zenius | Kampus Merdeka INDONESIA JAYA

Python II: Loop, Conditional Statement & Function

Senin, 20 Maret 2023

Arif Romadhan
Sr. Data Scientist

Program Zenius Studi Independen
Bersertifikat Bersama Kampus Merdeka

PT Cakra Syntesis Indonesia
Android Developer (2016)

Nawatech
Jr. ML Engineer (2017)

Bukalapak
Data Scientist (2018 - 2020)

Data Scientist Instructor
(Sept 2020 - Nov2020)

Data Scientist Instructor
(2021 - 2021)

Evermos
Sr. Data Scientist - Data Lead
(2021 - present)

Data Scientist Instructor
(2021- present)

Arif Romadhan

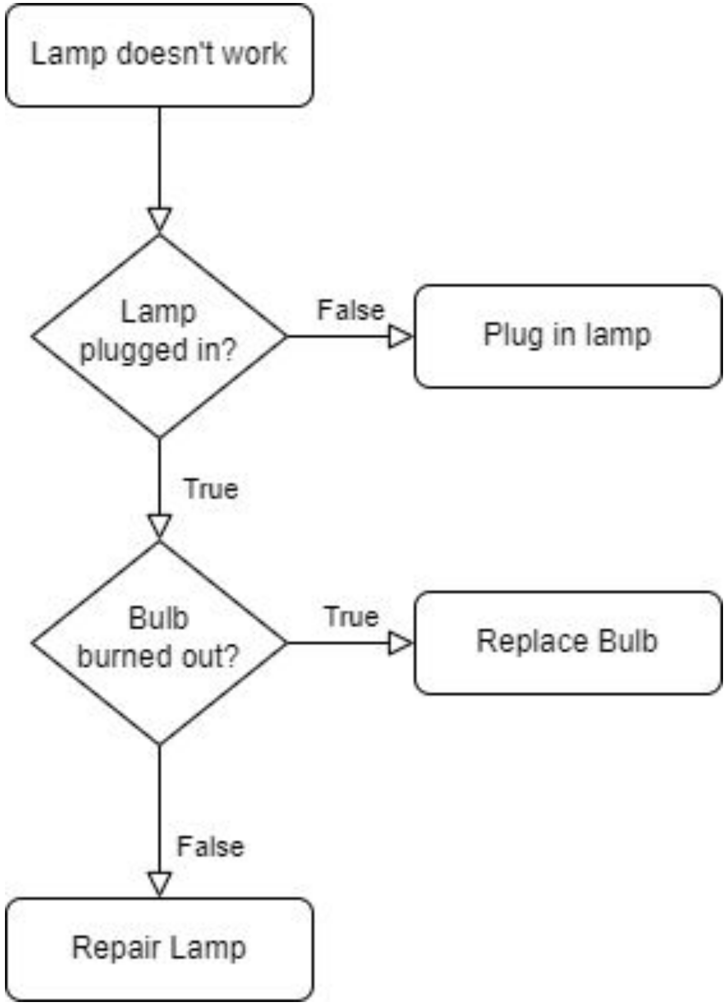
https://www.linkedin.com/in/arif-romadhan19/

1. Conditional Statement
2. Loop Statement
3. Functions
4. Error Handling

Conditional Statement in Python

Conditional Statement

Logical statement to execute python code based on conditions.

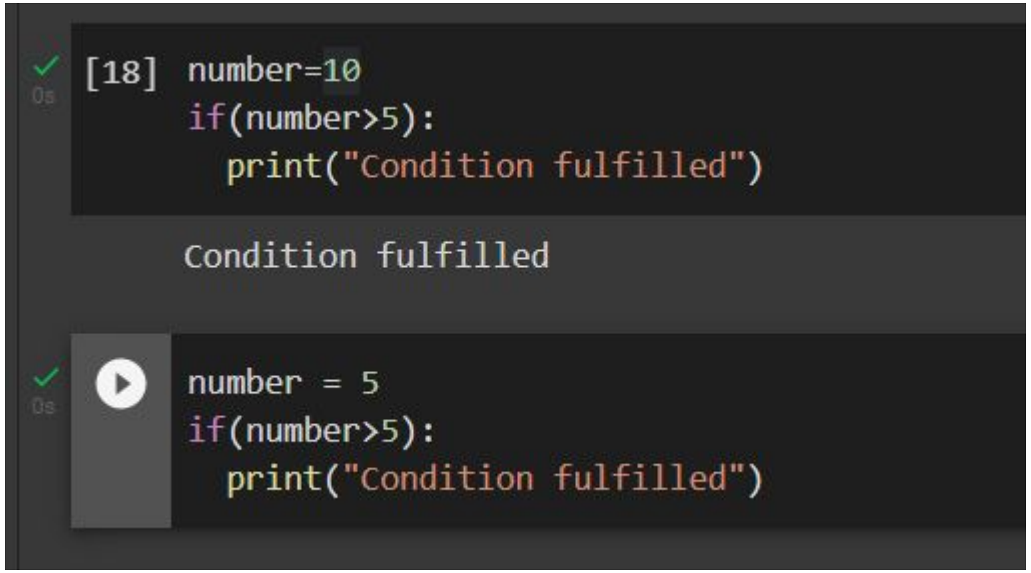


Live Coding

Let’s get your hand dirty

Conditional Statement

```
if condition:
  statement
elif condition:
  statement
elif condition:
  statement
else:
  statement
```



```
number=10
if(number>5):
  print("Condition fulfilled")
```

```
Condition fulfilled
```

