

(1)

* Python : It is a widely used high level programming language for general-purpose programming and it was created by Guido van Rossum & was first released in 1991 at Netherlands.

This features a dynamic type system and automatic memory management and supports multiple programming paradigms i.e., pattern, including object-oriented, imperative, functional programming and procedural styles. It has a large and comprehensive standard library.

The two major version of python that are currently in active are:-

- ↳ Python 3.x is the current version and is under active development
- ↳ Python 2.x is the legacy version & will receive only secured updates until 2020. No new features will be implemented.

'R' language:

- Data Science
- Machine learning
- Artificial Intelligence
- Deep learning
- Statistics

} These all can be performed in 'R' language

②

'Python Language' :-

- Data Science
 - Machine learning
 - Artificial Intelligence
 - Deep learning
 - Statistics
 - Desktop application
 - web application
 - Database applications
 - Network applications
 - Game Development
- All these can be done using Python

* Mobile applications can be executed.

① Module 1 :- Python

- Basics
- Object Oriented Programming
- web application development

② Module - 2 : Statistics

- Descriptive Statistics.
- Probability along with Numpy / Pandas
- Inferent Calculation.

③ Module - 3 → EDA & Exploratory Data analysis.

- Numpy / Pandas
- Matplotlib
- plotly
- DS project

(3)

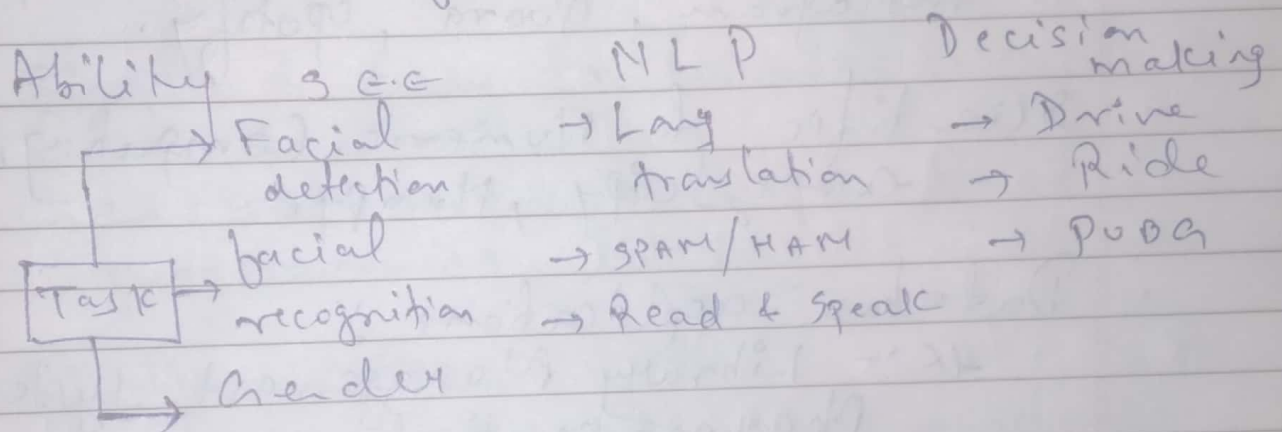
→ web Scrapping (Text-data entry & Images)

(4) Module 4 :- Machine learning

(5) Module 5 :- Deep learning

↳ Includes MLP & Computer vision

* ACI → Artificial General Intelligence



Programming in Python :-

* I/P → Print ("Hello")
O/P → Hello

* I/P → name = input("Enter your name:")
print("Hello {} , welcome!".format(name))

O/P → Enter your name : Akash Tripathi
Hello Akash Tripathi , welcome !

④

GitHub → similar to iDrive
↳ Used by programmers.

