PY-STEP INTO CS

Let’s start from 0 instead of 1

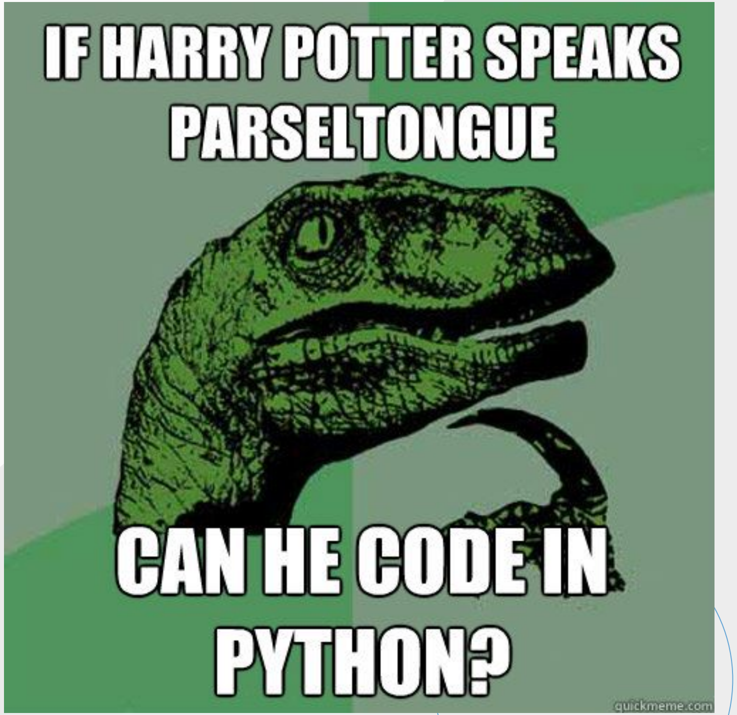


TABLE OF CONTENTS

00 FIRST STEP IN 0

01 PYTHON INSTALLATION

02 LET’S EAT PY

03 GAME TIME

AKADEMIK

- DIPLOMA IN COMPUTER SCIENCES - UiTM
- LEGOOM ACADEMY PROGRAMMING TRACK
- CERT BINARY EXPLOITATION – ASIA PACIFIC UNIV
- CERT CYBERSECURITY 0421A – CISCO NET ACADEMY

BIOGRAPHY

NOOR RAIHAN ABD RAHIM

- EXPERIENCE
  - 4 YEARS (FREELANCE GRAPHIC DESIGN / DEVELOPER)
    - ❑ LUSTREFX ENTERPRISE
    - ❑ LEGOOM VALLEY CONSULTANCY & SOLUTION
    - ❑ JOHORBIZ DIGITAL
    - ❑ SOSIAL MARKET SAYA ENTERPRISE
- ACHIEVEMENT
  - DEVELOP
    - ❑ JOMTERAPI BOOKING SYSTEM
    - ❑ I-LAHAD MANAGEMENT SYSTEM
    - ❑ STUDENT4U COMPLAINT MANAGEMENT SYSTEM
    - ❑ MANUFACTURING INVENTORY MGT SYSTEM
    - ❑ EZY CERT – ECERT GENERATOR
    - ❑ NURSKILL ACADEMY JOHOR
    - ❑ TUTOR ASSISTED IN PHP WEB SYSTEM DEV AT KKTM PASIR MAS
  - MALAYSIA TOP 10 DATA INNOVATION FINALIST
  - TOP 60 GLOBAL TRACELABS DEFCON OSINT COMPETITION
  - TOP 20 IN BATTLE OF HACKER COMPETITION

00

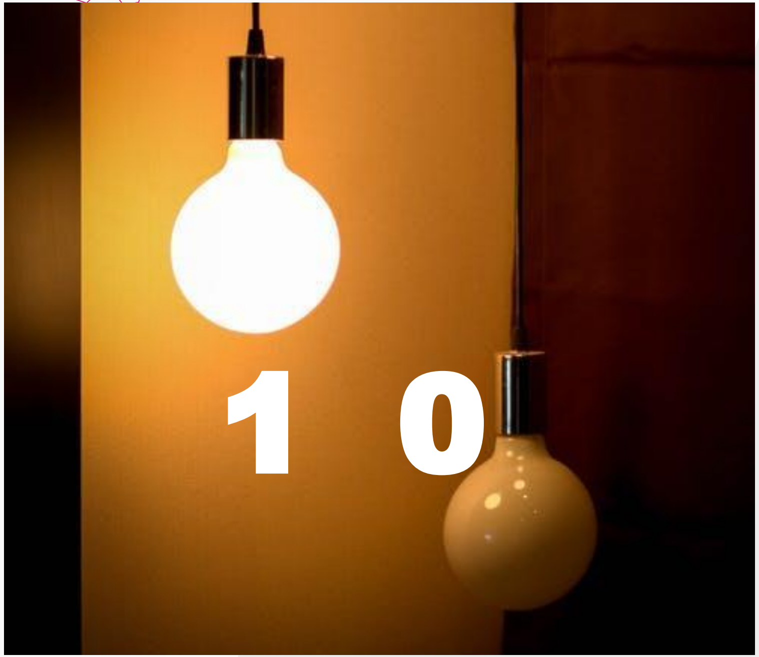
FIRST STEP IN 0

LET’S TALK WITH COMPUTER

0 1

But they also can store other information such as images, emoji and video but how??

Light bulb



ON = 1
OFF = 0

ASCII (American Standard Code)

For example

A-65 B-66 C-67
D-68
E-69 F-70 G-71 H-72
I-73 J-74 K-75 L-76
M-77 N-78 O-79
P-80
Q-81 R-82. S-83
T-84
U-85 V-86. W-87
X-88
Y-89 Z-90

Let’s say “HI”

H-72 I-73

So computers read as
72 73

How about other character for example emoji?

In ASCII we have Unicode

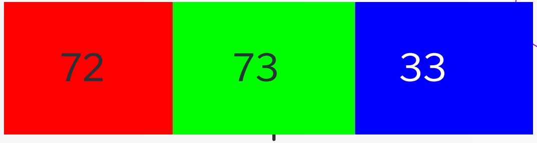
So, that means emoji contain more 10 than before

1F600	😀	grinning face
1F601	😁	grinning face with smiling eyes
1F602	😂	face with tears of joy

If you're familiar this is hexadecimal,
so if we change to decimal is 128512
And turn to binary is
11111011000000000

So, based on this light bulb theory.
The same things happened to others such as images

Eg: Images represent it colour using same



#484921

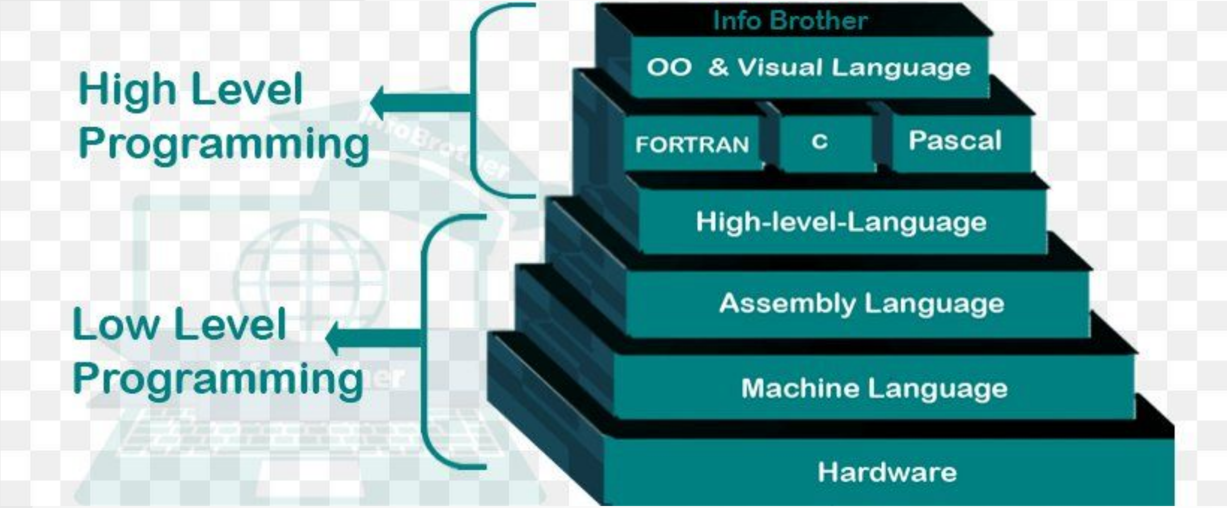
SO LIKE HOWWWW WE CAN COMMUNICATE WITH THEM

When you are trying to code assembler after java

WHAT IS CODING?

