

Python



J.V. Rassum (Guido van Rossum)

Python Foundation → 1991 Netherland

Serial - Monty Python Circus
(BBC Comedy Show)

KAWA → Gasoline

IDLE = Integrated Development Learning Environment

for -
writing,
testing,
Debugging

Python - & Dynamic typed language
 - programming language + scripting language
 - It is both compiled as well as
 an interpreted language,
 (first compiled and then interpreted)

A python is 5th generation programming language. It is written and developed by J.V Rassum in the year 1991 from Python foundation, Netherland. It is dynamically typed programming language.

NOTE : Python can be used to both procedural approach and object oriented approach.

Q. Where python stand?

Python is an interpreted object oriented high level programming language with dynamically typed and the name of the python that came from British ~~com~~ comedy show Monty python circus on BBC. Python is most popular language for web application, software development, Machine learning, Data science because we can create complex and multi-protocol operation.

The most popular application where built in python Instagram, youtube, and several social network.

Q. Why python is called python

The inspiration for the name came from BBC TV show Monty python circus or Monty python ~~flying~~ blind circus and GV Hassam was big fan of this TV show, due to this reason this language is known as Python.

NOTE : python is programming language or scripting language.

According to Linux Journal a python is programming as well as scripting language.

This language is have used for various purpose

- 1) software development
- 2) scripting for web application
- 3) software testing

IDLE (Integrated Development Learning Environment)

The working environment of python is known as IDLE because it integrated the several function package, module in common environment for editing, compiling, debugging is same environment \Rightarrow IDLE

Various IDLE available for python program

- 1) Pycharm
- 2) spider etc

* Features of Python

1) platform Independent : python coding is easily transfer from one system to another. There is no need to upgrade operating system. and also provide scalable database.

2) Database Scalable :

↓ with the help of API

API : Application programming Interface.

It will be support GUI tool for interactive pages with the help of several types of button and also establish the connection.

between any database either relational or Non-Relational.

(Sqlight, mysql, Sqlserver)

- 3) portable and compatible,
- 4) python has huge library and plenty of inbuilt function for developing simple and complex operation.
- 5) GUI supported programming environment.

* How to write python program

We can write python program into two ways:

- 1) & command line interpreter
- 2) By IDLE

| interpreter
| compile

First of all we can install python Jupiter / Anaconda and Pip installer.

Download - there is no need of licence
as it is open source.

Step I : Open python IDLE

Step II : Click on File menu and choose New file

Step III : New window / screen will be appear

Step IV : Hence type the python program and saved by .py extension.

Step V : To run python program

Choose Run Menu → click on Run Module.

LISP - Programming (Lab)

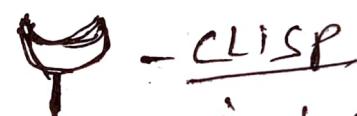
1ST of all download lisp Compiler \Rightarrow

clisp \Rightarrow full

clisp \Rightarrow install



Extract-



- CLISP

installed
successfully.

How to write program in Lisp

(write " ")

with the help of this function we can display the output and also pass the message in Lisp programming.

(write-line " ")

User entry: read() : with the

help of read() we can accept some input from the keyboard

setq () : with the help of this function
we can set the value of the variable

Program :

[.lsp] Save by
[.lisp] Lisp program.

WAP program in Lisp, to display
message Hello Ranchi

Command prompt -

Rules of Lisp programming

>

If we can write a lisp program
in any a text editor or Note pad /
VSCode)

then lisp program will be saved
.lisp or .lsp extension.