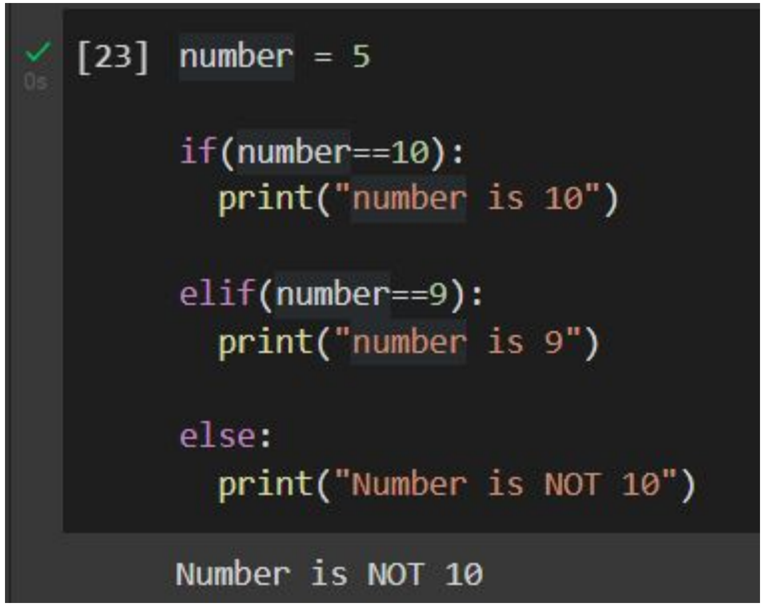
```
number = 5
if(number>5):
  print("Condition fulfilled")
```

No Output

Conditional Statement

```
if condition:
   statement
elif condition:
   statement
elif condition:
   statement
else:
   statement
```

If all conditions are not fulfilled ("if" and "elif") => "else" statement will be executed



```
number = 5

if(number==10):
  print("number is 10")

elif(number==9):
  print("number is 9")

else:
  print("Number is NOT 10")
```

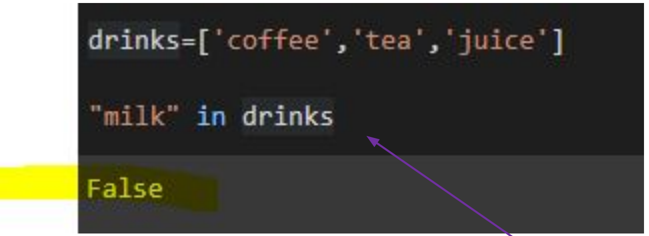
```
Number is NOT 10
```

Not Fulfilled

Not Fulfilled

✓

Check if specific value exist in a List



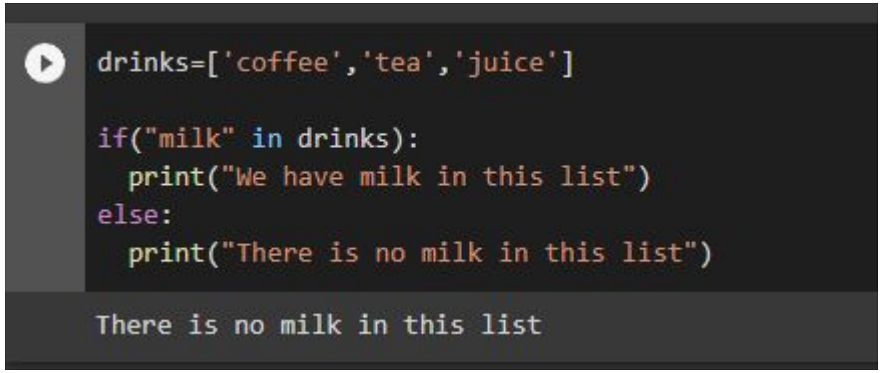
```
drinks=['coffee','tea','juice']

"milk" in drinks
```

```
False
```

1. Check if milk exist in the list name “drinks”

2.Result= False (milk doesn’t exist in this list)

Combine it with conditional statement:



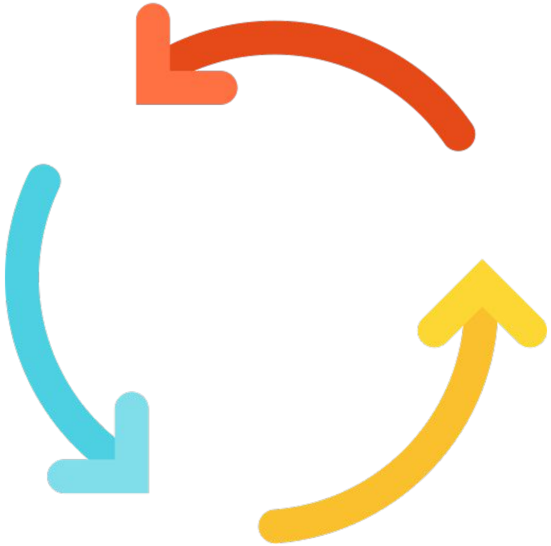
```
drinks=['coffee','tea','juice']

if("milk" in drinks):
  print("We have milk in this list")
else:
  print("There is no milk in this list")
```

```
There is no milk in this list
```