- Desktop application.
ex: Calculator, Notepad, etc
- Web application:
ex: - Youtube, Dropbox, Google, Instagram, Quora, Spotify.
- Scientific & Numeric Computing,
ex: - SciPy, NumPy.
- Database applications:
ex: - Library Management Systems, Pharmaceutical.
- Network applications,
ex: - Network sniffer, Network parameter abstraction, Routing Algo's like OSPF.
- Developing games:
ex: - Battlefield, Sims 4, PUBG.
- Data Science:-
ex: - Pandas, Matplotlib, Seaborn, ... etc.
- Machine learning:
ex: - SciKit.

(5)

- AI

ex: AIMA - AI: A modern approach

- POT:

ex: Raspberry Pi, Phillips (works great with low level code in C++)

Features of python:-

- (i) Simple & Easy to learn.
- (ii) Free & open source
- (iii) General purpose & high level programming language
- (iv) Platform Independent
ex: Mac, linux, windows
- (v) Case-sensitive (Upper & lower)
- (vi) Interpreted language
- (vii) Dynamically Typed
* no need to declare them as int for, float for anything.

* Rich library:-

* Writing the codes of AI, DS, ML, DL, Desktop & web applications

→ We can write console code using python.

Comments in python

⑥

Identifiers: Any variable name, function name, class name:-

* Rules of Identifiers:-

→ Allowed Characters:-

* Alphabets, Digits & Underscore symbol.

→ Identifier shouldn't start with a digit

→ Case sensitive

→ No length limit

→ Can't use reserved words for identifier

* I/P → `abc123 = 10.`

`ABC123 = 20`

`print(id(abc123))`

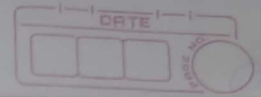
`print(id(ABC123))`

#id → address of the memory.

O/P → `14073220498480`
`14073220498800.`

* Reserved word in Python! (35)
(You can google it and get the list).

(7)



* Basic data - types in Python

* Numeric :- int, float, complex (Immutable)

• Boolean :- Bool (true/false)

• Strings: Immutable

• List → (Mutable, mostly used to store homogeneous data types)

• Tuple → (Immutable, faster as compared to list)

• Set → (Unordered collection of items, mutable & removes duplicates)

• Dictionary :- Unordered collection of key-value pairs, Mutable, ~~the~~ keys are unique - values may not be unique)

→ I/P → a = 2

b = 3.0

c = 5 + 7j

print(type(a))

print(type(b))

print(type(c))

O/P → int

float

Complex

Note :- In complex, any case (lower/upper) can be used & no error occurs

8

```
i/p → c = 10 + 20j
print(type(c))
print(c + 5)
print(isinstance(c, complex))
print(c.real)
print(c.imag)
```

```
o/p → <class 'complex'>
15 + 20j
True
10.0
20.0
```

```
* a = input("Enter 1st no.: ")
b = input("Enter 2nd no.: ")
print(type(a), type(b))
print(a + b)
```

```
o/p - enter 1st no.: 1
enter 2nd no.: 1
<class 'str', class str'>
11
```

Note :-

Number + Number = addition
String + String = joining together

(9)

I/P $\rightarrow a = 20$

$b = 2$

print (a ~~*~~ ~~*~~ b) \downarrow Power of something
i.e. a^b .

O/P $\rightarrow 400 //$

* Operators :-

\rightarrow Arithmetic : $+, -, *, \%, /, //, **$

\rightarrow Relational : $>, >=, <, <=$

\rightarrow Equality : $=, != \rightarrow$ not equal
 \downarrow comparison.

\rightarrow Logical : and, or, not.

\rightarrow Bitwise : $\&, !, ^, \sim, <<, >>$

\rightarrow Assignment : $:=, +=, -=, *=, /=, \dots$ etc

\rightarrow Ternary.

\rightarrow Identity : is, is not (used for address comparison)

\rightarrow membership : in, not in (for while)

$a = 20$ $b = 2$ print (a/b)	$a = 20$ $b = 2$ print (a//b)	$a = 20$ $b = 2$ print (a % b)
O/P $\rightarrow 10.0 //$	O/P $\rightarrow 10$	O/P $\rightarrow 2$ give remainder