. if

. if else

. else if

. nested if

In [1]:
```
if True:
    print('Data science')
```

```
Data science
```

In [3]:
```
if False:
    print('Data science')
print('bye for now')
```

```
bye for now
```

In [4]:
```
if True:
    print('Data scince')
print('bye for now')
```

```
Data scince
bye for now
```

In [5]:
```
if True:
    print('Data science')
else:
    print('bye for now')
```

```
Data science
```

Lets do one program as if divide by 2 then reminder is 0 then it is even number if reminder is not 0 then it is odd number

In [6]:
```
if False:
    print('Data science')
else:
    print('bye for now')
```

```
bye for now
```

In [7]:
```
x = 4
r = x % 2

if r == 0:
        print('Even number')
```

```
Even number
```

In [10]:
```
#to print only even number
x = 5
```

```
r = x % 2

if r == 0:
    print('Even number')
```

In [9]:
```
x = 5
r = x % 2

if r == 0:
    print('Even number')
else:
    print('odd number')
```

```
odd number
```

In [16]:
```
x = 4
r = x % 2

if r == 0:
    print('Even number')

if r == 1:
    print('odd number')

if r == 2:
    print('even number')
```

```
Even number
```

In [17]:
```
x = 1

if x == 1:
    print('one')
if x == 2:
    print('Two')
if x == 3:
    print('Three')
if x == 4:
    print('four')
```

```
one
```

In [18]:
```
x = 5
r = x % 2

if r == 0:
    print('Even number')
else:
    print('Odd Number')
```

```
Odd Number
```

In [19]:
```
x = 3
r = x % 2

if r == 0:
    print('Even number')
    if x>5:
        print('greater number')
```

```
else:
    print('Odd Number')
```

```
Odd Number
```

In [20]:
```
x = 4
r = x % 2

if r == 0:
    print('Even number')
    if x>5:
        print('greater number')

else:
    print('Odd Number')
```

```
Even number
```

In [21]:
```
x = 4
r = x % 2

if r == 0:
    print('Even number')

    if x>5:
        print('greater number')


    else:
        print('number is lesser number')

else:
    print('Odd Number')
```

```
Even number
number is lesser number
```

In [22]:
```
# elif it wont check till the block once you find the output it wont go to next
# you can try with multiple parameter 1, 2 & 3 value in x

x = 4

if(x == 1):
    print('one')
elif(x == 2):
    print('Two')
elif(x == 3):
    print('Three')
elif(x == 4):
    print('four')
```

```
four
```

In [23]:
```
# elif it wont check till the block once you find the output it wont go to next
# you can try with multiple parameter 1, 2 & 3 value in x

x = 7

if(x == 1):
    print('one')
elif(x == 2):
    print('Two')
```

```
elif(x == 3):
    print('Three')
elif(x == 4):
    print('four')
```

In [24]:
```
# elif it wont check till the block once you find the output it wont go to next
# you can try with multiple parameter 1, 2 & 3 value in x

x = 7

if(x == 1):
    print('one')

elif(x == 2):
    print('Two')
elif(x == 3):
    print('Three')
elif(x == 4):
    print('four')

else:
    print('number not found')
```

```
number not found
```

Loops

LOOPS -- in programing world some time we keep on repeating , may be you want to repeat 5 statement so one way is copy & paste multiple times or other way is

if you want to print the datascience 1000 times then what you will you cant copy for 1000 times , if you want to print 1000 times then you cant do manualy . that is the reason why we need to apply loop -> 2 type of loops -- While loop & For loop

In [25]:
```
i = 1
while i<=5:  #condition
    print('Data science')
    i = i + 1 #increment
```

```
Data science
Data science
Data science
Data science
Data science
```

In [26]:
```
i = 1

while i<=5:
    print('data science') # when we mention end then new line will not create
    j = 1
    while j<=4:
        print('technology')
        j = j + 1

    i = i + 1
    print()
```

```
    # the output which we got is very lengty but how to make them one line lets
```

```
data science
technology
technology
technology
technology

data science
technology
technology
technology
technology

data science
technology
technology
technology
technology

data science
technology
technology
technology
technology

data science
technology
technology
technology
technology
```

In [27]:
```
i = 1
while i<=5:
    print(' datascience', end = "") # when we mention end then new line will not
    j = 1
    while j<=4:
        print(' technology', end="")
        j = j + 1

    i = i + 1
    print()
```

```
datascience technology technology technology technology
datascience technology technology technology technology
datascience technology technology technology technology
datascience technology technology technology technology
datascience technology technology technology technology
```

In [ ]:
```

```

In [11]:
```
for i in range(1,51):

    if i%3 == 0:

        print(i)
print('end')
```

```
3
6
9
12
15
18
21
24
27
30
33
36
39
42
45
48
end
```

