Introduction to Python Programming

Day7_HelloPython.md

recalling

LAST TIME TOPICS

Today’s Topics

- What is Programming language
- History of python
- why python is used in hacking
- How to install python
- what is IDE and code editor
- Creating python script file
- Outputs and comments
- Variables and Datatypes

What is Programming language

- It is language which helps to communicate with computers {they are not able to understand human languages}
- We humans have lots of languages(english,france,amharic,arabic..)
- Computers have many languages too
  - Assembly,C,C++,Java,Javascript,Python,Ruby,Perl,Go
- Prog lang. Helps us to write programs using those languages

What is Program?

- A program is an algorithm expressed in a programming language.
- An algorithm is a **detailed sequence of actions** to perform to accomplish some task. Named after an Iranian mathematician, Al-Khwarizmi.
- Technically, an algorithm must reach a result after a finite number of steps.
- With those steps Programs Do a Specific task Correctly.

Algorithm Example

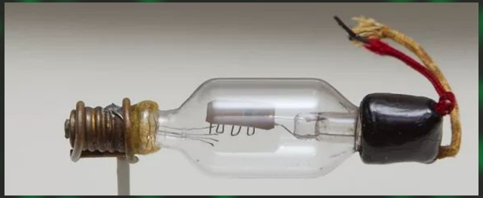
- To ask someone, his/her name, you will do the following step:
  a. You Walk to the person
  b. Greetings
  c. Wait for answer back
  d. “What is your name?”
- There will be a lot of kinds of algorithms to do a specific task

Evolution of I/O { Input / Output }

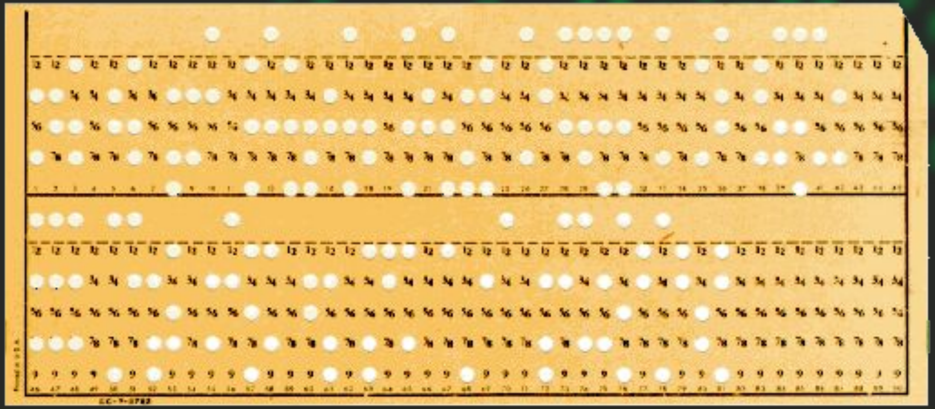
- Early in the **history of computing**, programs were submitted on punch cards with all the data they required and executed together with other programs that used the same libraries. Output was to a line printer.

- Later developments introduced interactive processing which allowed the user to **provide data while the program was running**. This normally takes place in a Question & Answer format.

Generation of Computers

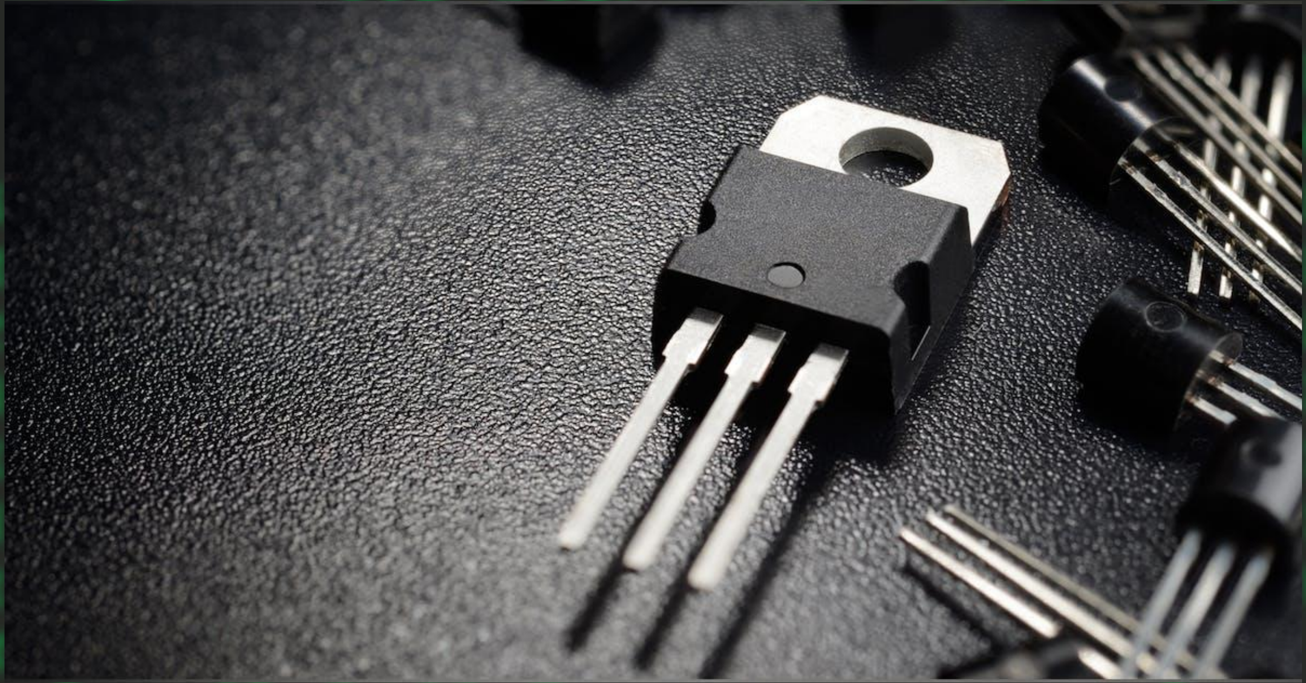


1. First Generation: Vacuum Tubes => punch cards
2. Second Generation: Transistors => Programming started here with Assembly
3. Third Generation: Integrated Circuits => **BASIC, COBOL, Pascal, Fortran, C, C++, Perl and Ada**
4. Fourth Generation: Microprocessors => Python, SQL,Matlab
5. Fifth Generation: Artificial Intelligence



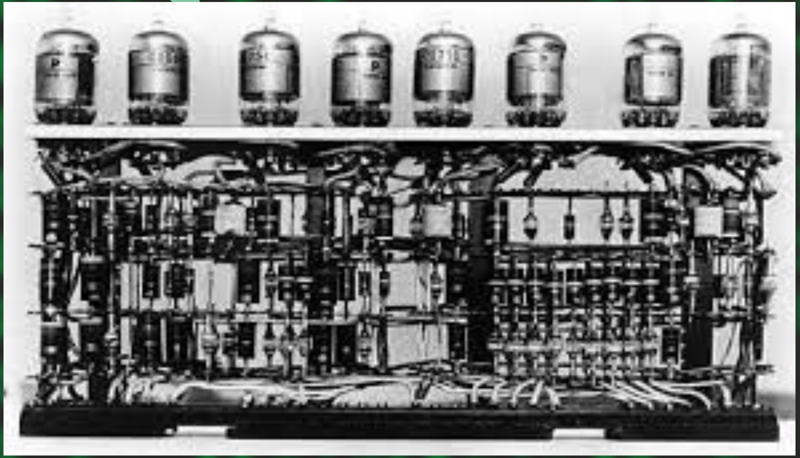
they could only solve one problem at a time. It would take days or even weeks to set up a new Program on First Generation.