Loop/Iteration Statement in Python

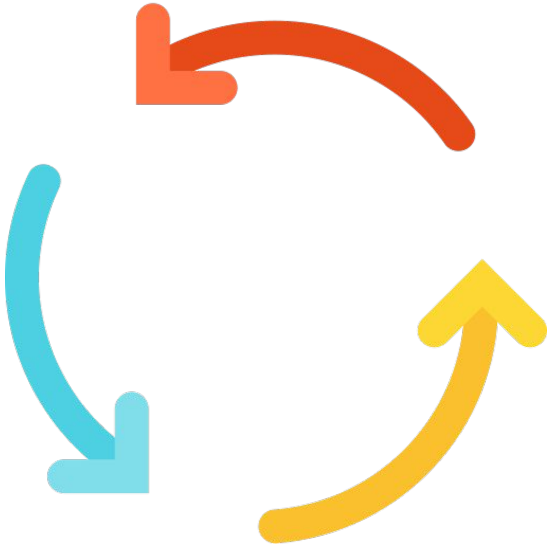
Loop/Iteration in Python

Looping means repeating something over and over until a particular condition is satisfied



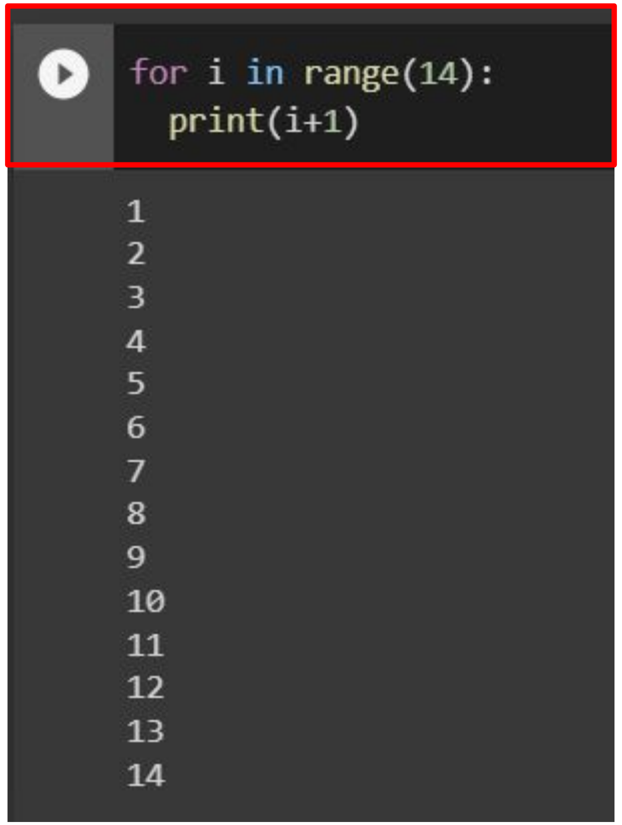
Loop/Iteration in Python

In Python, Loop functions will help us to execute code repeatedly, without the need to rewrite the code.



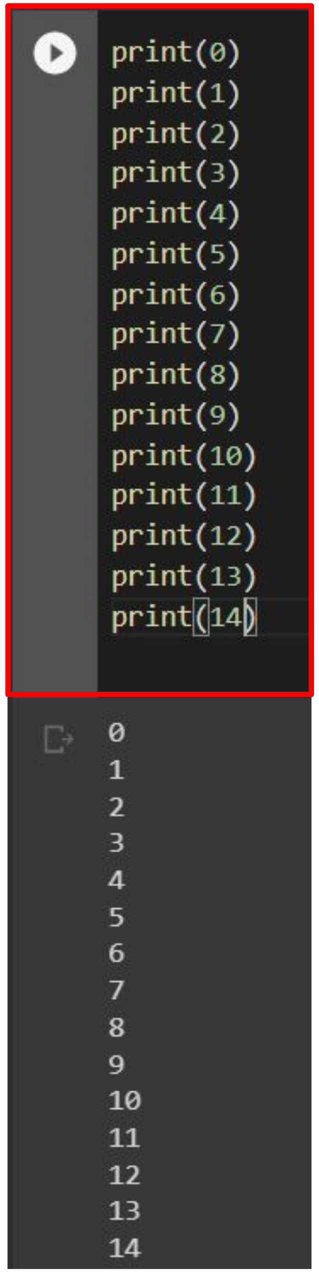
Loop/Iteration in Python

Compare 2 of these codes (refer to red boxes).
Which one is more efficient ?



```
for i in range(14):
  print(i+1)
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
```

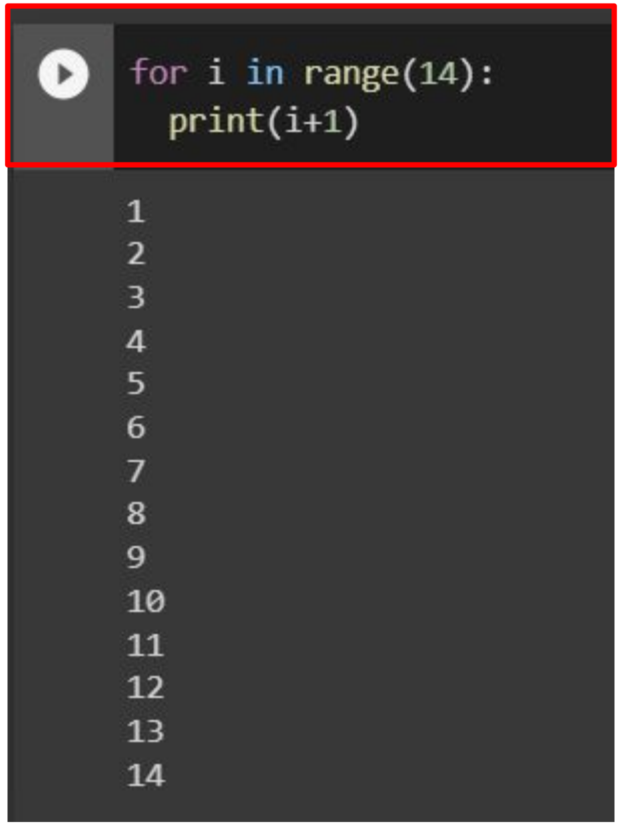


```
print(0)
print(1)
print(2)
print(3)
print(4)
print(5)
print(6)
print(7)
print(8)
print(9)
print(10)
print(11)
print(12)
print(13)
print(14)
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
```