In Lisp programming we can write
each and every program inside
> the parenthesis i.e. we can developed

each and every program in Lisp
inside the function and function
followed by parenthesis () .

we can write program or expression ~~in~~
in prefix notation order

> (write "Hello Ranchi") o/p \Rightarrow Hello Ranchi

wAP to find Multiplication of two
number using command prompt

> (write (* 5 3))

> (write-line (+ 5 5))

o/p \Rightarrow 10

must maintain
the space after
otherwise it will give
error.

wAP to input and number and add
them

> (write (+ 5 (read)))

7

o/p \Rightarrow 12

1. Display Hello world.

> (write "Hello World") o/p :- "Hello World"

2. Add two Numbers

> (write (+ 23 7)) o/p :- 30

3. Multiply , 5 with inputting a number,

> (write (* 5 (read)))
2 o/p :- 10

4. Display using hello using write-line.

> (write-line "hello") o/p :- hello

5. Write a program to enter any two number and find their sum using function.

> (defun sum ()
(terpri)
(princ "Enter 1st Number =: ")
(setq num1 (read)))

(princ "Enter 2nd Number =: ")
(setq num2 (read)))

(setq s (+ num1 num2))

(print "sum of two no =: ")

(write s))

SUM

> (SUM)

Enter 1st number =: 82

Enter 2nd number =: 7

"Sum of two no =: 89
89

6. WAP in Lisp Program to input the radius and find Area of circle using function

```
> (defun Area()
  (terpri)
  (princ "Enter radius = ")
  (setq rad (read))
  (setq ar (* 3 rad rad))
  (print "Area is := ")
  (write ar))
AREA
```

```
> (Area)
```

Enter Radius = : 10

" Area is := " 300

300

→ write vs writeln

[In Python]

- write is used to output data without adding a newline character, while writeln adds a new line character after the output.
- Both write() and writeln() only work with strings.
If you're writing numbers or any other type, you need to convert them to strings using str().

→ write vs writeln

[in List]

write writes one object (like a number or a string) whereas writeln writes a string, and adds a new line.

7. write a program to enter base and height and find area of triangle.

```
> (defun triar()
  (terpri)
  (write-line "Area of Triangle")
  (write "Enter base = ")
  (setq b (read))
  (write "Enter height = ")
  (setq h (read))
  (princ "Area = ")
  (write (* 0.5 b h)))
```

TRIAR

O/P :-

> (triar)

Area of Triangle

"Enter base = " 2

"Enter height = " 7

Area = 7.0

7.0

8. Write a program input any two number and perform all arithmetic operation using clisp.

```
> (defun arith-opr()
  (terpri)
  (write-line "Arithmetic Operations")
  (write "Enter 1st Number = ")
  (setq num1 (read))
  (write "Enter 2nd Number = ")
  (setq num2 (read))
  (terpri)
  (princ "Addition = ")
  (princ (+ num1 num2))
  (princ "Subtraction = ")
  (princ (- num1 num2))
  (princ "Multiplication = ")
  (princ (* num1 num2))
  (princ "Division = ")
  (princ (/ num1 num2)))
```

O/P :-

> (arith-opr)

Arithmetic Operations

"Enter 1st Number = " 26

"Enter 2nd Number = " 3

"Addition = " 29

"Subtraction = " 23

"Multiplication = " 78

"Division = " 26/3

26/3

* OOPs in Python

OOPs :- It is a notation / symbol of class and object that communicate with each other with help of attribute (property) and method to solve a complete task.

In other word we can say that OOPs is methodology of class and object that interact with each others.

With the help of state and behaviour to perform a complete task.

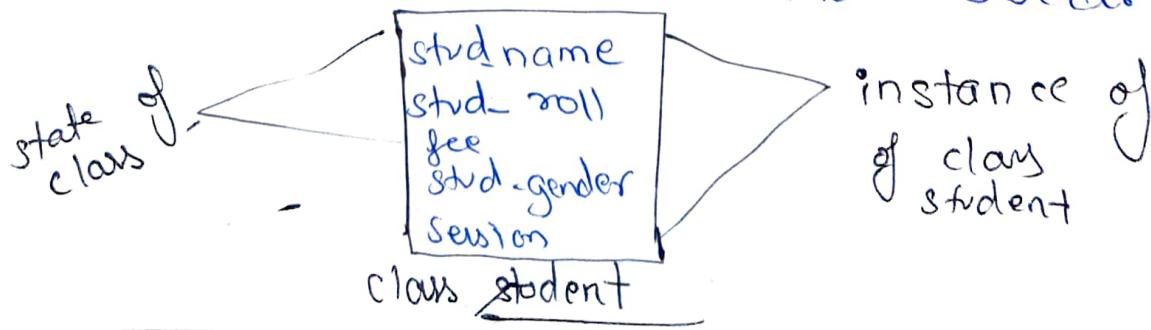
→ download - Python for any web browser
 → can run on - pycharm, spyder, Jupiter, Anaconda, vs code.