Coding is a process or a way to communicate with a computer by giving the instruction to the computer through it owns language

PROGRAMMING LANGUAGE

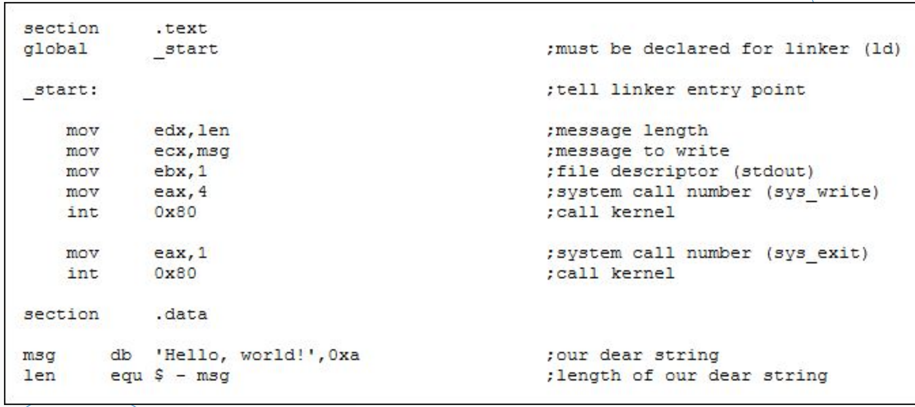


LOW LEVEL PROGRAMMING

- Better control over the code, and the possibility for programmers to optimize the program
- Programs run more efficiently, even if there is limited memory and storage
- Writing low-level code requires the programmer to have good knowledge of the hardware being used
- Different hardware, different style of code

```
01 1111 0111 1100 0000 1000 0100 0010 0001 0001 0011
11 1100 0000 1000 0100 0010 0001 0001 0011 0101 1111
00 1000 0100 0010 0001 0001 0011 0101 1111 0111 1100
00 0010 0001 0001 0011 0101 1111 0111 1100 0000 1000
01 0001 0011 0101 1111 0111 1100 0000 1000 0100 0010
11 0101 1111 0111 1100 0000 1000 0100 0010 0001 0001
11 0111 1100 0000 1000 0100 0010 0001 0001 0011 0101
00 0000 1000 0100 0010 0001 0001 0011 0101 1111 0111
00 0100 0010 0001 0001 0011 0101 1111 0111 1100 0000
```

MACHINE LANGUAGE

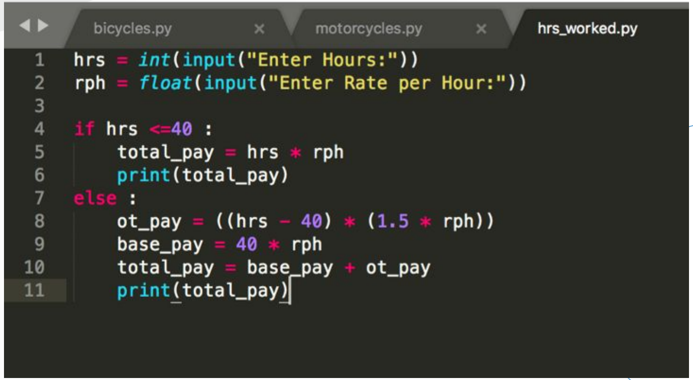


```
section     .text
global      _start                              ;must be declared for linker (ld)

_start:                                         ;tell linker entry point

    mov     edx,len                             ;message length
    mov     ecx,msg                             ;message to write
    mov     ebx,1                               ;file descriptor (stdout)
    mov     eax,4                               ;system call number (sys_write)
    int     0x80                                ;call kernel

    mov     eax,1                               ;system call number (sys_exit)
    int     0x80                                ;call kernel

section     .data

msg     db  'Hello, world!',0xa                 ;our dear string
len     equ $ - msg                             ;length of our dear string
```

ASSEMBLY LANGUAGE

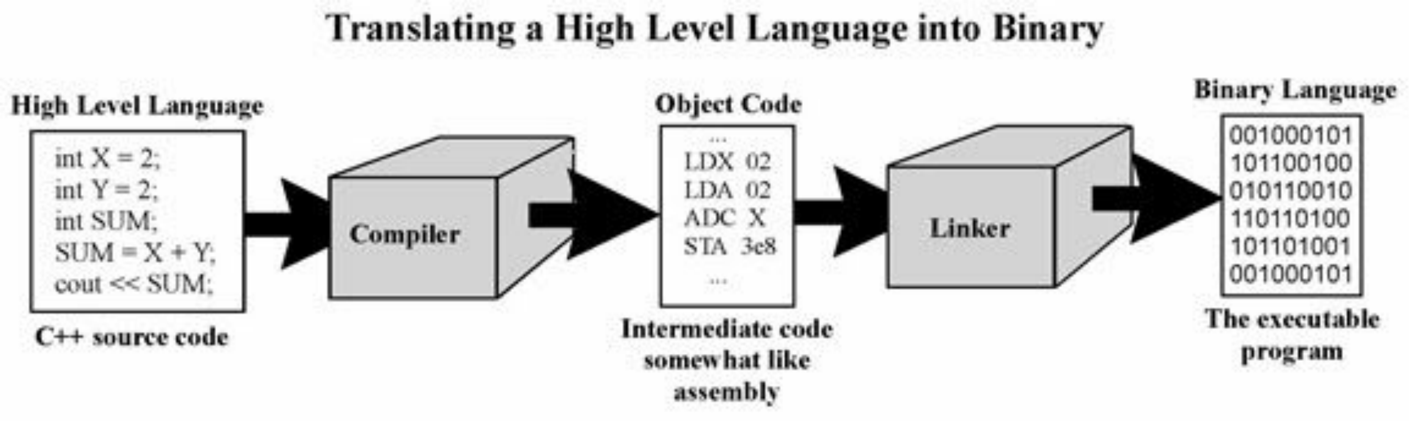
HIGH LEVEL PROGRAMMING

- Easier to understand and learn since use the “English” language.
- Easy to debug.
- One can be run on multiple platforms.
- Require compiler or interpreter to translate into the machine code



```python
hrs = int(input("Enter Hours:"))
rph = float(input("Enter Rate per Hour:"))

if hrs <=40 :
    total_pay = hrs * rph
    print(total_pay)
else :
    ot_pay = ((hrs - 40) * (1.5 * rph))
    base_pay = 40 * rph
    total_pay = base_pay + ot_pay
    print(total_pay)
```

THE TRANSLATION



Translating a High Level Language into Binary

```
int X = 2;
int Y = 2;
int SUM;
SUM = X + Y;
cout << SUM;
```

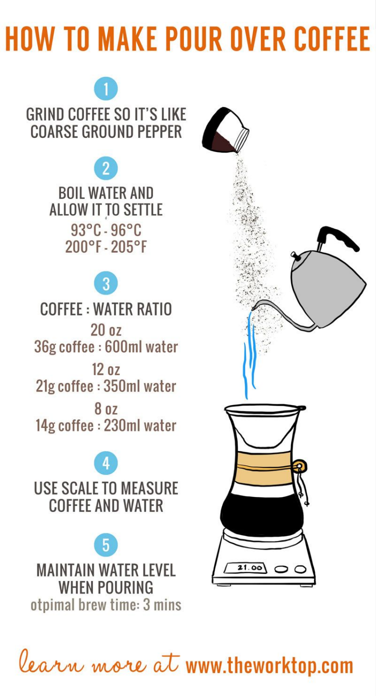
```
...
LDX  02
LDA  02
ADC  X
STA  3e8
...
```

```
001000101
101100100
010110010
110110100
101101001
001000101
```

THE KHAWARIZMI

WHAT IS ALGORITHM

- Set of instruction or rules to solve a particular problem.