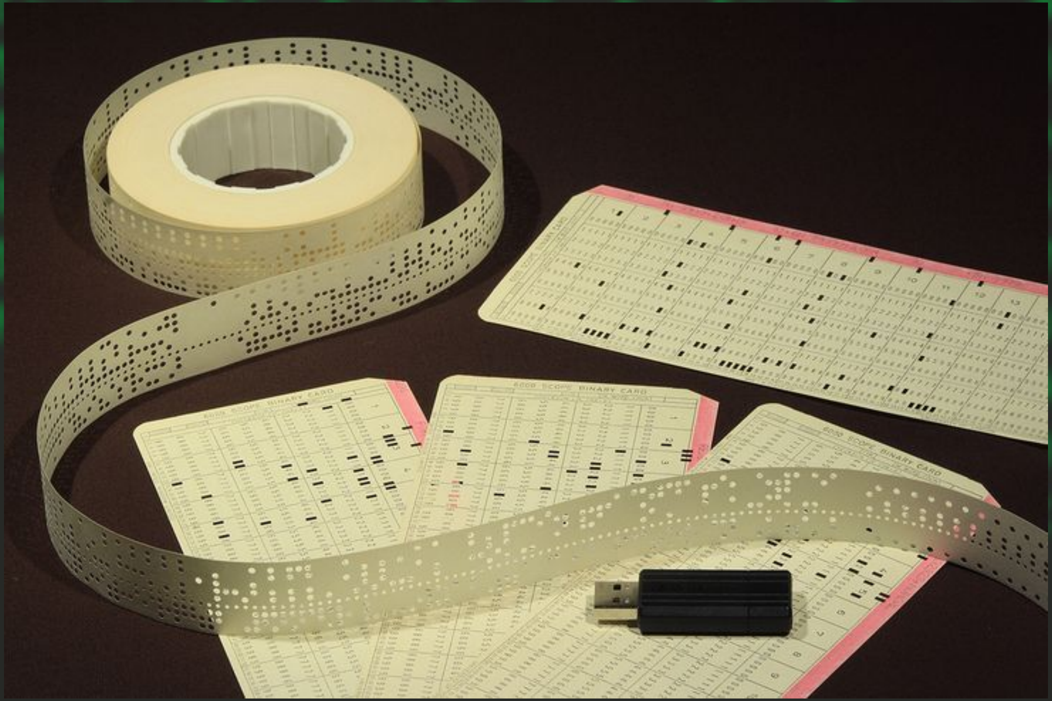
Transistors



Small electronic Device that changed the world.

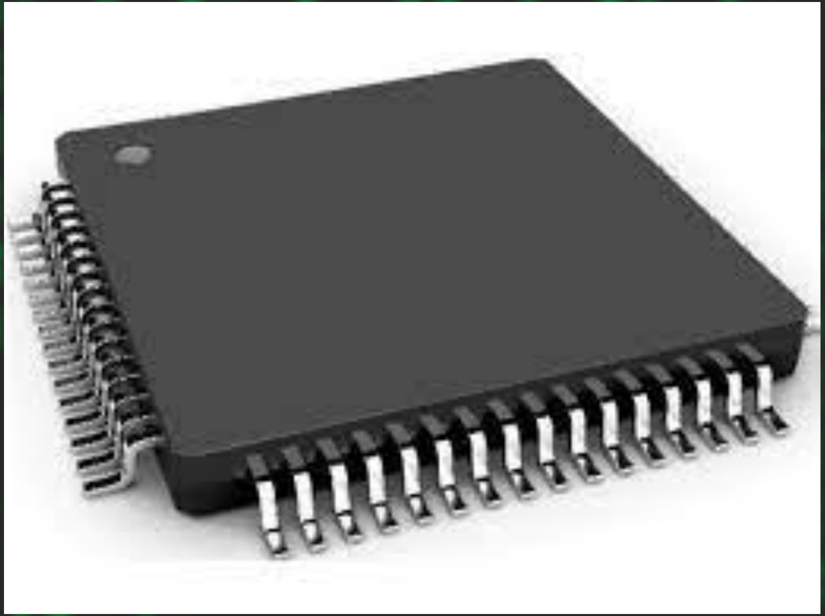
First generation computer





Integrated Circuit

Micro Controller



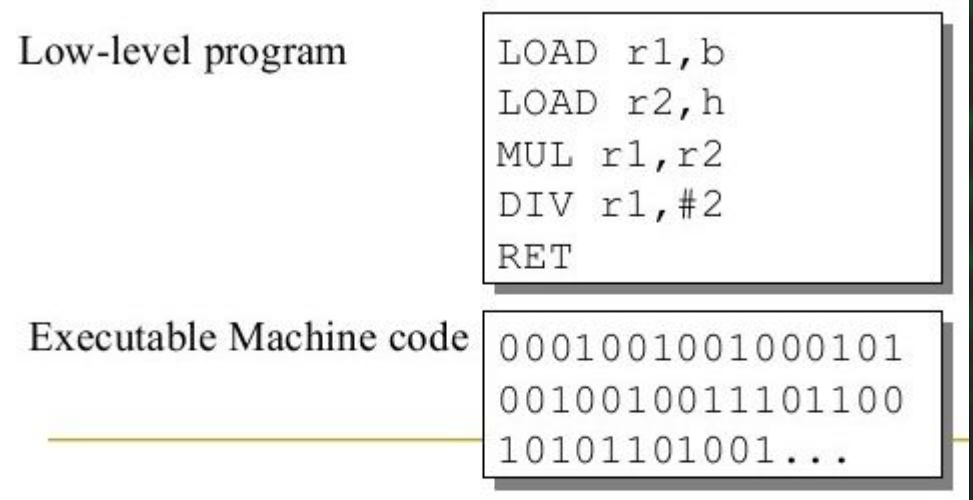
More like CPU

Types of Programming Languages

- Computers Understand binary(0/1) , humans don’t understand this
- SO based on the closeness of the language to humans we classify it into 3
- The more they become low to the machine they are faster.
- The more they become like human language they are slower.

A) LOW level programming language

- These languages are more like machines but with lots of effort people can understand them. They are close to the hardware of the computer.
- Ex: Assembly



Low-level program

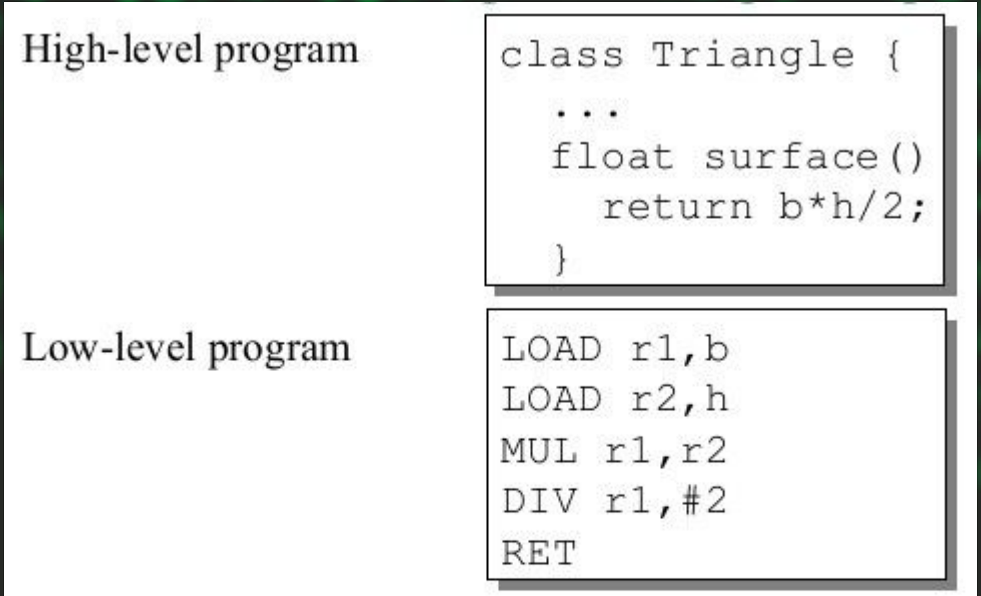
```
LOAD r1,b
LOAD r2,h
MUL r1,r2
DIV r1,#2
RET
```