→ python is open-source and free

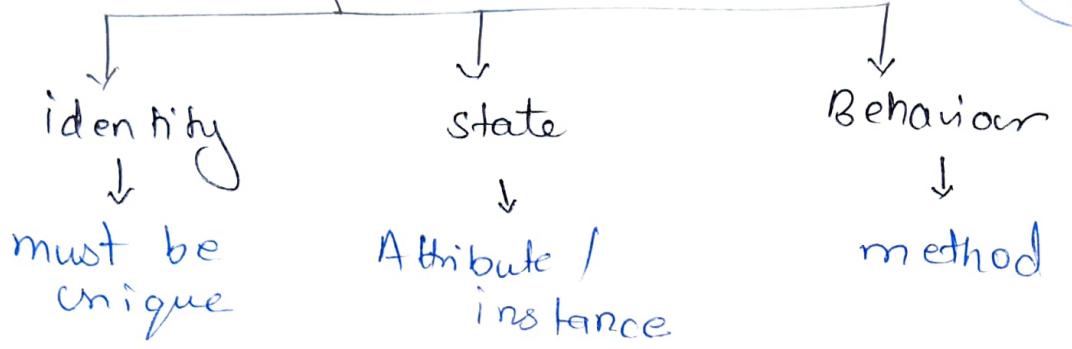
property = attribute = state
 method = behaviour.

Python is truly object oriented language that means All most things to things will be form of object and it will be implemented by property and method.

Class and Object :- Class is a template / blueprint of the object.



Object - A real-world entity.



e.g:- Dog

eye
color
leg
mouth

(appearance)

Barking

* Class : A class is collection of object that common identity, common behaviour with common state.

Syntax :

```
class classTagName:  
    def method-name(self):  
        body of method  
    ==  
    body of class
```

where, classTagName is valid python identifier.

def : def is keyword that is used for defining the method

self : self is a special parameter that allows to defined method.

s
Before swap

Enter 1ST No 5

Enter 2nd No 7

After swap

7 5

- strings and characters
 → case-sensitive
Hightly → Case-insensitive → variables, functions, symbols
 → print, princ, write, write-line
 all are used for output

Function	Purpose	Quote & Escapes not	Newlines	Use case
print	For debugging	yes	Before & after	Show data with structure
princ	for users (clean)	NO	None	Display clean output.
write	Flexible Output	configurable	None (default)	Format control (e.g. base, case)
write-line	String lines only	NO	After	Write strings line-by line

- All accept number except write-line, which needs a string
 → convert numbers for write-line using
 (write-line (princ -to- string 42))

Swap of two Number with 3rd variable.

```
>(defun swap()
  (terpri)
  (write-line "Before swap:=")
  (print "Enter 1st Number=")
  (setq num1 (read))
  (princ "Enter 2nd Number=")
  (setq num num2 (read))
  (terpri)
  (write-line "After swap:=")
  (setq num3 num1)
  (setq num1 num2)
  (setq num2 num3)
  (print "1st Number =")
  (write num1)
  (print "2nd Number =")
  (write num2))
```