In [12]:
```python
for i in range(1,51):

    if i%3 == 0:
        continue
    print(i)
print('end')
```

```
1
2
4
5
7
8
10
11
13
14
16
17
19
20
22
23
25
26
28
29
31
32
34
35
37
38
40
41
43
44
46
47
49
50
end
```

In [13]:
```
for i in range(1,51):

    if i%3 == 0 or i%5 == 0:
        continue
    print(i)
#print('end')
# it will skip all the value which is divisible by 3 or 5
```

```
1
2
4
7
8
11
13
14
16
17
19
22
23
26
28
29
31
32
34
37
38
41
43
44
46
47
49
```

In [14]:
```
for i in range(1,50):
    if i%3 == 0 or i%5 == 0:
        continue
    print(i)
print('end')
# when you apply and you wont get the value which is divisible by both 3 & 5 (15
```

```
1
2
4
7
8
11
13
14
16
17
19
22
23
26
28
29
31
32
34
37
38
41
43
44
46
47
49
end
```

In [15]:
```python
# i dont want to print the values which are even numbers that means print only o

for i in range(1,51):

    if (i%2 == 0):
        #print('even')
        continue
    else:
        print(i)
print('bye')
```

```
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
bye
```

In [16]:
```python
# i dont want to print the values which are even numbers that means print only o

for i in range(1,51):

    if (i%2 == 0):
        #print('even')
        continue
    else:
        print(i)
print('bye')
```

```
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
bye
```

PRINTING PATTERN IN PYTHON

In [18]:
```
print('# # # #')
print('# # # #')
print('# # # #')
print('# # # #')
```

```
# # # #
# # # #
# # # #
# # # #
```

In [19]:
```
for i in range(1,5):
    i=i+1
    print('# # # # ')
```

```
# # # #
# # # #
# # # #
# # # #
```

In [20]:
```
for i in range(1,5):
    if i<=5:
        print('# # # #')
```

```
# # # #
# # # #
# # # #
# # # #
```

In [21]:
```
for j in range(4):
    print('#')
```

```
#
#
#
#
```

In [22]:
```
for j in range(4):
    print('# # # #')
```

```
# # # #
# # # #
# # # #
# # # #
```

In [23]:
```
for j in range(4):
    print('#', end = " ")
```

```
# # # #
```

In [24]:
```
for j in range(4):
    print('#', end=" ")

for j in range(4):
    print('#', end=" ")
```

```
# # # # # # # #
```

In [25]:
```
for j in range(4):
    print('#', end=" ")

print()

for j in range(4):
    print('#', end=" ")
```

```
# # # #
# # # #
```

In [26]:
```
for j in range(4):
    print('#', end="  ")

print()

for j in range(4):
    print('#', end="  ")

print()

for j in range(4):
    print('#', end="  ")

print()

for j in range(4):
    print('#', end="  ")
```

```
#  #  #  #
#  #  #  #
#  #  #  #
#  #  #  #
```

In [27]:
```
for i in range(4):
    for j in range(4):
```

```
        print('#', end="  ")
    print()
    # pease use debug mode in pycharm
```

```
#  #  #  #
#  #  #  #
#  #  #  #
#  #  #  #
```

In [28]:
```
for i in range(4):
    for j in range(i+1):
        print('#', end = " ")
    print()
```

```
#
# #
# # #
# # # #
```

In [29]:
```
for i in range(1,5):
    print("# "*i)
```

```
#
# #
# # #
# # # #
```

In [30]:
```
for i in range(1,5):
    for j in range(4):
        if i>j:
            print("#",end=" ")
    print()
```

```
#
# #
# # #
# # # #
```

In [31]:
```
list(range(5))
```

Out[31]:
```
[0, 1, 2, 3, 4]
```

In [32]:
```
for i in range(4):
    for j in range(i):
        print('#', end="  ")
    print()
```

```
#
#  #
#  #  #
```

In [33]:
```
for i in range(4):
    for j in range(i+1):
        print('#', end="  ")
    print()
```

```
#
#  #
#  #  #
#  #  #  #
```

In [34]:
```
for i in range(4):
    for j in range(4-i):
        print('#', end="  ")
    print()
```

```
#  #  #  #
#  #  #
#  #
#
```

In [35]:
```
for i in range(1,5):
    print("# "*(5-i))
```

```
# # # #
# # #
# #
#
```

FOR ELSE

- For|Else in python
- In other language for else not supportable but in python it is supportable

eg- lets print the number from 1- 20 & we dont want print number which is divisible by 5

In [36]:
```
nums = [12,15,18,21,26, 30,40]

for num in nums:
    if num % 5 == 0:
        print(num)
```

```
15
30
40
```

In [37]:
```
nums = [12,14,18,21,25,30,35]

for num in nums:
    if num % 5 == 0:
        print(num)
```

```
25
30
35
```

In [38]:
```
nums = [12,14,18,21,25,20]

for num in nums:
    if num % 5 == 0:
        print(num)
```

```
25
20
```