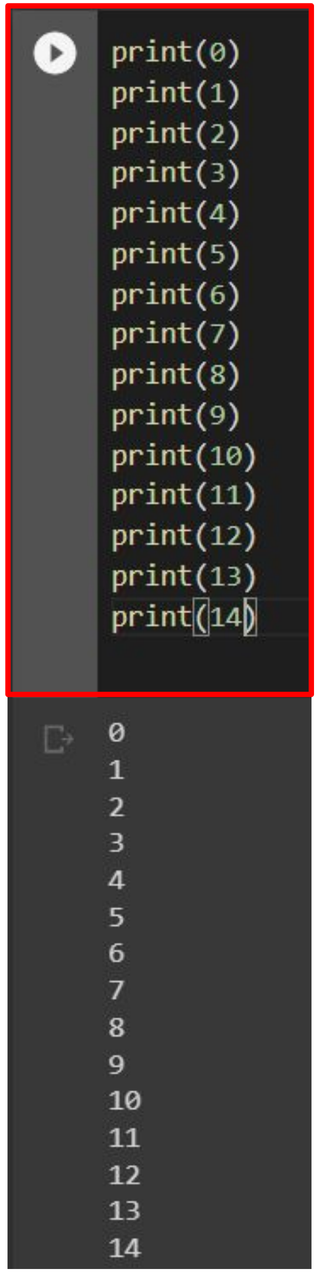
Loop/Iteration in Python

Compare 2 of these codes (refer to red boxes).
Which one is more efficient ?



```
for i in range(14):
  print(i+1)
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
```

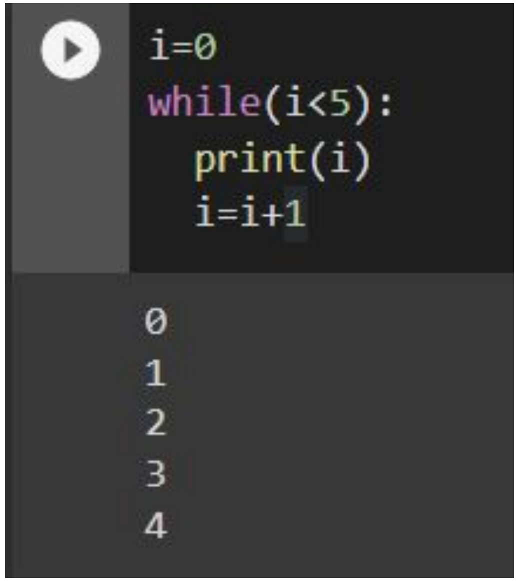


```
print(0)
print(1)
print(2)
print(3)
print(4)
print(5)
print(6)
print(7)
print(8)
print(9)
print(10)
print(11)
print(12)
print(13)
print(14)
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
```

While Loop

While loop will enable iteration until the condition is False.



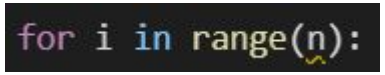
```
i=0
while(i<5):
  print(i)
  i=i+1
```

```
0
1
2
3
4
```

condition

Loop stops since the condition is not fulfilled (False). Printing stop when *i=5*

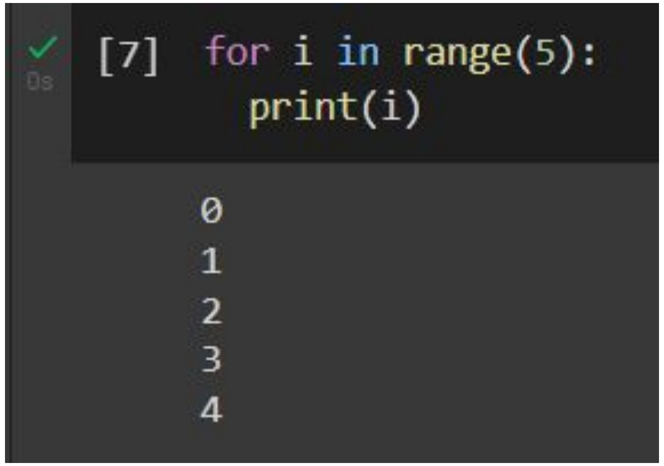
For Loop – Range



```
for i in range(n):
```

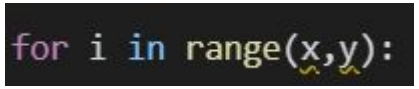
Doing n iteration from 0

I.e. In this case, inputting 5 will return 5 iterations starting from 0



```
for i in range(5):
  print(i)

0
1
2
3
4
```

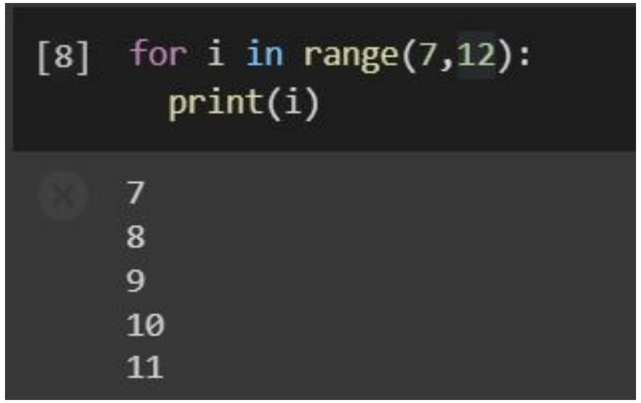
For Loop – Range



```
for i in range(x,y):
```