HOW TO MAKE POUR OVER COFFEE

1 GRIND COFFEE SO IT'S LIKE COARSE GROUND PEPPER

2 BOIL WATER AND ALLOW IT TO SETTLE
93°C - 96°C
200°F - 205°F

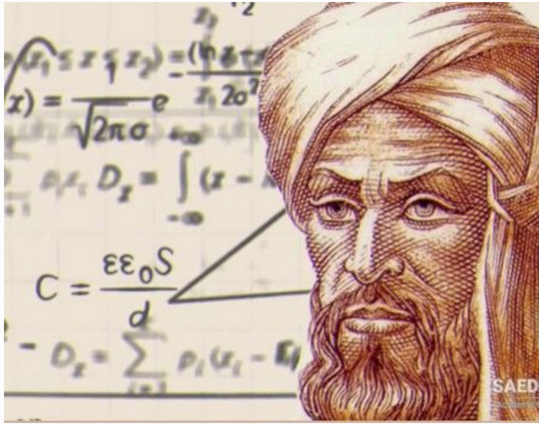
3 COFFEE : WATER RATIO
20 oz
36g coffee : 600ml water
12 oz
21g coffee : 350ml water
8 oz
14g coffee : 230ml water

4 USE SCALE TO MEASURE COFFEE AND WATER

5 MAINTAIN WATER LEVEL WHEN POURING
otpimal brew time: 3 mins

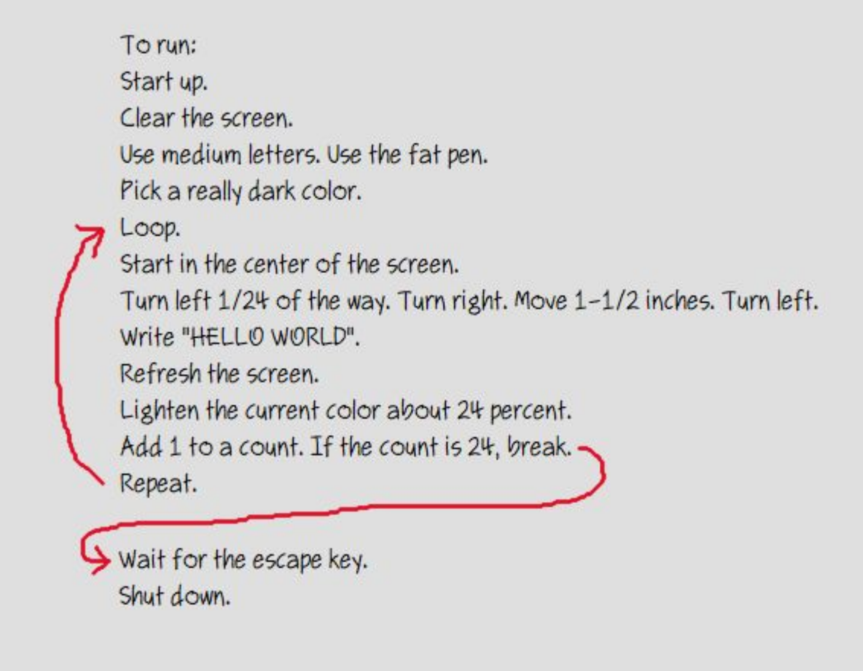
learn more at www.theworktop.com

Coffee Algorithm



PSEUDOCODE

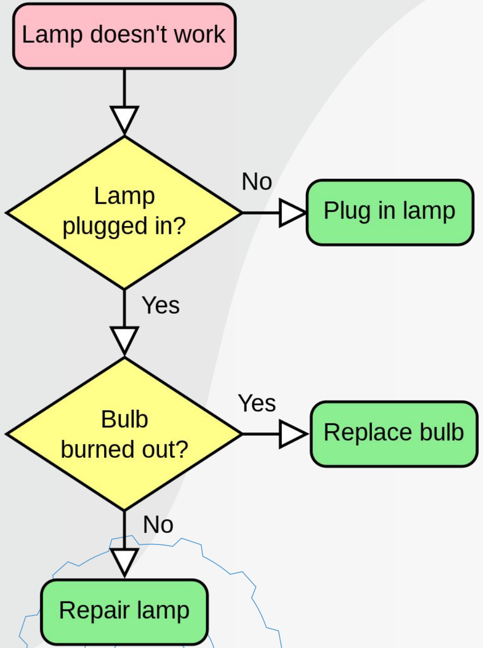
- Plain language description of the steps in an algorithm or process.

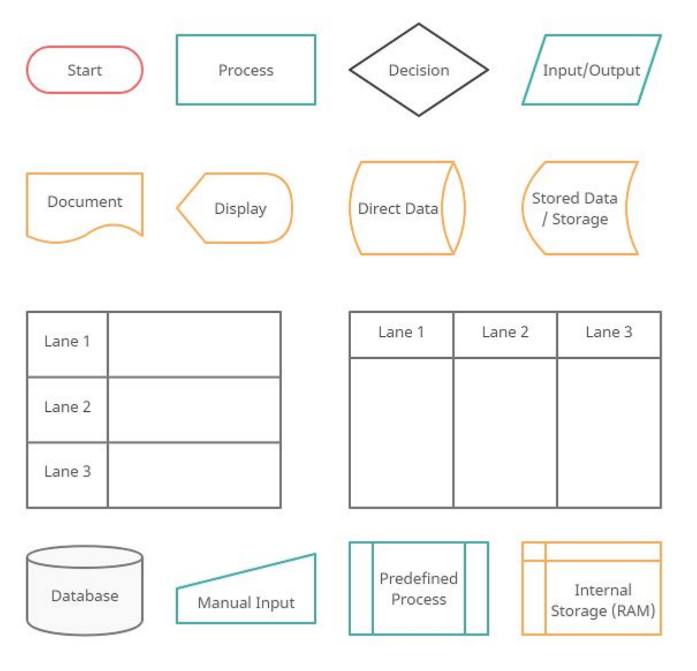


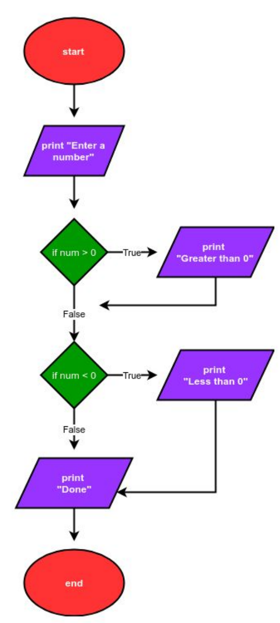
```
To run:
Start up.
Clear the screen.
Use medium letters. Use the fat pen.
Pick a really dark color.
Loop.
Start in the center of the screen.
Turn left 1/24 of the way. Turn right. Move 1-1/2 inches. Turn left.
Write "HELLO WORLD".
Refresh the screen.
Lighten the current color about 24 percent.
Add 1 to a count. If the count is 24, break.
Repeat.

Wait for the escape key.
Shut down.
```

FLOWCHART

- A representation of sequence of steps in an algorithms using symbols and shapes.