In [39]:
```
nums = [12,14,18,21,20,25]

for num in nums:
    if num % 5 == 0:
```

```
        print(num)
        break
```

```
20
```

In [40]:
```
nums = [12,14,18,21,25,20,10]

for num in nums:
    if num % 5 == 0:
        print(num)
        break
```

```
25
```

In [41]:
```
nums = [10,14,18,21,5,10]

for num in nums:
    if num % 5 == 0:
        print(num)
        break #it will print only 1 number then it break
```

```
10
```

In [42]:
```
nums = [7,14,18,21,23,27] #hear there is no number which is divisible by 5 we go

for num in nums:
    if num % 5 == 0:
        print(num)
        break
```

In [43]:
```
nums = [7,14,18,21,23,27,29] #hear there is no number which is divisible by 5 we

for num in nums:
    if num % 5 == 0:
        print(num)
        break
else:
   print('Number Not Found') #every iteration it cheking condition
```

```
Number Not Found
```

In [44]:
```
nums = [7,14] #hear there is no number which is divisible by 5 we got output as
for num in nums:
    if num % 5 == 0:
        print(num)
        break
    else:
        print('Number Not Found') #every iteration it cheking condition
```

```
Number Not Found
Number Not Found
```

In [45]:
```
nums = [7,14,18,21,23,27] #hear there is no number which is divisible by 5 we go

for num in nums:
    if num % 5 == 0:
        print(num)
        break
else:
        print('Number Not Found') # hear else we dont write in if block but we c
```

```
Number Not Found
```

In [46]:
```
nums = [10,14,18,21,20,27] #hear there is no number which is divisible by 5 we g

for num in nums:
    if num % 5 == 0:
        print(num)
        break
else:
        print('Not Found')
```

```
10
```

In [47]:
```
nums = [10,14,18,21,20,27,30] #hear there is no number which is divisible by 5 w

for num in nums:
    if num % 5 == 0:
        print(num)
        #break
else:
        print('Not Found')
```

```
10
20
30
Not Found
```

In [48]:
```
nums = [10,14,18,21,20,27] #hear there is no number which is divisible by 5 we g
for num in nums:
    if num % 5 == 0:
        print(num)
        break
else:
        print('Not Found')
```

```
10
```

- prime number - how to check given number is prime number OR

• not : PRIME NUMBER 7 13 19

In [50]:
```
num = 14

for i in range(2,num):
    if num % i == 0:
        print('Not prime Number')
        break
else:
    print('Prime Number')
```

```
Not prime Number
```

In [51]:
```
num = 13

for i in range(2,num):
    if num % i == 0:
        print('Not prime Number')
        break
else:
    print('Prime Number')
```

```
Prime Number
```

array in python

In [ ]:
```
from array import *
arr = array('i',[])

n = int(input('Enter the length of the array'))

for i in range(5):
    x = int(input('Enter the next value'))
    arr.append(x)
print(arr)
```

In [ ]:
```
from array import *
arr = array('i',[])

n = int(input('Enter the length of the array'))

for i in range(5):
    x = int(input('Enter the next value'))
    arr.append(x)
print(arr)
```

In [ ]:
```
from array import *
arr = array('i',[])

n = input('Enter the length of the array')

for i in range(5):
    x = input('Enter the next value')
    arr.append(x)
print(arr)
```

In [1]:
```
# Way of creating array using numpy
```

In [2]:
```
from numpy import *
arr = array([1,2,3,4,5])
print(arr)
type(arr)
```

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

Out[2]:
```
numpy.ndarray
```

In [3]:
```
print (arr.dtype)
```

```
int64
```

In [4]:
```
arr = array ([1,2,3,4,5.9])
print(arr)
```

```
[1.  2.  3.  4.  5.9]
```

In [5]:
```
print(arr.dtype)
```

```
float64
```

In [7]:
```
arr2 = array([1,2,3,4,5.9],float)
arr2
```

Out[7]:
```
array([1. , 2. , 3. , 4. , 5.9])
```

In [8]:
```
arr3 = array([1,2,3,4,5.6],int)
arr3
```

Out[8]:
```
array([1, 2, 3, 4, 5])
```

In [9]:
```
import numpy as np
```

In [10]:
```
arr4 = np.linspace(0, 16, 10) # break the code between 10 spaces between 0 to 16
arr4
```

Out[10]:
```
array([ 0.        ,  1.77777778,  3.55555556,  5.33333333,  7.11111111,
        8.88888889, 10.66666667, 12.44444444, 14.22222222, 16.        ])
```

In [11]:
```
arr5 = np.arange(0,10,2) # arange - as range
arr5
```

Out[11]:
```
array([0, 2, 4, 6, 8])
```

In [12]:
```
arr6 = np.zeros(5)
arr6
```

Out[12]:
```
array([0., 0., 0., 0., 0.])
```

In [13]:
```
arr7 = np.ones(5)
arr7
```

Out[13]:
```
array([1., 1., 1., 1., 1.])
```

In [ ]:
```

```