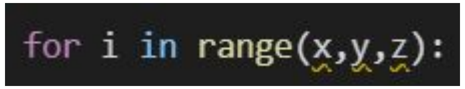
Doing iteration from x to y

I.e. In this case, inputting (7,12) will return iteration from 7 to 11



```
[8] for i in range(7,12):
      print(i)

    7
    8
    9
    10
    11
```

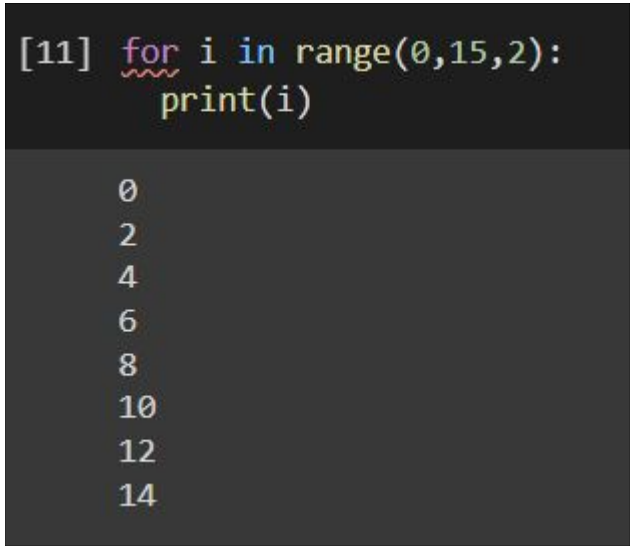
For Loop – Range



```
for i in range(x,y,z):
```

Doing iteration from x to y with incremental z

I.e. In this case, inputting (0,15) will return iteration from 0 to 15 with 2 incremental value

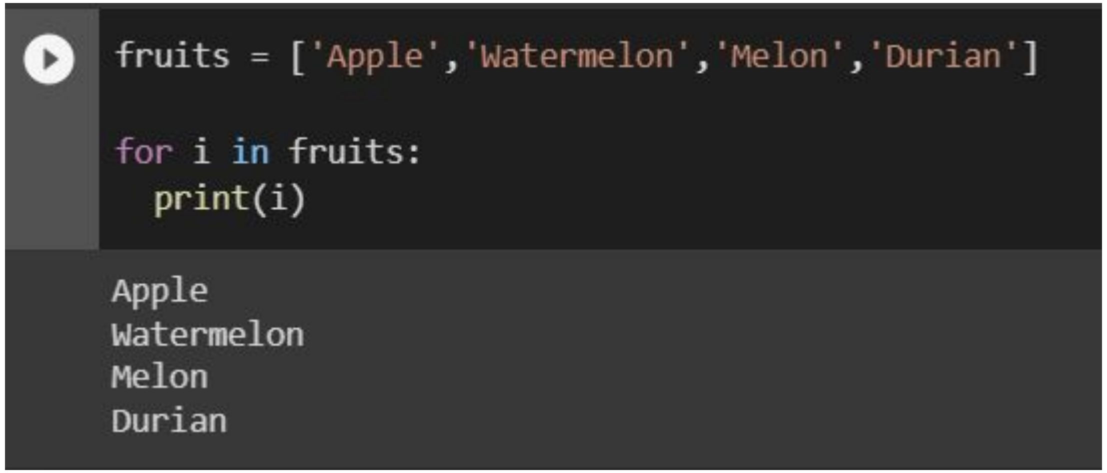


```
[11] for i in range(0,15,2):
       print(i)

     0
     2
     4
     6
     8
     10
     12
     14
```

Loop/Iteration Statement in Python List

For Loop – List

For loop can be used on list to iterate the list value.
Examples are below:



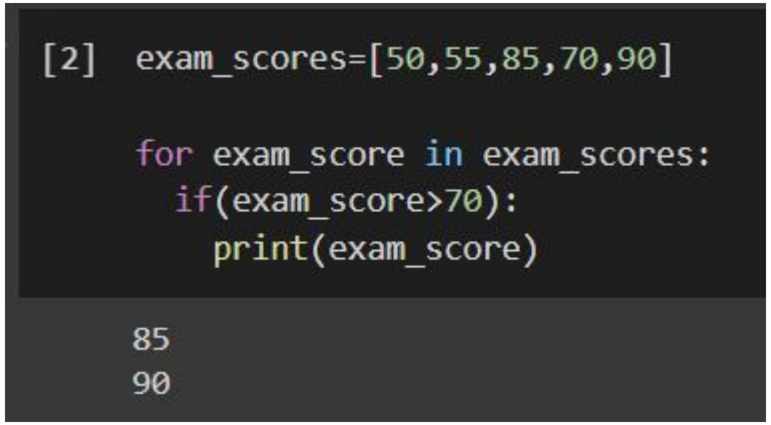
```
fruits = ['Apple','Watermelon','Melon','Durian']

for i in fruits:
  print(i)
```

```
Apple
Watermelon
Melon
Durian
```

The loop will iterate all of the values from the list starting from index 0

For Loop – with Condition Statement

Condition statement can be applied in the for loop.