SWAP

> (swap)

before swap:=

"Enter 1st number = " 45
Enter 2nd Number = 32

After swap:=

"1st number = 32
"2nd number = "45

Swap of two number without 3rd variable

```
> (defun swap()
  (terpri)
  (write-line "before swap:=")
  (print "Enter 1st number = ")
  (setq num1 (read))
  (print "Enter 2nd number = ")
  (setq num2 (read))
  (terpri)
  (write-line "after swap :=")
  (setq num1 (+ num1 num2))
  (setq num2 (- num1 num2))
  (setq num1 (- num1 num2))
  (print "1st number = ")
  (write num1)
  (print "2nd number = ")
  (write num2))
```

SWAP

```
> (swap)
```

before swap:=

" Enter 1st number = " 45
Enter 2nd number = 36

after swap:=

" 1st number= " 36
" 2nd number= " 45

WAP to input name, roll and 3 subject marks
and find sum and average in clisp programming
in AI

```
> (defun data ()  
  (terpri)  
  (write "Enter your name = ")  
  (setq n (read))  
  (write "Enter your roll = ")  
  (setq r (read))  
  (print "enter marks of three subjects : ")  
  (print "enter marks of python = ")  
  (setq p (read))  
  (write "enter marks of java = ")  
  (setq j (read))  
  (write "enter marks of c++ = ")  
  (setq c (read))  
  (terpri)  
  (setq sum (+ p j c))  
  (print "sum = ")  
  (write sum)  
  (print "avg = ")  
  (write (/ sum 3)))
```

DATA

```
> (data)
```

"Enter your name = " vibha

"Enter your roll = " 76

"enter marks of three subjects : "

"enter marks of python = 98

"enter marks of java = " 82
"enter marks of c++ = " 93

"sum = " 273
"avg = " 91
91

Python = 96
java = 86
C++ = 86

sum = 268
avg = 268 / 3

In another word we can say that, It is a special argument that refers to the instance of the class and allow to initiate the method and attribute.

self parameter passed automatically on a 1st argument when method are called.

* Creation of Object

object-name = class-name()

With the help of object-name and dot operator we can invoke/called the attribute and method for the purpose of execution.

Example

class FACULTY :

id
name
Des
Salary

FACULTY

def put-data(self):

 self.id = int(input("Enter Faculty id ="))
 self.name = input("Enter Faculty name =")
 self.des = input("Enter your Designation =")
 self.sal = int(input("Enter Yr Salary ="))

def display-data(self):

 print("The faculty Name id is =", self.id)
 print("The faculty name is =", self.name)
 print("The faculty designation is =", self.des)
 print("The faculty Salary is =", self.sal)

my-faculty = FACULTY()

my-faculty.putdata()

my-faculty.display-data()

EMACS → IDLE

12/04/28

* Serialization (using constructor)

instance variable create
& initialize

● constructor : `--init--()`

A constructor is special method of the class that will be automatically invoked when instance variable or attribute is created.

The main task of constructor is to create and initialize the object of the class.

NOTE : A constructor is a special method typically named `--init--()` that is automatically called when attribute of the class is created and initialize.

- constructor name and class name always be different.
- constructor name always start with `--init--()`
- init → initialize the python object.

Example :-

→ A function has two section

- 1) Definition section
- 2) Calling section

NOTE : If class have constructor the 1st of all do the work of constructor than any other task.

class PERSON :

def __init__(self, name, id, gender, age, city):

self.name = name

self.id = id

self.gender = gender

self.age = age

self.city = city

def intro_disp(self):

print("Person name is = " + self.name)

to use multiple string and nos together
pass multiple argument together

person id is = " + self.id + " Gender is = " + self.gender + " Age is = " + self.age + " and city is = " + self.city + ")")

my_person = PERSON ("Rakesh", 1234, "Male", 21, "Ranchi")

my_person1 = PERSON ("PRIYA", 234, "Female", 22, "Patna")

my_person.intro_disp()

my_person1.intro_disp()

f = Taken the value as argument.

• Serialization object

ENCODING,

It is also known as MARSHLING or PICKLING.
 It is a process of conversion of Python supported object into file supported object or Network supported object. This phenomena is known as Pickling or serialization.