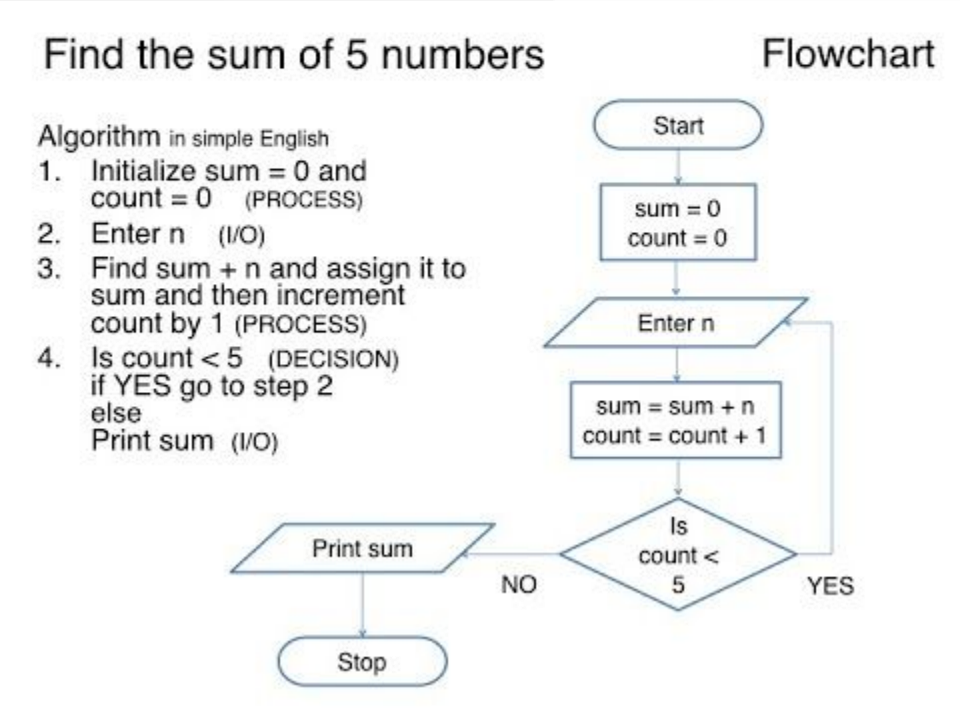




FLOWCHART & PSEUDO

Find the sum of 5 numbers using flowchart or pseudocode

FLOWCHART & PSEUDO



Find the sum of 5 numbers

Flowchart

Algorithm in simple English

1. Initialize sum = 0 and count = 0 (PROCESS)
2. Enter n (I/O)
3. Find sum + n and assign it to sum and then increment count by 1 (PROCESS)
4. Is count < 5 (DECISION)
   if YES go to step 2
   else
   Print sum (I/O)

Start

sum = 0
count = 0

Enter n

sum = sum + n
count = count + 1

Is count < 5

YES

NO

Print sum

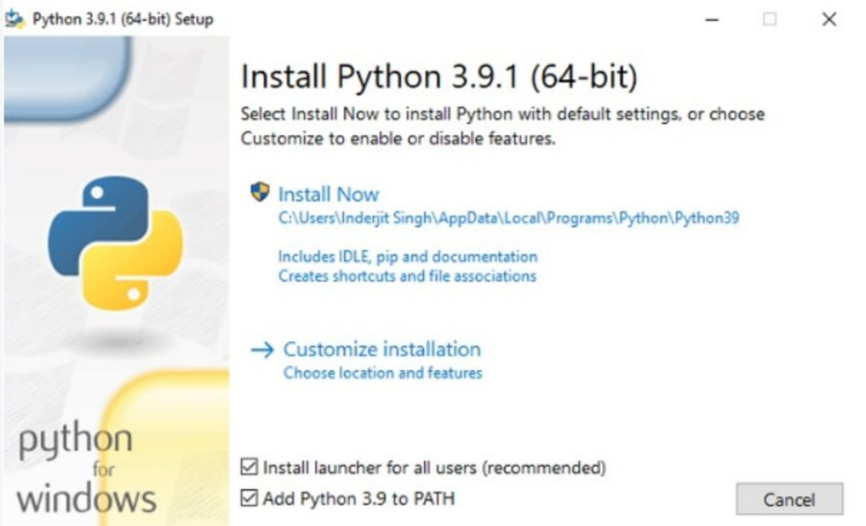
Stop

01

PYTHON INSTALLATION

INSTALLATION

1. Visit **https://www.python.org/downloads/** and choose which version are suitable. In this case, we are using Python3 since it have the latest update.
2. After finish the downloading, run the executable Installer and make sure to **tick on *Add Python 3.x to the path***



Install Python 3.9.1 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\Inderjit Singh\AppData\Local\Programs\Python\Python39

Includes IDLE, pip and documentation
Creates shortcuts and file associations

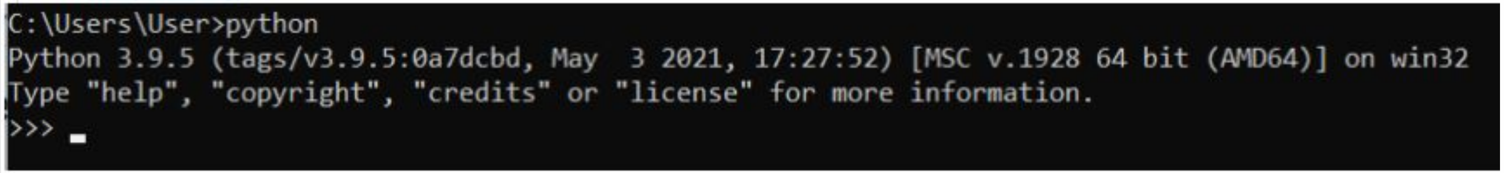
→ Customize installation
Choose location and features

☑ Install launcher for all users (recommended)
☑ Add Python 3.9 to PATH

INSTALLATION

3. Verify if the Python is successfully installed.

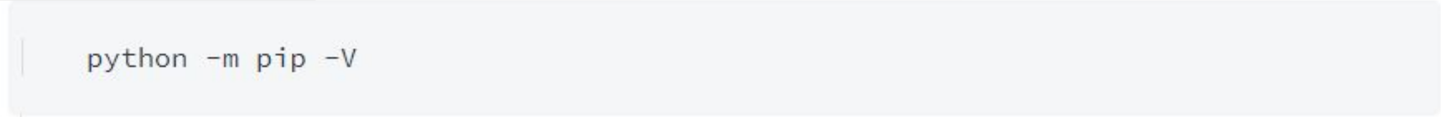
- Open the command prompt.
- Type 'python' and press enter.
- The output should be like this if the python is successfully installed.



```
C:\Users\User>python
Python 3.9.5 (tags/v3.9.5:0a7dcbd, May  3 2021, 17:27:52) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

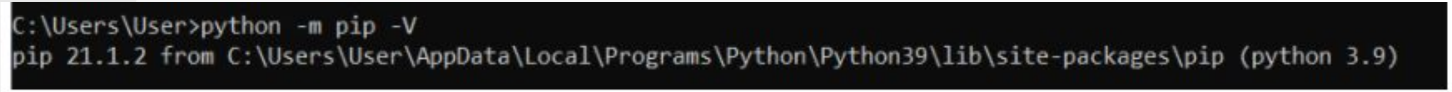
INSTALLATION

4. Verify if pip is executable through the python command.
   - Open the command prompt.
   - Type this command in the command prompt and press enter.