Executable Machine code

```
0001001001000101
0010010011101100
10101101001...
```

B) High level Programming languages

- They are more close to human languages.
- Example: Python,C++,Java,JS…



High-level program

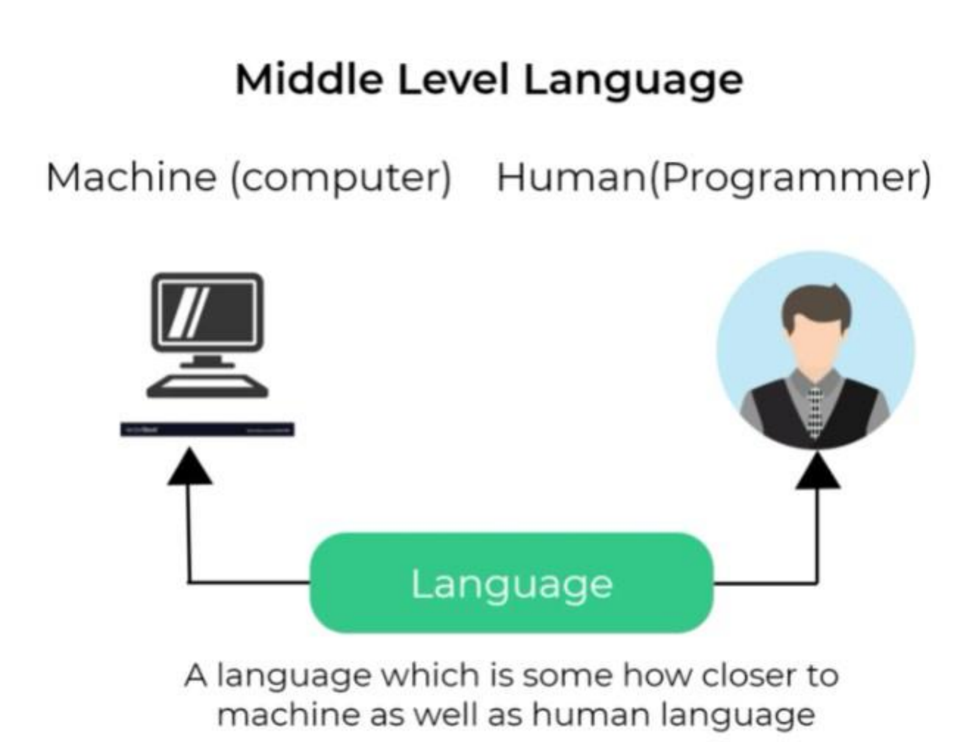
```
class Triangle {
  ...
  float surface()
    return b*h/2;
  }
```

Low-level program

```
LOAD r1,b
LOAD r2,h
MUL r1,r2
DIV r1,#2
RET
```

C) Medium Level

- Languages Between Low level and High level, they combine both
- Ex: C-lang



A language which is some how closer to machine as well as human language

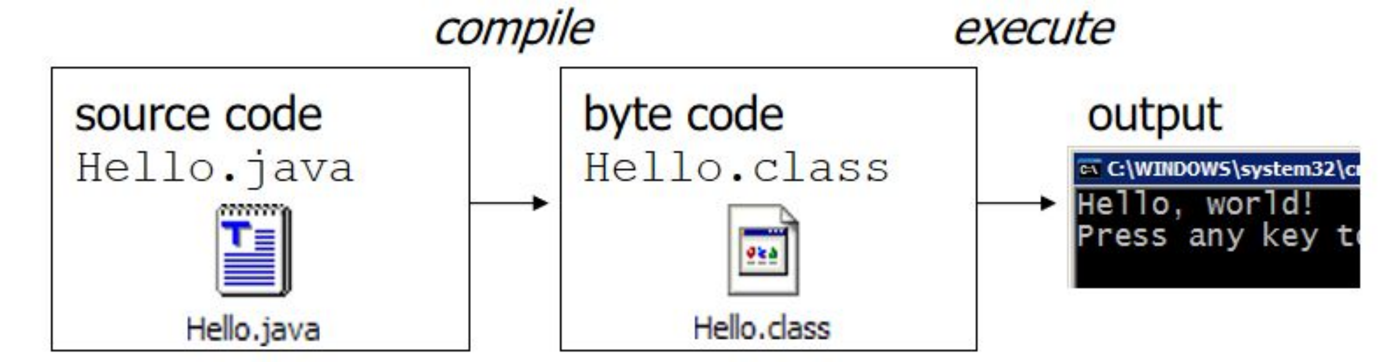
How do high level languages work?

As we saw earlier we have said that computers know only binarys, and if we code with high level languages how do computers understand us?

1) Compilers: are tools which helps to convert the whole code to bytecode then computer will execute it
   a) Example: C,C++,Java,..
2) Interpreter: can directly execute the code by reading the source code line by line
   a) Example: python

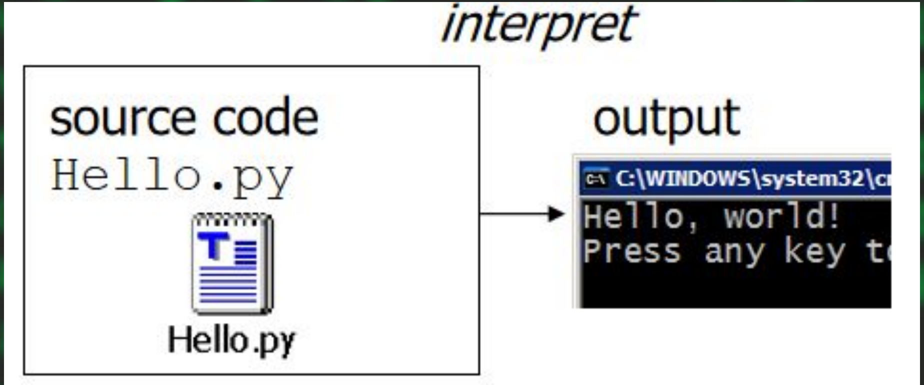
Cont...

A)



```
Hello, world!
Press any key t
```

B)



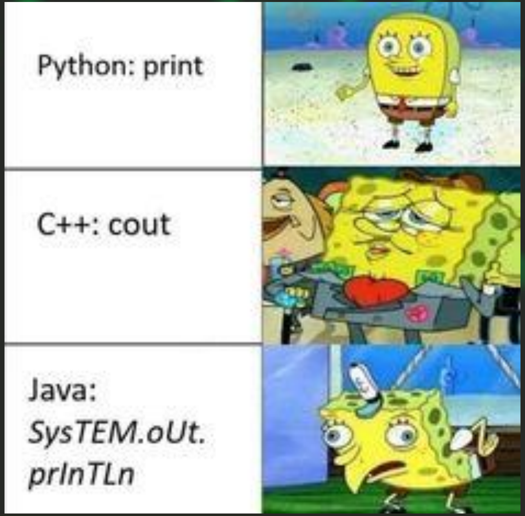
```
Hello, world!
Press any key t
```

Uses of Programming language

- Android Application Development
- Website Development
- Machine learning
- Artificial Intelligence
- Game Development
- Big data Technology
- Desktop software development
- Hacking tool development
- ….

What is Python Programming?

- Python is a High level & interpreted programming language. => Very easy to learn
- It is Very SImplified language any one can write with it, also can read it.



History of python

- Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.
- Python is **derived from many other languages**, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

- Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing it's progress.