• Deserialization :

It is also known as DECODING, UNMARSHLING, or UNPICKLING.

It is a process of converting file supported object or Network supporting object into python supporting object is known as Unpickling or Unmarshling.

JSON - Java script object notation

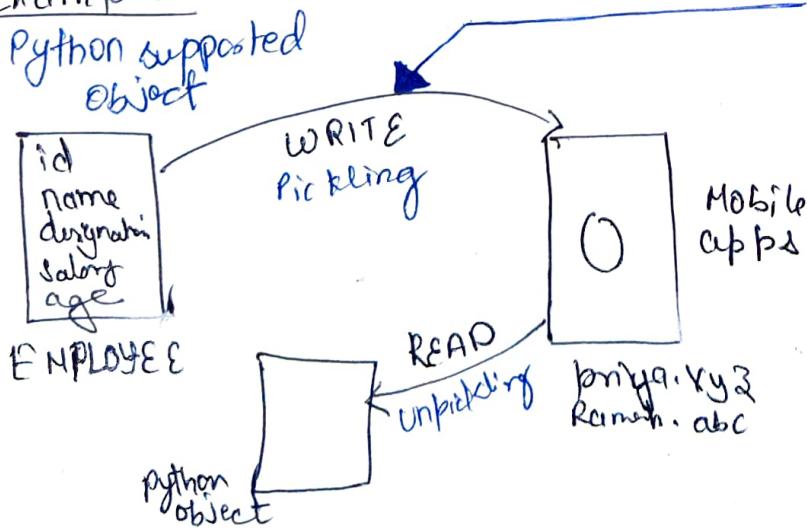
YAML - yet Another Markup language

Network/
file supported
object

Python $\xrightarrow{\text{generation}}$ JSON /
YAML

YAML $\xrightarrow{\text{deserialisation}}$ Python
JSON $\xrightarrow{\text{deserialisation}}$ Python