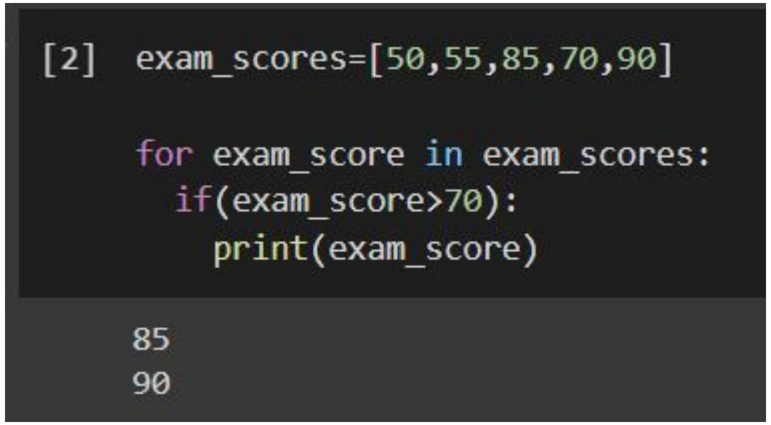
```
[2]  exam_scores=[50,55,85,70,90]

     for exam_score in exam_scores:
       if(exam_score>70):
         print(exam_score)
```

```
85
90
```

This loop will only print *exam_scores* which are higher than 70

For Loop – with Condition Statement

Condition statement can be applied in the for loop.

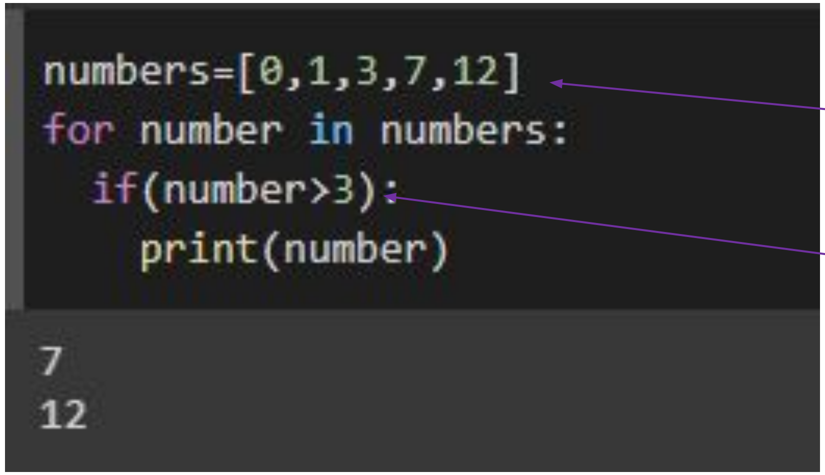


```
[2]  exam_scores=[50,55,85,70,90]

     for exam_score in exam_scores:
       if(exam_score>70):
         print(exam_score)
```

```
     85
     90
```

This loop will only print *exam_scores* which are higher than 70

Loop with Condition Statement



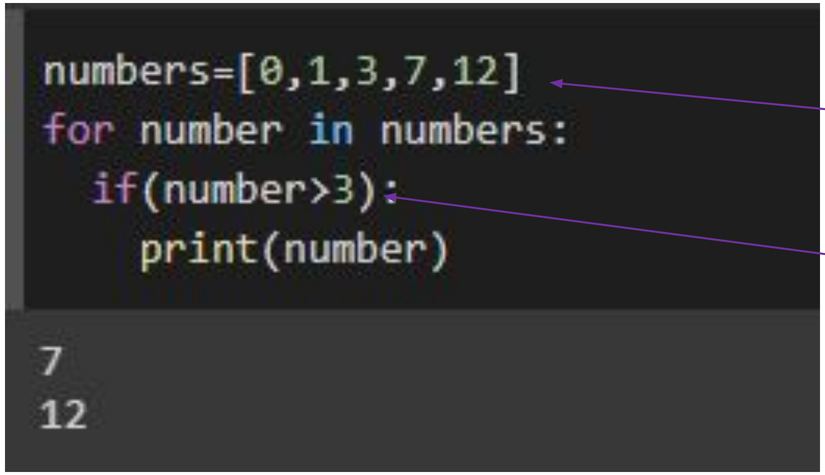
```
numbers=[0,1,3,7,12]
for number in numbers:
  if(number>3):
    print(number)
```

```
7
12
```

Loop all list

Print if value is above 3

Loop with Condition Statement



```
numbers=[0,1,3,7,12]
for number in numbers:
  if(number>3):
    print(number)
```

```
7
12
```

Loop all list

Print if value is above 3

Challenge 1

Given this list of numbers:

Numbers=[4,5,10,20,40,60,80]

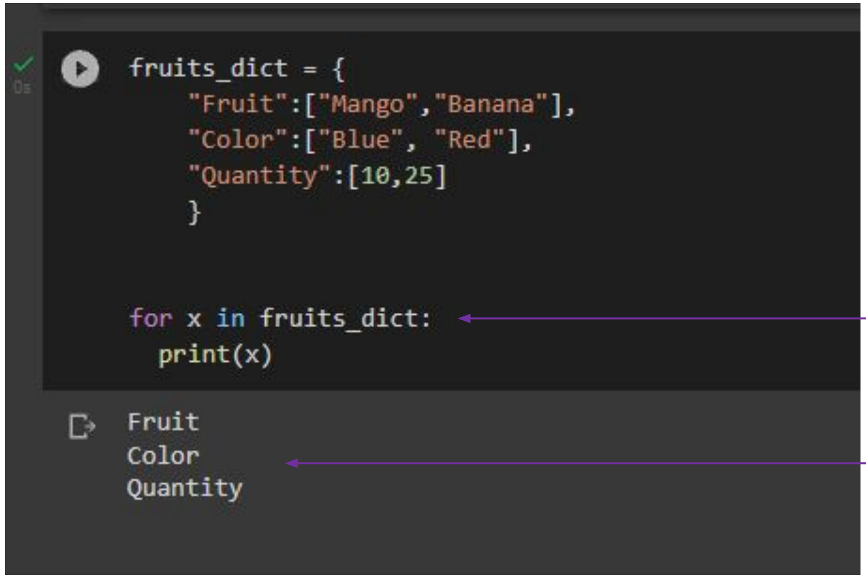
□ Create new list called “new_list” containing all number higher than 10