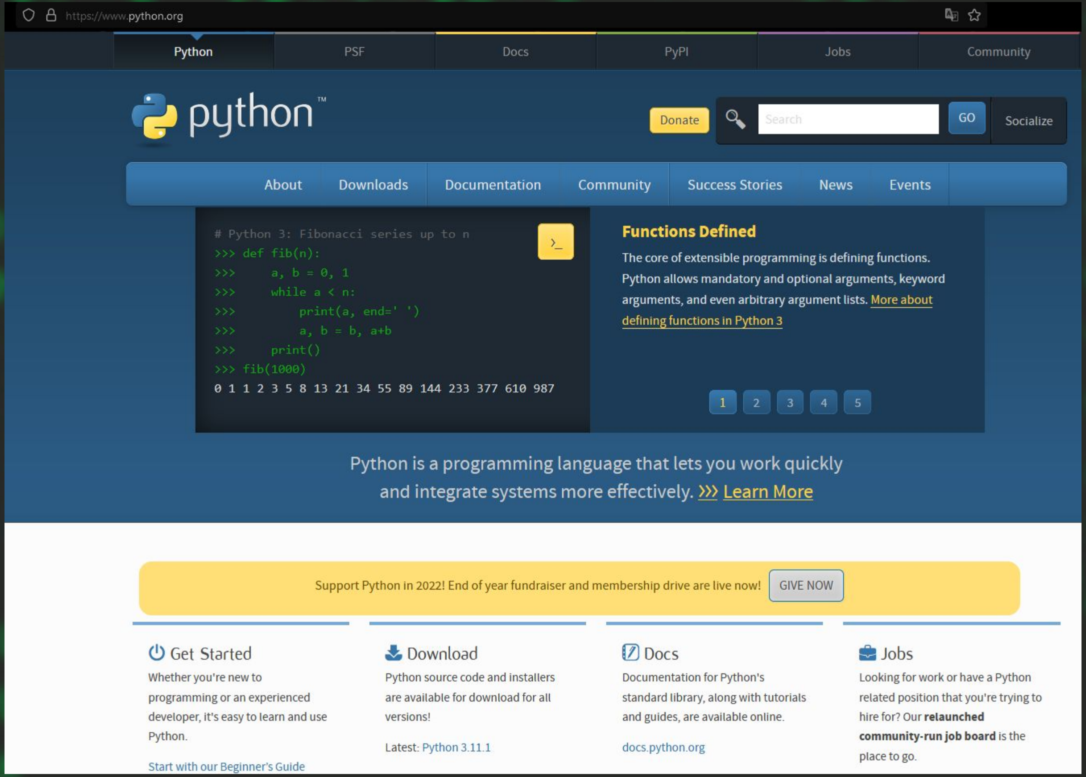
Uses of Python

- Data visualization
- Data analysis
- Machine learning
- Artificial intelligence
- Back-end web development (with frameworks like Django and Flask)
- Game development
- Hacking Script writing

Install python

- On windows you will download python from their website
- On linux it comes pre installed, else apt install python3

Website: https://www.python.org/



```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

Functions Defined

The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. More about defining functions in Python 3

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> Learn More

Support Python in 2022! End of year fundraiser and membership drive are live now!

Get Started

Whether you're new to programming or an experienced developer, it's easy to learn and use Python.

Start with our Beginner's Guide

Download

Python source code and installers are available for download for all versions!

Latest: Python 3.11.1

Docs

Documentation for Python's standard library, along with tutorials and guides, are available online.

docs.python.org

Jobs

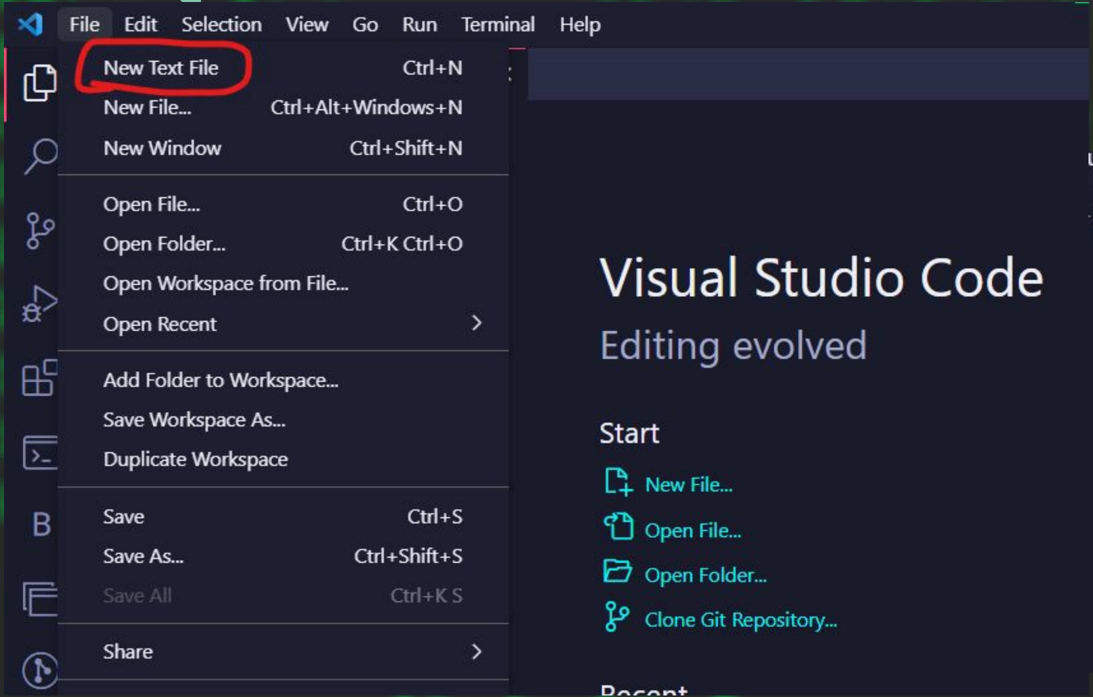
Looking for work or have a Python related position that you're trying to hire for? Our **relaunched community-run job board** is the place to go.

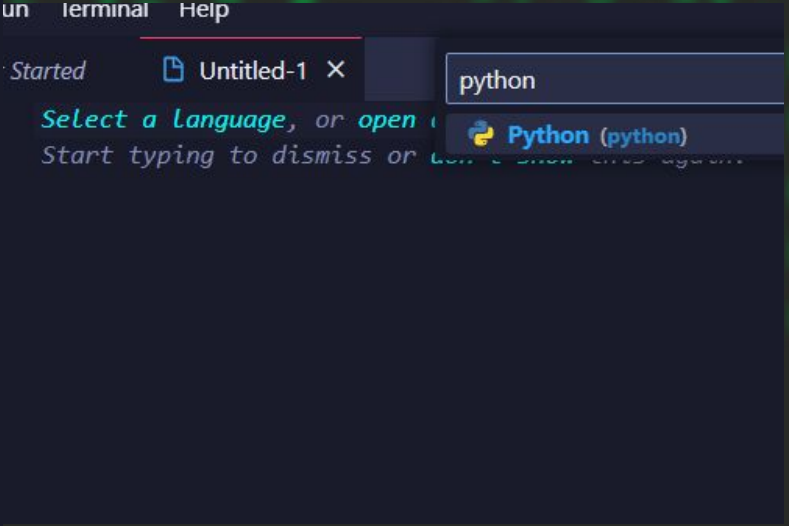
IDE & Code editors

- **IDE (Integrated Development Enviroment)**: Is a Software that helps to write & run a Specific Programming language. Example: PythonIDE
- **Code Editors**: are softwares those can help to write any kind of programming languages. And also by adding some compiling/ interpreting feature they can run programs/scripts Example: Sublime,Vscode

How to use Vscode for linux

After Installing python we have to configure somethings on VScode to run



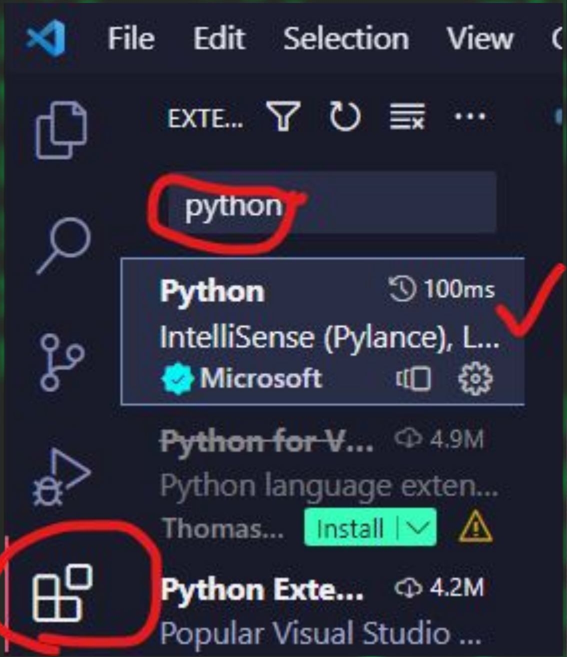


python

Select a language, or open

Start typing to dismiss or

Cont...

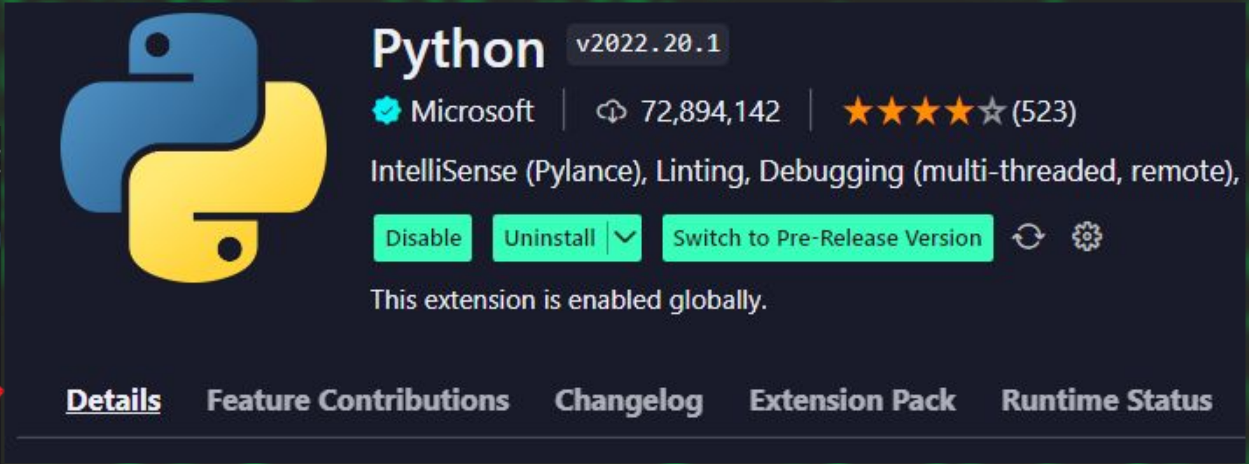


python

Python 100ms
IntelliSense (Pylance), L...
Microsoft

~~Python for V...~~ 4.9M
Python language exten...
Thomas... Install

Python Exte... 4.2M
Popular Visual Studio ...

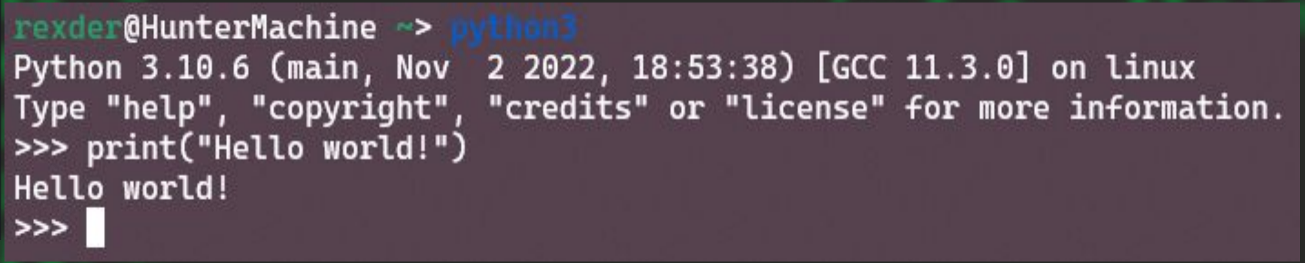


Python v2022.20.1

Microsoft | 72,894,142 | ★★★★☆ (523)

IntelliSense (Pylance), Linting, Debugging (multi-threaded, remote),

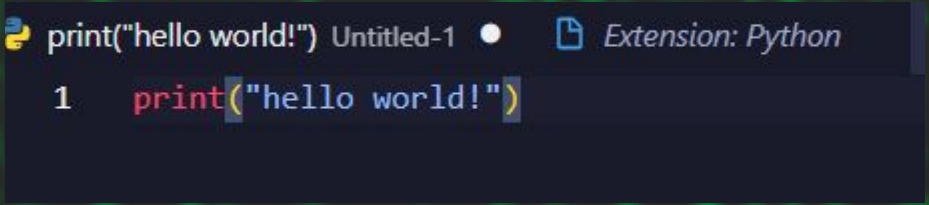
This extension is enabled globally.