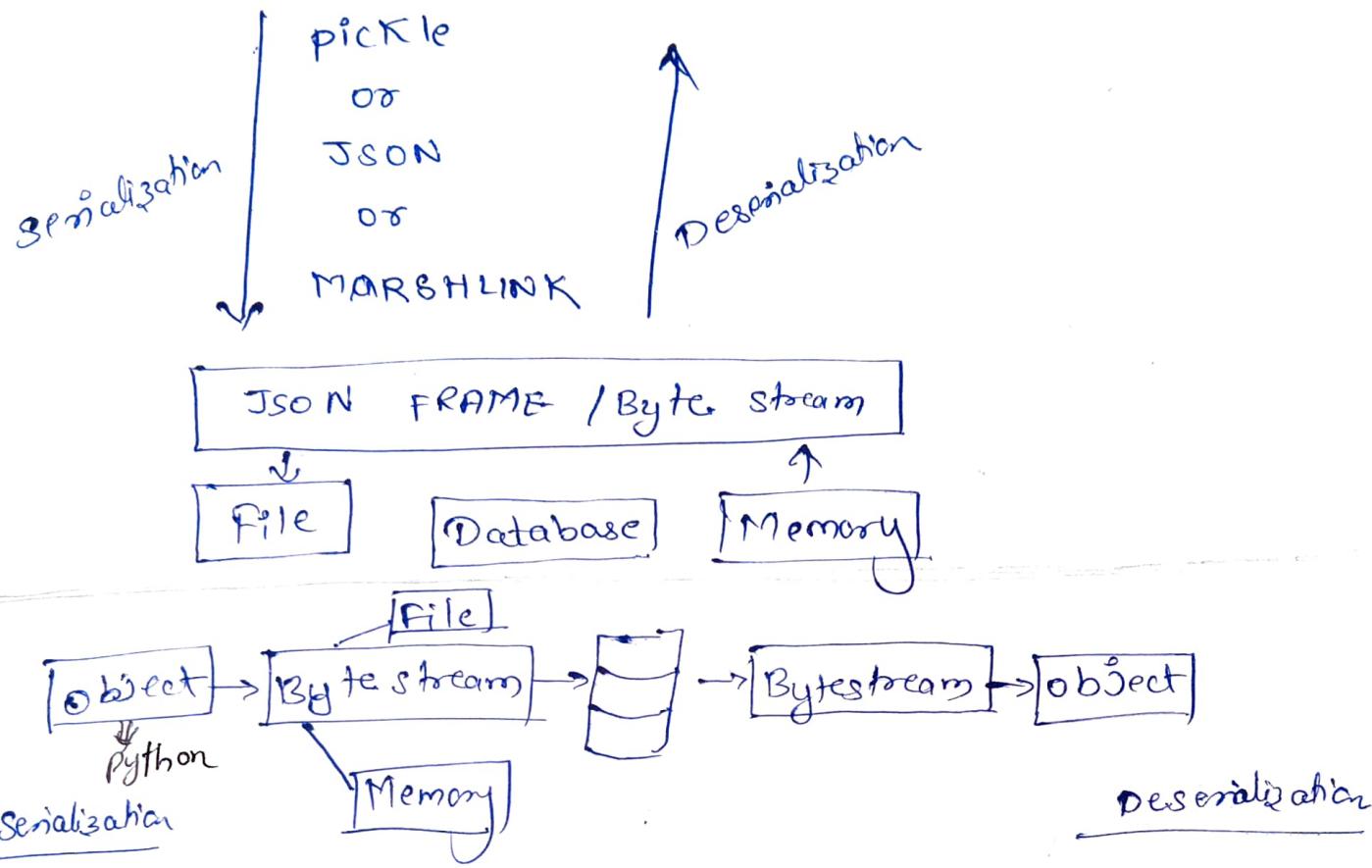
Example



This object will be save in file supported object / Cloud(Network, supported object)

English
 \downarrow
serialisation
 \downarrow
Byte
(e.g. English)

* Python serialization



pickle: It is a method to convert python object into ByteStream (Not human readable)

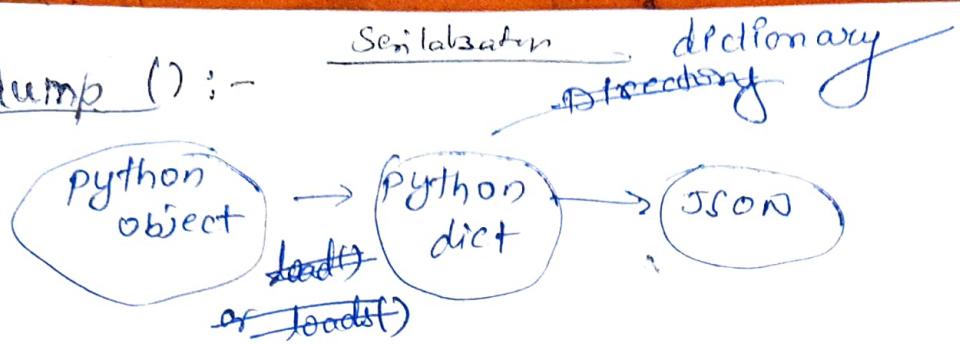
to written into the file supported format or memory is known as Pickle or Marshal.

unpickle: As well as it is a method to convert file supported object into python object through byteStream is known as unpickle.

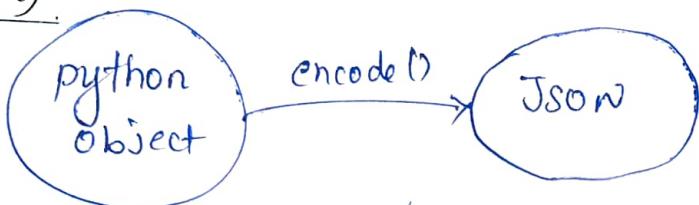
We can use 4 method for executing the program or unpickle program

- | | |
|--|--------------------------|
| 1) <code>Dump()</code> or <code>dump(s())</code> | 3) <code>encode()</code> |
| 2) <code>load()</code> or <code>loads()</code> | 4) <code>decode()</code> |