```
python -m pip -V
```

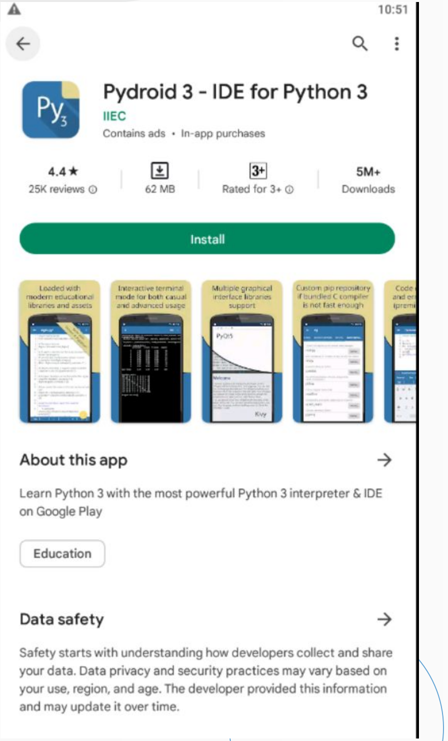
- If the installation successful, you may see the pip version displayed as below.



```
C:\Users\User>python -m pip -V
pip 21.1.2 from C:\Users\User\AppData\Local\Programs\Python\Python39\lib\site-packages\pip (python 3.9)
```

INSTALLATION

For mobile version, you can download Pydroid 3 on Playstore



10:51

Pydroid 3 - IDE for Python 3

IIEC

Contains ads • In-app purchases

4.4★ 25K reviews | 62 MB | Rated for 3+ | 5M+ Downloads

Install

Loaded with modern educational libraries and assets
Interactive terminal mode for both casual and advanced usage
Multiple graphical interface libraries support
Custom pip repository if bundled C compiler is not fast enough

About this app

Learn Python 3 with the most powerful Python 3 interpreter & IDE on Google Play

Education

Data safety

Safety starts with understanding how developers collect and share your data. Data privacy and security practices may vary based on your use, region, and age. The developer provided this information and may update it over time.

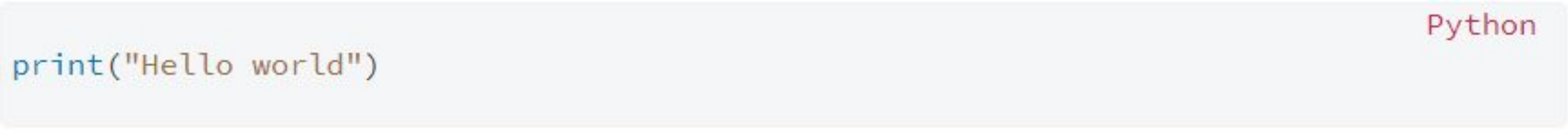
02

Let’s eat PY

Executing Py file

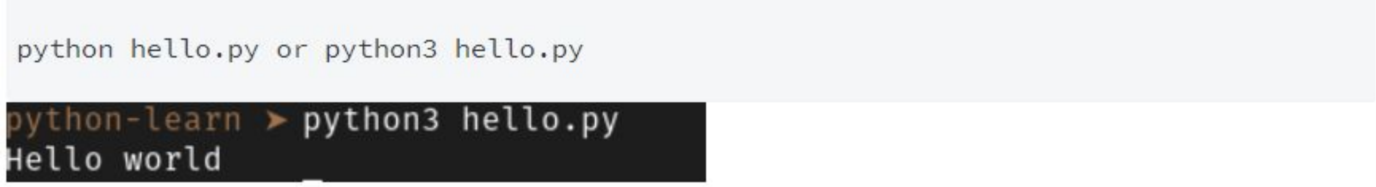
All python programs and files are ended with **.py** as file extension. It can be executed by using the terminal or command prompt. Both are using the same concept and same command.

You can try create a file called **hello.py** and type this code in the file.



```python
print("Hello world")
```

After that, type the following command in the terminal and make sure the terminal directory are in the same file as the python file.

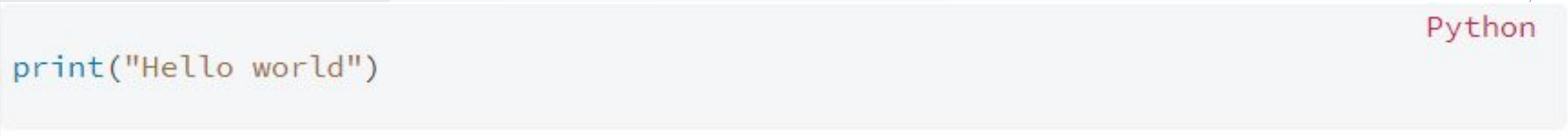


```
python hello.py or python3 hello.py
```

```
python-learn ➤ python3 hello.py
Hello world
```

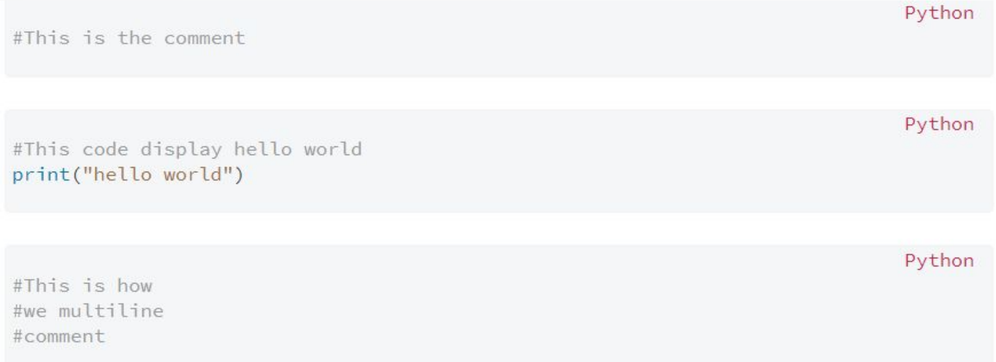
Displaying Output & Comments

All programming language start with displaying **Hello World**. In Python3, displaying text are very simple.



```
print("Hello world")
```

Commenting are very important in explaining what's goin on with our code. It will make the code more readable.



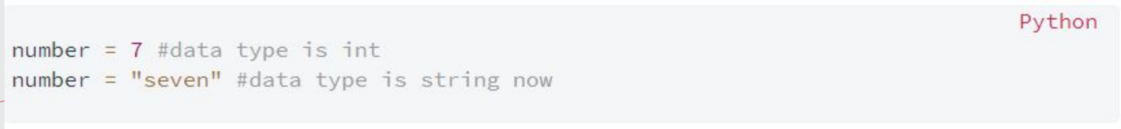
```
#This is the comment
```