```
rexder@HunterMachine ~> python3
Python 3.10.6 (main, Nov  2 2022, 18:53:38) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello world!")
Hello world!
>>> 
```

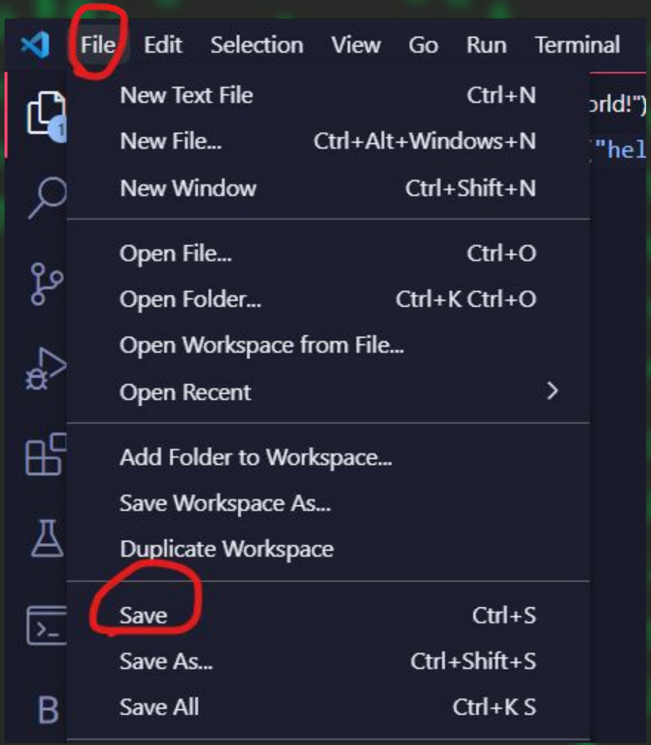
Testing

1

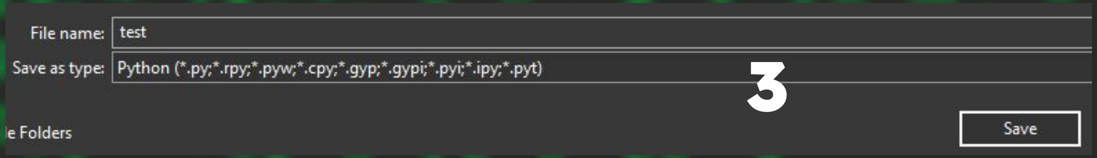