Hint = Use “For” Loop with conditional statement and list operation i.e **append()**

Loop/Iteration Statement in Python Dictionary

Iteration in Dictionary

Iteration in dictionary will always **return key** from the dictionary



```
fruits_dict = {
    "Fruit":["Mango","Banana"],
    "Color":["Blue", "Red"],
    "Quantity":[10,25]
    }


for x in fruits_dict:
  print(x)
```

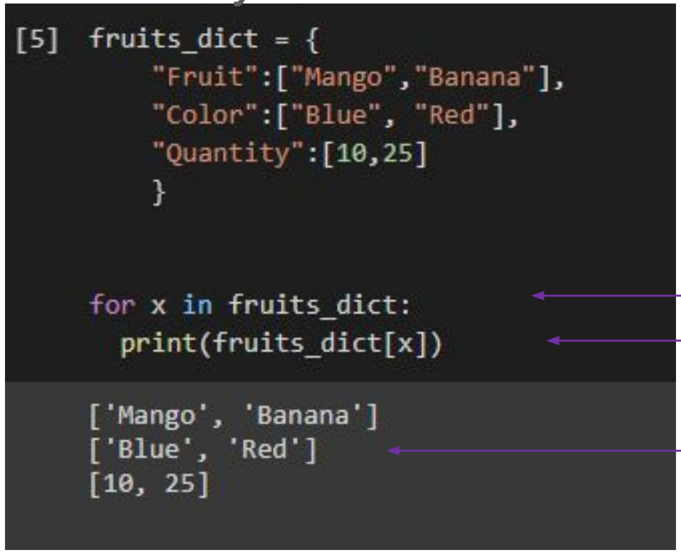
```
Fruit
Color
Quantity
```

For Loop on Dictionary

Keys

Iteration in Dictionary

Having the key value in the loop. We can get the value of the keys in dictionary.



```
[5]  fruits_dict = {
         "Fruit":["Mango","Banana"],
         "Color":["Blue", "Red"],
         "Quantity":[10,25]
         }


     for x in fruits_dict:
       print(fruits_dict[x])
```

```
['Mango', 'Banana']
['Blue', 'Red']
[10, 25]
```

For Loop on Dictionary

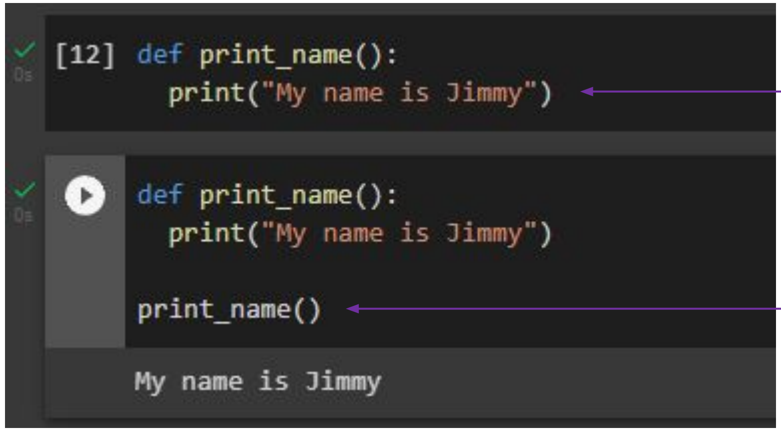
Print Value

Value

Function

Function

A block of code that only runs when it is called.



```
[12] def print_name():
       print("My name is Jimmy")
```

```
def print_name():
  print("My name is Jimmy")

print_name()
```

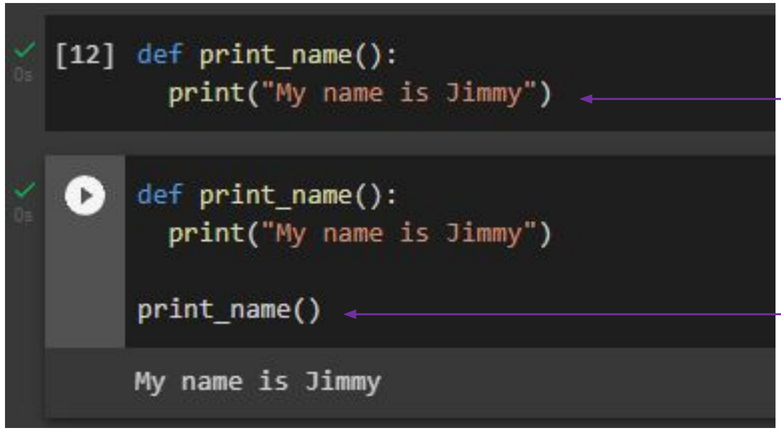
```
My name is Jimmy
```

Function not called => no output

Function called

Function

A block of code that **only runs when it is called.** Functions allow developers to reduce repetition in their code because they can execute the same block of code multiple time resulting in organized code



```
[12] def print_name():
       print("My name is Jimmy")
```

```
def print_name():
  print("My name is Jimmy")

print_name()
```