```
#This code display hello world
print("hello world")
```

```
#This is how
#we multiline
#comment
```

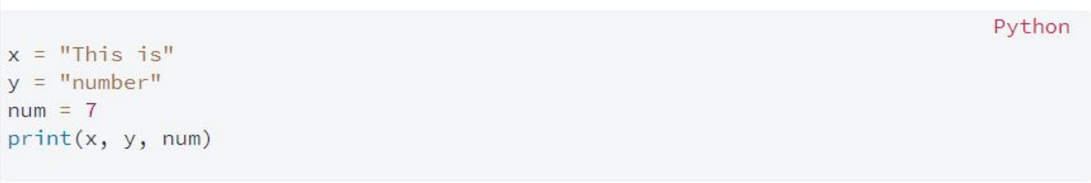
Variables

Declaring variable in python are quite simple. You do not need to declare any particular data type and even change the same variable with another *datatype.*



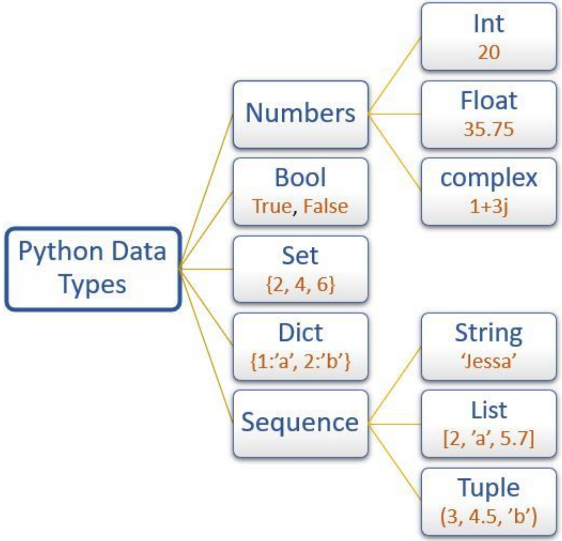
Python

```
number = 7 #data type is int
number = "seven" #data type is string now
```



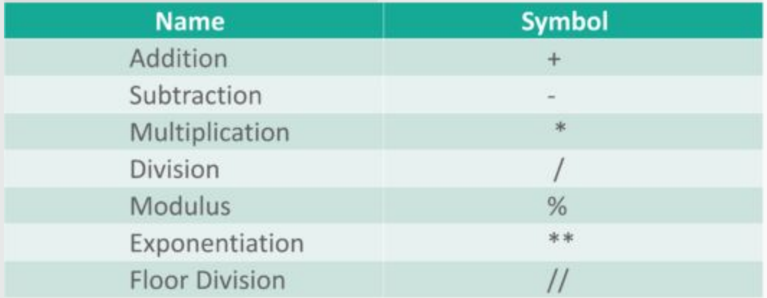
Python

```
x = "This is"
y = "number"
num = 7
print(x, y, num)
```

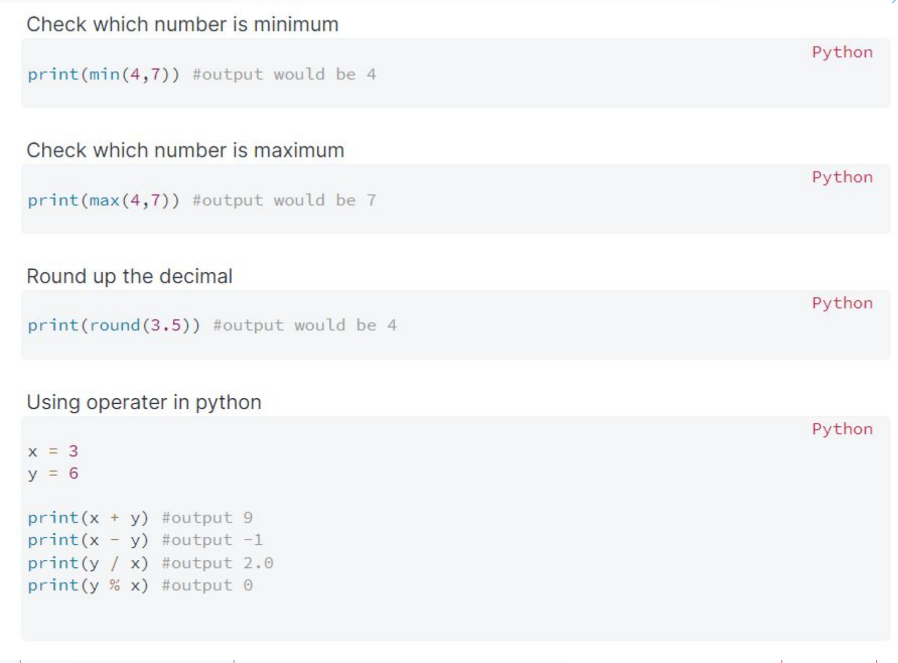


Operator

In programming, of course we can do math and we usually use math in programming to solve a problem.



Name	Symbol
Addition	+
Subtraction	-
Multiplication	*
Division	/
Modulus	%
Exponentiation	**
Floor Division	//



Check which number is minimum

```
print(min(4,7)) #output would be 4
```

Check which number is maximum

```
print(max(4,7)) #output would be 7
```

Round up the decimal

```
print(round(3.5)) #output would be 4
```

Using operater in python

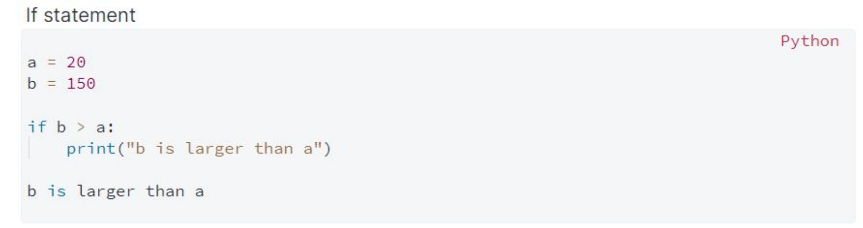
```
x = 3
y = 6

print(x + y) #output 9
print(x - y) #output -1
print(y / x) #output 2.0
print(y % x) #output 0
```

Condition

In programming, computer can do decision but we need to declare what it needs to do if particular condition is fulfilled.

Example, if the number entered by the user is bigger than 3, then the computer will display “The number is bigger than 3!”

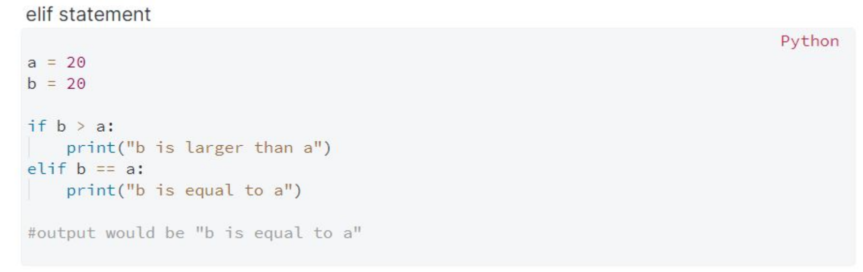
In mathematics, we have >,<,>=,<=, ==, != symbol right? Yes we use it too in programming.



If statement

Python

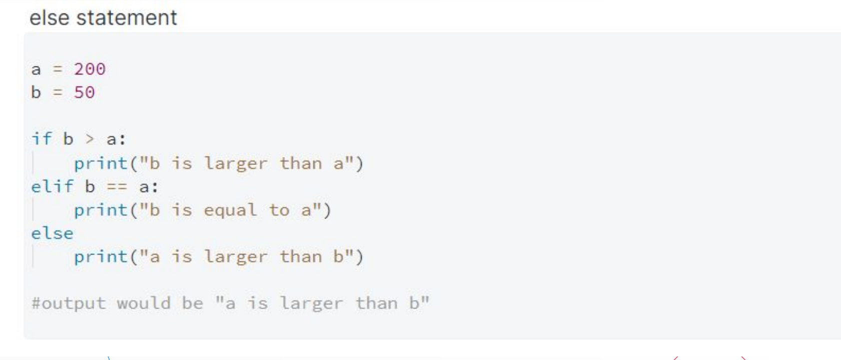
```
a = 20
b = 150

if b > a:
    print("b is larger than a")

b is larger than a
```



elif statement

Python

```
a = 20
b = 20

if b > a:
    print("b is larger than a")
elif b == a:
    print("b is equal to a")