```
print("hello world!")
```

2

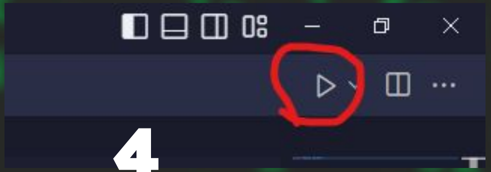


3

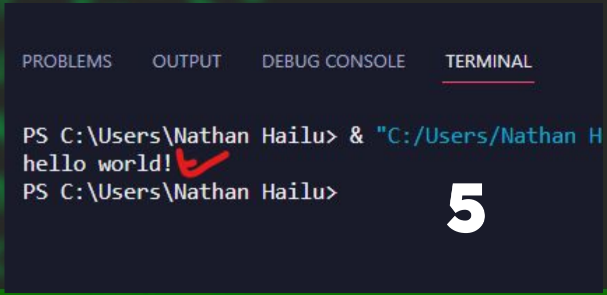

File name: test

Save as type: Python (*.py;*.rpy;*.pyw;*.cpy;*.gyp;*.gypi;*.pyi;*.ipy;*.pyt)

4



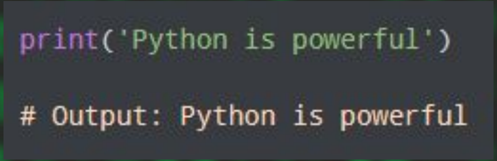
5



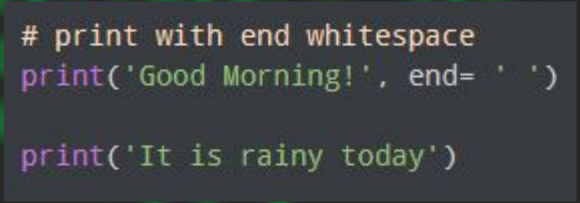
```
PS C:\Users\Nathan Hailu> & "C:/Users/Nathan H
hello world!
PS C:\Users\Nathan Hailu>
```

Outputs and COmments

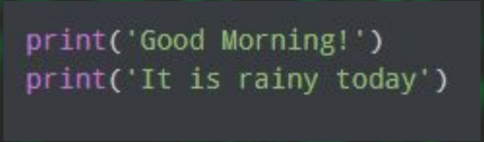
- On python, to display output we use keyword ‘print’
- Syntax: `print(object=’’, sep=’’, end=’’)`



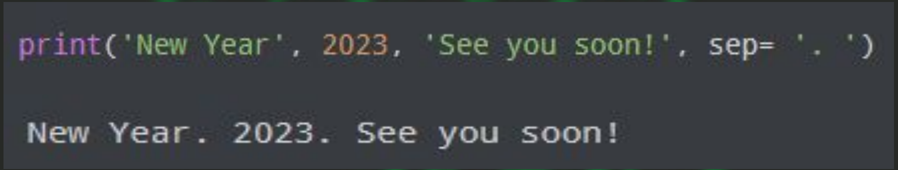
```
print('Python is powerful')

# Output: Python is powerful
```



```
# print with end whitespace
print('Good Morning!', end= ' ')

print('It is rainy today')
```



```
print('Good Morning!')
print('It is rainy today')
```



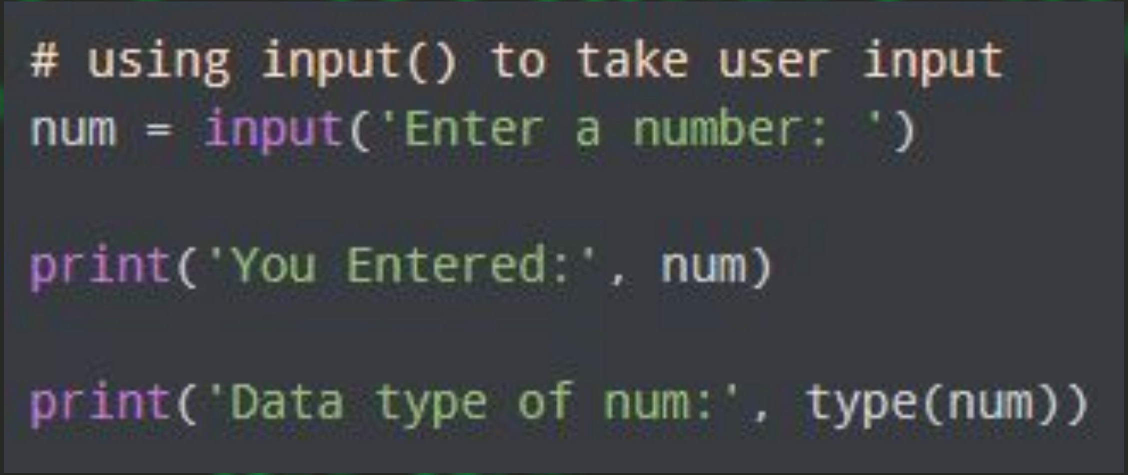
```
print('New Year', 2023, 'See you soon!', sep= '. ')


 New Year. 2023. See you soon!
```

\n - new line
\t - tab space

`print(text1,text2,text3…)`

Comments

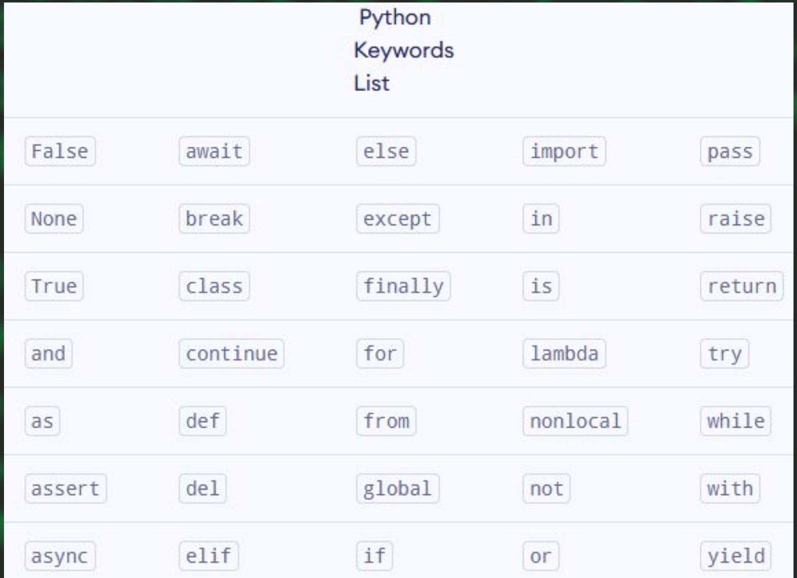
- This are a simple notes written on our codes those can help as to remember the function of the code or to make it simple for peoples to understand our code.
- Comments won’t be executed.
- Syntax: # This is a comment line



```
# using input() to take user input
num = input('Enter a number: ')

print('You Entered:', num)

print('Data type of num:', type(num))
```

Python Keywords

Keywords are predefined, reserved words used in Python programming that have special meanings to the compiler.



Python Keywords List

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield

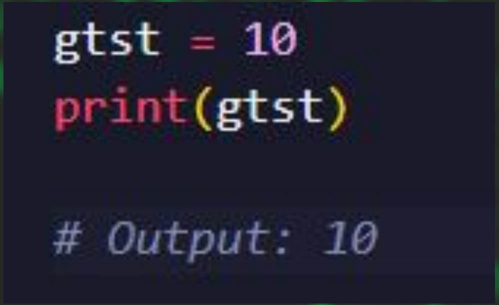
BreakTime

10 MIN

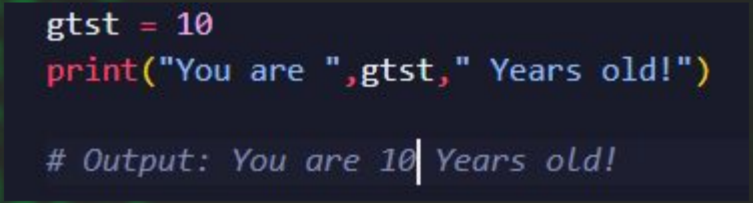
1) Write a code that displays “Welcome GTST to Python”
2) Write a code that displays
   a) Programming is Fun!
   b) Hackers can code.
3) Write a code that displays “Welcome GTST” & “to Python” those are separated by comma.

Variables

- Variables are a value holders /containers/
- They store data
- We give some value to some word.
- example : number = 10 => from now on my python program knows the value of number is 10.
- The process of giving value to word is called Variable Declaration
- The word that holds the data is called Identifier
- We can Print value of variables by giving the identifier



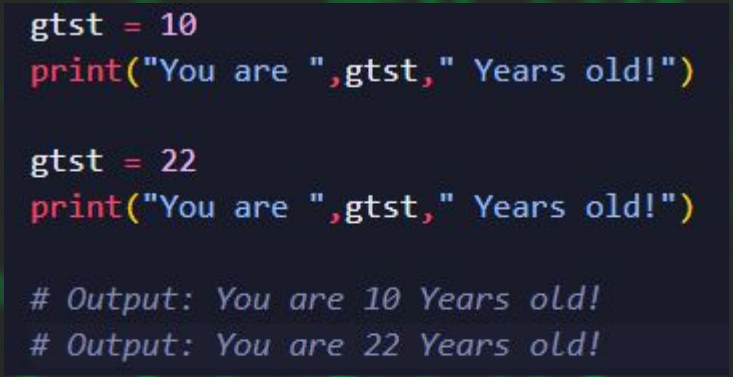
```
gtst = 10
print(gtst)