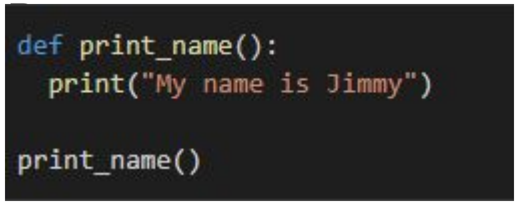
```
My name is Jimmy
```

Function not called => no output

Function called

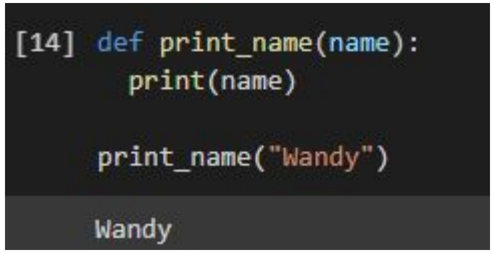
Defining Function

Function without



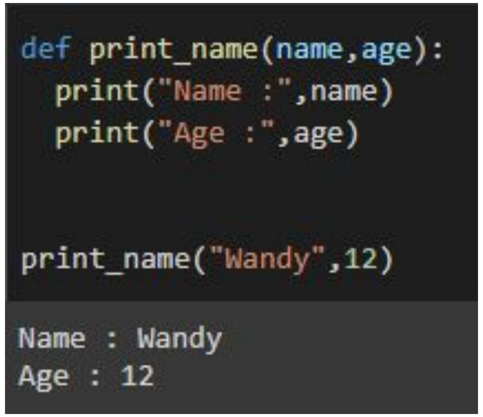
```
def print_name():
  print("My name is Jimmy")

print_name()
```

Function with Parameter



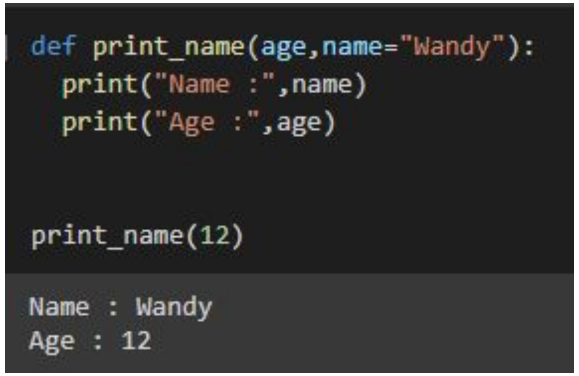
```
[14] def print_name(name):
       print(name)

     print_name("Wandy")

     Wandy
```

Function with multiple parameter



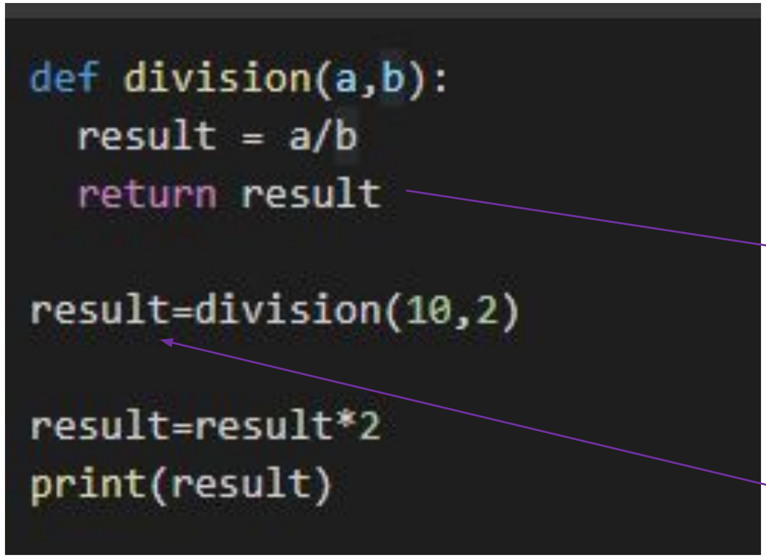
```
def print_name(name,age):
  print("Name :",name)
  print("Age :",age)


print_name("Wandy",12)

Name : Wandy
Age : 12
```

Function with Default Parameter



```
def print_name(age,name="Wandy"):
  print("Name :",name)
  print("Age :",age)


print_name(12)

Name : Wandy
Age : 12
```

"Return" in Function



```
def division(a,b):
  result = a/b
  return result

result=division(10,2)

result=result*2
print(result)
```

return the value of "result"

value is passed

Challenge 2

Create a function to find the area of circle. Take *r* as an input in parameter:

Area of Circle formula = πr^2

Challenge 3

Create a function to add two input of numbers (a and b) and print

- The addition of both number
- The multiplication of both number
- Highest number

Assignment

Assignment 2

Kerjakan soal-soal yang terdapat pada bagian assignment di Canvas dengan judul

Topic 3 & 4 - Assignment | Hands-On Python 1 di **Google Colab**.

Kumpulkan link Google Colab dan beri nama notebook dengan format nama:

Topik 3 4 - [Nama Lengkap]

Available from	Until
Mar 20 at 09.00 PM	Mar 27 at 11.59PM

Reference

- https://www.flaticon.com/free-icon/arrows_1635629
- https://prepinsta.com/python/if-else-statement/

Thanks!
Any Questions?