#output would be "b is equal to a"
```

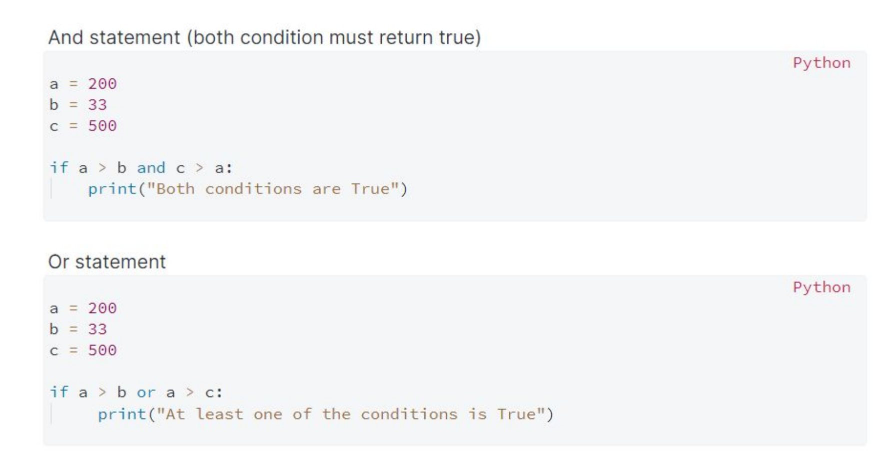


else statement

```
a = 200
b = 50

if b > a:
    print("b is larger than a")
elif b == a:
    print("b is equal to a")
else
    print("a is larger than b")

#output would be "a is larger than b"
```

Condition

We also can use the mathematical logic such as AND, OR



And statement (both condition must return true)

Python

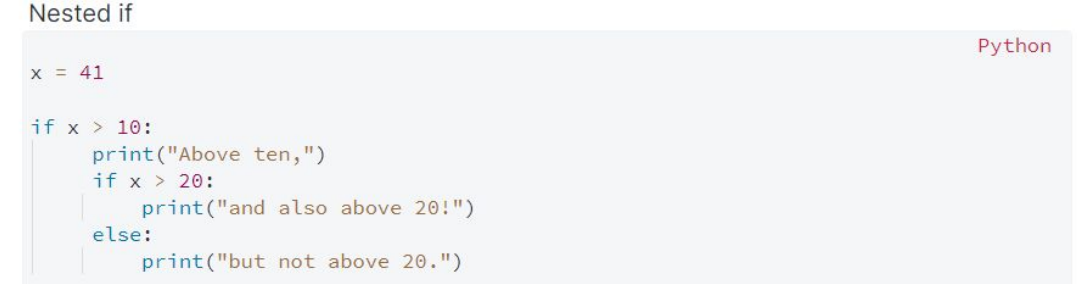
```
a = 200
b = 33
c = 500

if a > b and c > a:
    print("Both conditions are True")
```

Or statement

Python

```
a = 200
b = 33
c = 500

if a > b or a > c:
    print("At least one of the conditions is True")
```

We also can create a condition in condition. If x above ten then if x above 20 then it will display (“x above 10 and 20!”)



Nested if

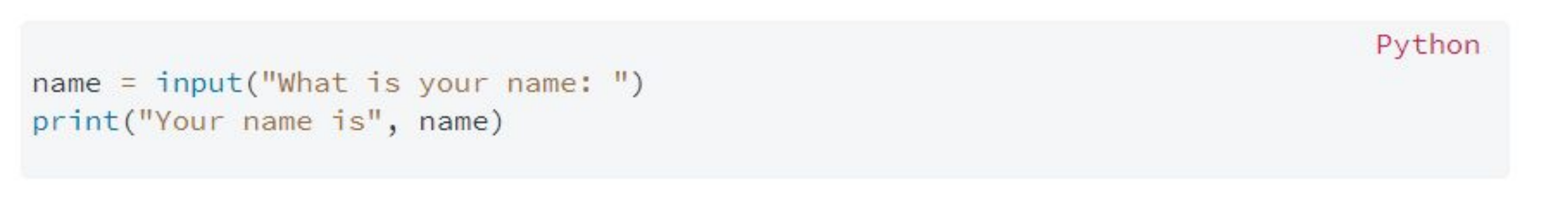
Python

```
x = 41

if x > 10:
    print("Above ten,")
    if x > 20:
        print("and also above 20!")
    else:
        print("but not above 20.")
```

User Input

User input is very important in every programming language. This is how we receive user input in python3.

****Remind: input in python always return as string****

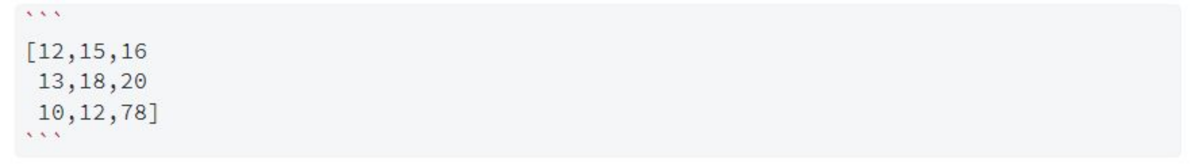


```
name = input("What is your name: ")
print("Your name is", name)
```

Array

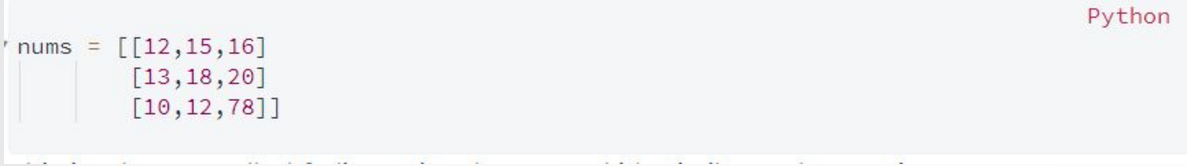
Array is very important in a programming language. Array make us as programmer able to store multiple temporary data or value in single variable.

we can imagine the array as matrices in modern mathematics



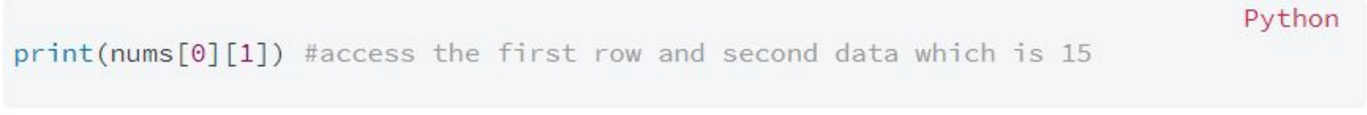
~~~~
```
[12,15,16
 13,18,20
 10,12,78]
```
~~~~

similar to matrices right? let's convert it into the array



```python
nums = [[12,15,16]
         [13,18,20]
         [10,12,78]]
```

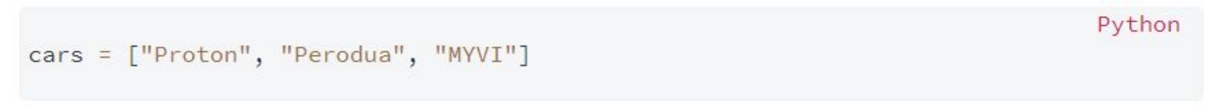
this is what we called 2 dimensional array and it's similar to the matrices concept and we can access it like this