# Output: 10
```



```
gtst = 10
print("You are ",gtst," Years old!")

# Output: You are 10 Years old!
```

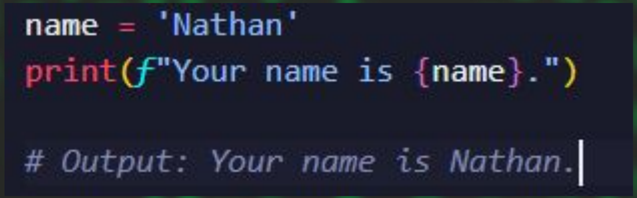
Cont...

We can change value of variable in a code.



```
gtst = 10
print("You are ",gtst," Years old!")

gtst = 22
print("You are ",gtst," Years old!")

# Output: You are 10 Years old!
# Output: You are 22 Years old!
```

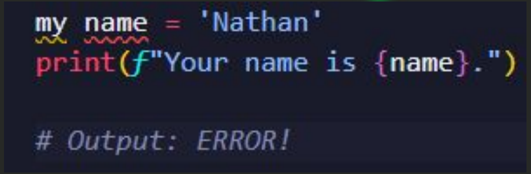
- You can print the variable with {variableName} on print keyword
- Syntax: `print(f”yourtext {variable}”)`



```
name = 'Nathan'
print(f"Your name is {name}.")

# Output: Your name is Nathan.
```

Remember!
On naming the identifier:

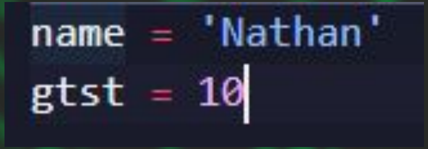
a) Dont use space between words use _
b) Dont use numbers as identifier



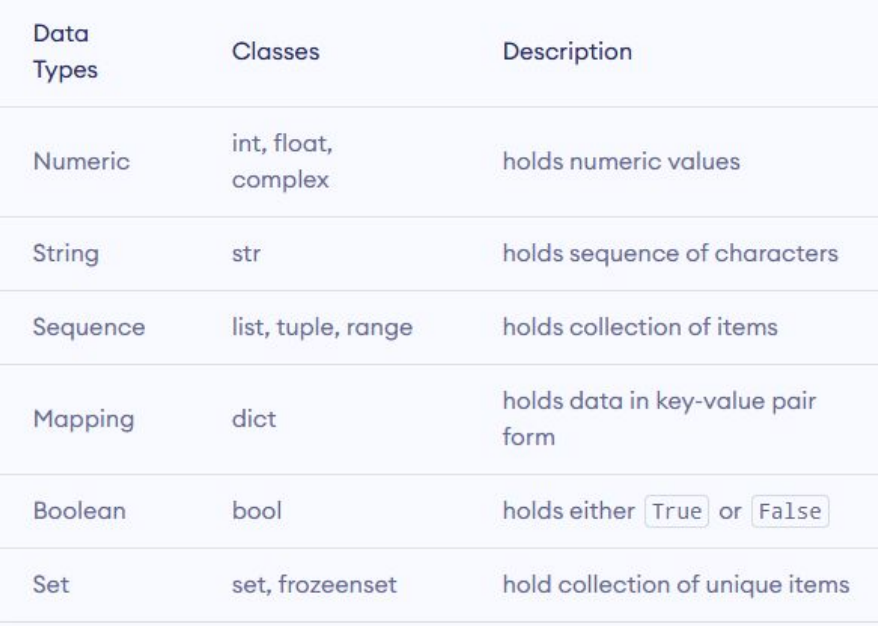
```
my name = 'Nathan'
print(f"Your name is {name}.")

# Output: ERROR!
```

Data types

- Have you seen the data types we used on the variable slide?
- There are a lot of Data types on python



```
name = 'Nathan'
gtst = 10
```

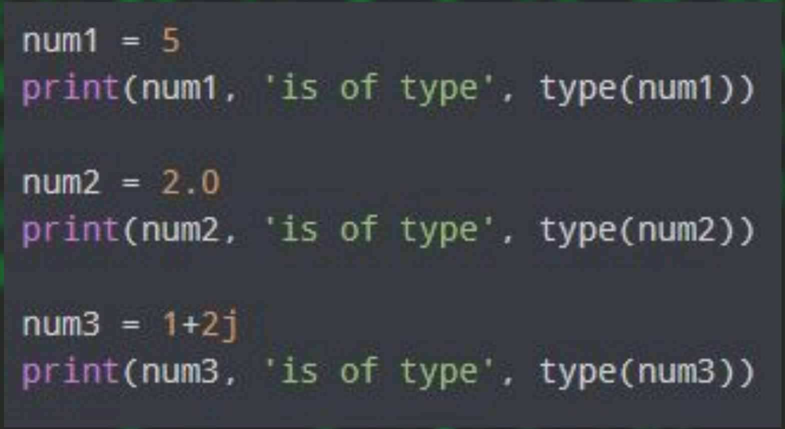


Data Types	Classes	Description
Numeric	int, float, complex	holds numeric values
String	str	holds sequence of characters
Sequence	list, tuple, range	holds collection of items
Mapping	dict	holds data in key-value pair form
Boolean	bool	holds either `True` or `False`
Set	set, frozeenset	hold collection of unique items

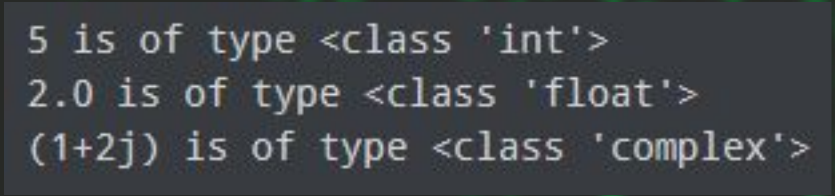
A) Numeric Data type

- int(integer) - holds signed integers of non-limited length.
- float - holds floating decimal points and it's accurate up to **15** decimal places.
- complex - holds complex numbers.

★ You can Identify The type of a variable with the keyword ‘type()’



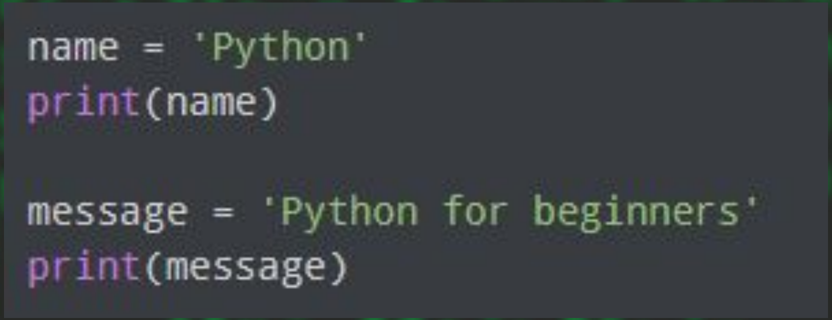
```python
num1 = 5
print(num1, 'is of type', type(num1))

num2 = 2.0
print(num2, 'is of type', type(num2))

num3 = 1+2j
print(num3, 'is of type', type(num3))
```



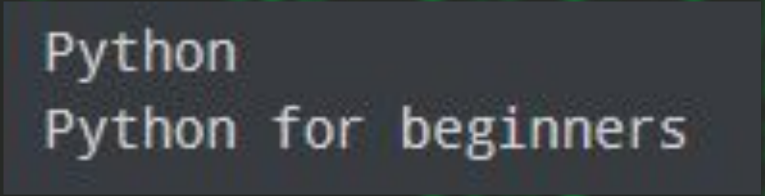
```
5 is of type <class 'int'>
2.0 is of type <class 'float'>
(1+2j) is of type <class 'complex'>
```

B) String Data

- String is a sequence of characters represented by either single or double quotes. For example, var = “ ” or var = ‘ ’



