1 + is_int("5"))

TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

In [146…
```python
def my_function(x, y=5):
    if x % 2 == 0:
        return x * y
    else:
        print(x * y)
```

In [147…
```python
my_function(10)
```

Out[147…
```
50
```

In [148…
```python
type(my_function(10))
```

Out[148…
```
int
```

In [149…
```python
my_function(y=2)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_12372/4143578859.py in <module>
----> 1 my_function(y=2)

TypeError: my_function() missing 1 required positional argument: 'x'
```

In [150…
```python
my_function(y=3, x=2)
```

Out[150…
```
6
```

In [151…
```python
my_function(2, 3)
```

Out[151…
```
6
```

In [152…
```
def my_function(x, y=5):
    if x % 2 == 0:
        return x * y
    else:
        print(x * y)
```

In [153…
```
my_function(y=3, 2)
```

```
  File "C:\Users\ASUS\AppData\Local\Temp/ipykernel_12372/33905871.py", line 1
    my_function(y=3, 2)
                     ^
SyntaxError: positional argument follows keyword argument
```

In [154…
```
my_function(x=3, y=2)
```

```
6
```

In [155…
```
type(my_function(x=3, y=2))
```

```
6
NoneType
```

Out[155…

In [156…
```
def my_function(x, y=5):
    if x % 2 == 0:
        return x * y
    else:
        print(x * y)
```

In [157…
```
my_function(3, 2) == my_function(2, 3)
```

```
6
False
```

Out[157…

In [ ]:
```
my_function(3, 2)
```

In [ ]:
```
A == B # Apakah A sama dengan B?
```

In [158…
```
def my_function(x, y=5):
    if x == 2:
        return x

    if x % 2 == 0:
        return x * y
```

In [159…
```
my_function(x=2, y=10)
```

```
2
```

Out[159…

In [160…
```
def my_function(x, y=5):
    return x
```

```
    print(y) # Tidak dijalankan
    return y # Tidak dijalankan
```

In [161…

```
my_function(x=2, y=10)
```

Out[161…

```
2
```

In [168…

```
def my_function(x, y=5):
    if x == 2:
        return x
    print(x)
    return y
```

In [163…

```
my_function(x=3, y=10)
```

Out[163…

```
3
10
```

In [169…

```
print(1 + my_function(x=3, y=10))
```

```
3
11
```

In [ ]:

```
def label_encoding(x):
    if x == ...:
        return sesuatu
    elif x == ...:
        return sesuatu
```