```python
print(nums[0][1]) #access the first row and second data which is 15
```

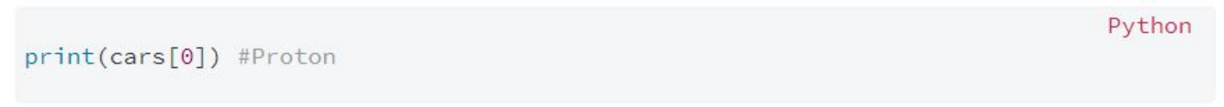
Array

This is how we can create an basic array in the python



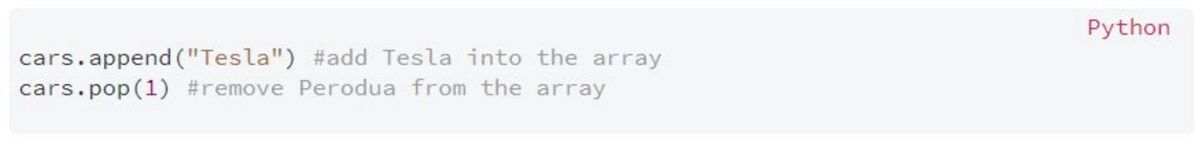
```python
cars = ["Proton", "Perodua", "MYVI"]
```

So we can access the above array by mentioning its index and it's always start with 0



```python
print(cars[0]) #Proton
```

we also can append and remove the array

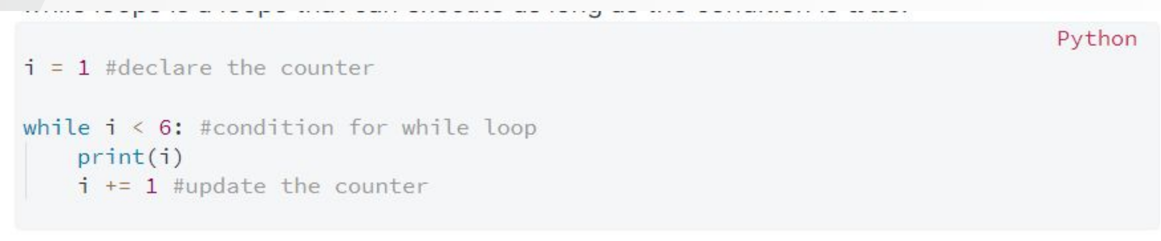


```python
cars.append("Tesla") #add Tesla into the array
cars.pop(1) #remove Perodua from the array
```

Loops

There is two type of loops in python.
- **while** loops
- **for** loops

While Loop
while loops is a loops that can execute as long as the condition is **true**.



Python

```python
i = 1 #declare the counter

while i < 6: #condition for while loop
    print(i)
    i += 1 #update the counter
```

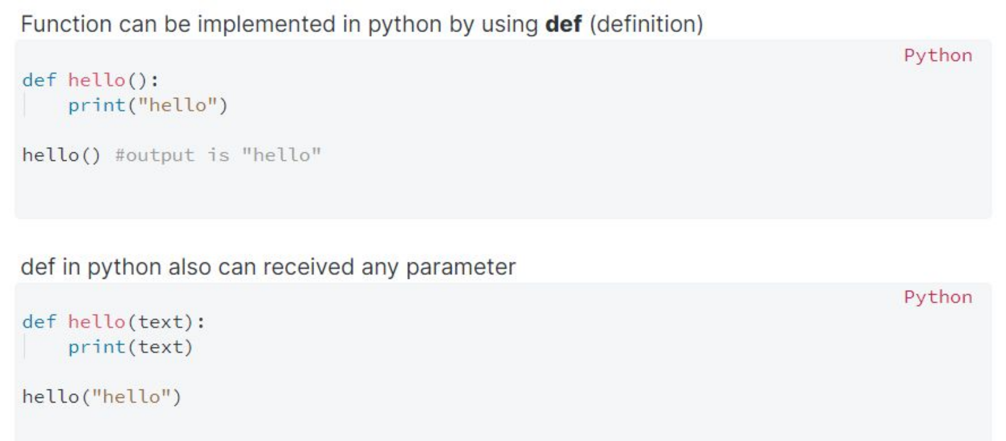
Function

Function is very important in programming and we can have abstract function or user-defined function

There got operation to get the value and this we called as function
eg: TodayDate(), MaxValue(), Min()

Some of the function already built-in python and we call it abstract function
eg: max(), min(), round()

Let’s say we have a problem which is 1+x=3
We can create a function called getX(1,3) which it will do 3-1 and return the value of the subtraction between both numbers which is 2



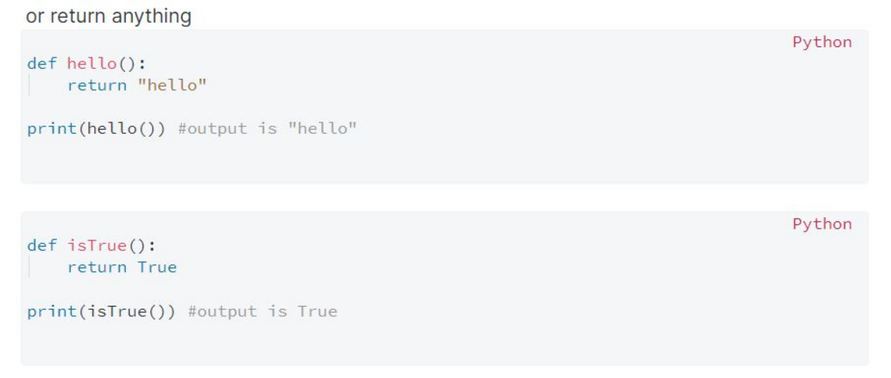
Function can be implemented in python by using **def** (definition)

```python
def hello():
    print("hello")

hello() #output is "hello"
```

def in python also can received any parameter

```python
def hello(text):
    print(text)

hello("hello")
```

Function



or return anything

Python

```python
def hello():
    return "hello"

print(hello()) #output is "hello"
```

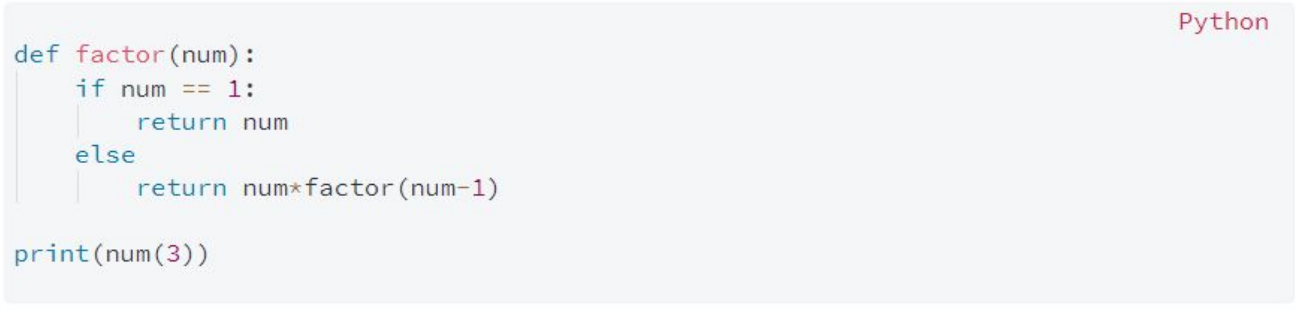
Python

```python
def isTrue():
    return True

print(isTrue()) #output is True
```

but what if we does not know the value but we know that the value is increase by summation or subtraction??

We will use the technique called RECURSIVE (call itself)!

Let's take a look in factorial concept.
We knew that factorial 3 is equal to 3x2x1=6 but how do it in python?



Python

```python
def factor(num):
    if num == 1:
        return num
    else
        return num*factor(num-1)

print(num(3))
```

03

Game Time

Game Time

Create a guessing game that generate random numbers from integer 1-50 and it ask the user to guess the number. Shows the appropriate message that shows the guessed number is larger than the answer or lower than the answer. The guess attempt can be only 5 times.

Use

```
import random

answer = random.randint(1,50)
```

Thank You

CLAIM YOUR CERTIFICATE

https://legoom.net/claim/

PYST3P30JULY