```python
name = 'Python'
print(name)

message = 'Python for beginners'
print(message)
```



```
Python
Python for beginners
```

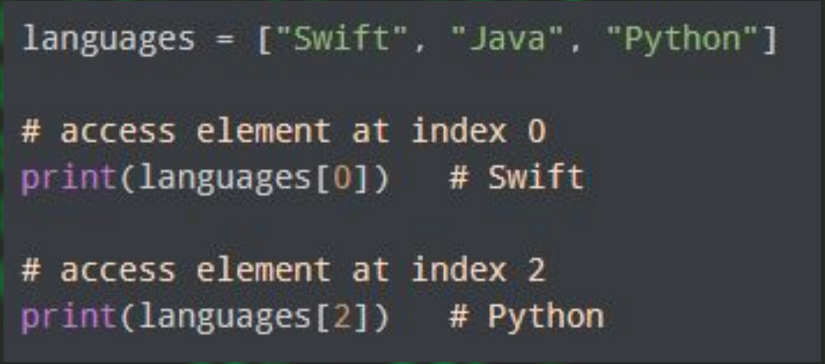
C) Sequence Data

A) Lists

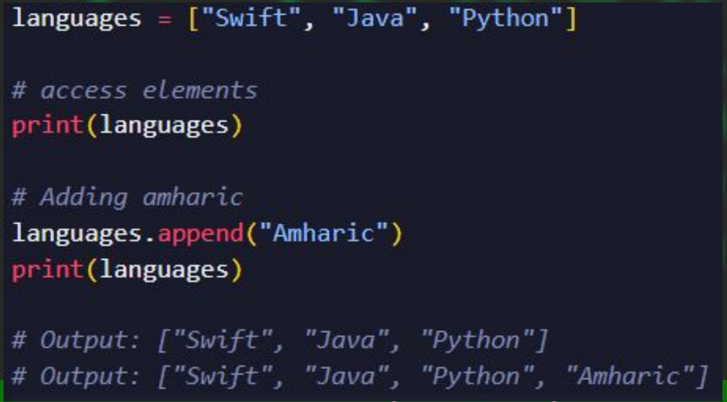
a) List is an ordered collection of similar or different types of items separated by commas and enclosed within brackets [ ]. For example, `languages = ["Swift", "Java", "Python"]`

b) To access items from a list, we use the index number **(0, 1, 2 ...)**. For example, languages[0]

c) We can add/modify objects to the list, `languages.append(“Amharic”)`



```
languages = ["Swift", "Java", "Python"]

# access element at index 0
print(languages[0])   # Swift

# access element at index 2
print(languages[2])   # Python
```

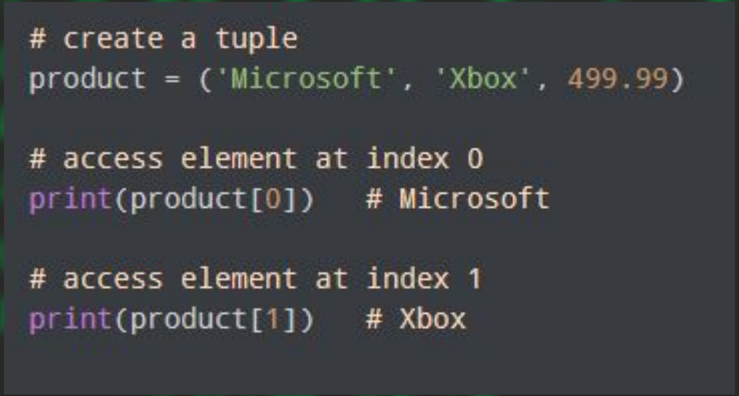


```
languages = ["Swift", "Java", "Python"]

# access elements
print(languages)

# Adding amharic
languages.append("Amharic")
print(languages)

# Output: ["Swift", "Java", "Python"]
# Output: ["Swift", "Java", "Python", "Amharic"]
```

Cont...

2) Tuple

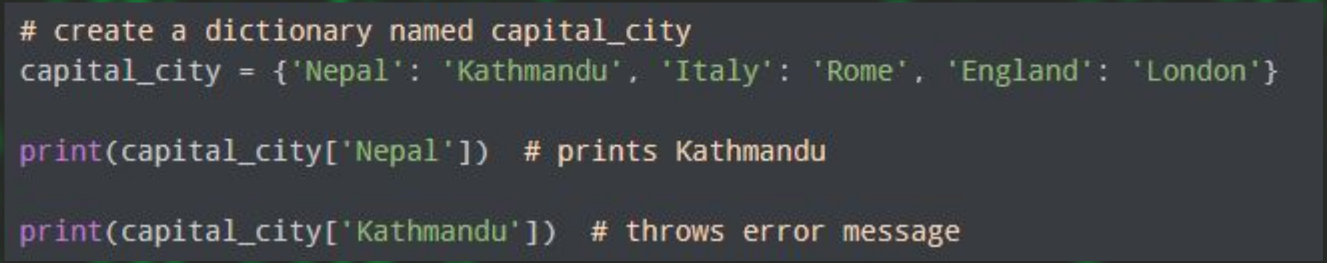
- Tuple is an ordered sequence of items same as a list. The only difference is that tuples are immutable. Tuples once created **cannot be modified**.
- we use the parentheses () to store items of a tuple. For example, `product = ('Xbox', 499.99)`
- Similar to lists, we use the index number to access tuple items in Python



```
# create a tuple
product = ('Microsoft', 'Xbox', 499.99)

# access element at index 0
print(product[0])   # Microsoft

# access element at index 1
print(product[1])   # Xbox
```

D) Dictionary data

- Python dictionary is an unordered collection of items. It stores elements in key/value pairs.
    - user1 = {'username':’nathan26,’password’:’p@$$word’}
        - username and password = key
        - nathan26 & p@$$word = value
- We use keys to retrieve the respective value. But not the other way around. For example,



```
# create a dictionary named capital_city
capital_city = {'Nepal': 'Kathmandu', 'Italy': 'Rome', 'England': 'London'}

print(capital_city['Nepal'])  # prints Kathmandu

print(capital_city['Kathmandu'])  # throws error message
```

Execise

20MIN

1) Create a variable called gtst with value “Day 7”
2) Display a text saying => “Hello Today is our Day 7 course” insert the “Day 7” from the variable gtst
3) Create a list of 0-5 even numbers, and print the list element with text
   a) => “The 1st Even number is: yournumber”
   b) => “The 2nd Even number is: yournumber”
   c) => “The 3rd Even number is: yournumber”
4) Create a dictionary called fruits with the following value
   a) apple = 10
   b) banana =15
   c) pineapple = 20
5) Create a variable called choice,the value will be users fruit choice,then display text saying “The value of fruitname is: value birr” using the dictionary on question 4,
   a) Example: if the value of choice is apple, the output is “The value of apple is: 10 birr”

CLASS IS OVER

1) Do your notes
2) Practice it well

Thank you!