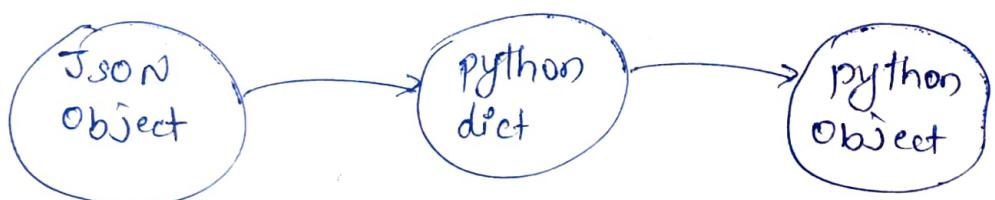
dump () :-



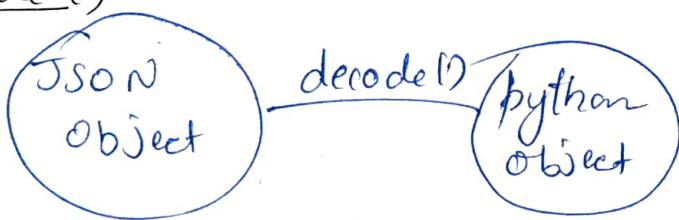
encode()



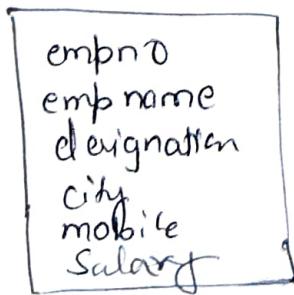
load ()



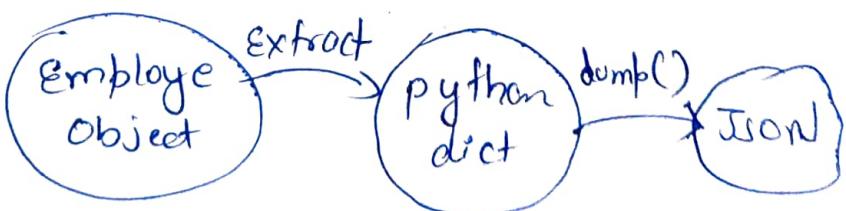
decode ()



@ example :-



employee class

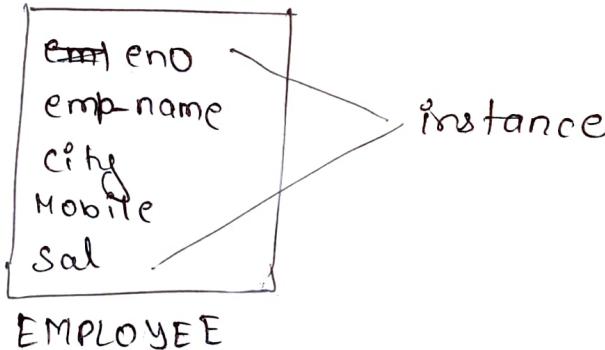


By using JSON Pickle module we can serialize or pickle our custom class objects directly to JSON object by the help of encode method..

Similarly we can deserialize JSON object to our custom class object with the help of decode method.

NOTE : JSON module not directly available by default in python IDLE (Integrated development Learning Environment). so we can install the Pip installer with JSON Pickle.

Q WA program to convert python object to JSON object in case of pickling using concept of OOPs



Class EMPLOYEE :

```
def __init__(self, empno, emp-name, city, mobile, sal):  
    self.empno = empno  
    self.emp-name = emp-name  
    self.city = city  
    self.mobile = mobile;  
    self.salary = sal;
```

```
def emp-Result(self):
```

```
    print(f"EMP NO = {self.empno} EMPName =
```

{self.emp-name} city = {self.city} Mobile =
{self.mobile} and salary is = {self.sal}")

emp1 = EMPLOYEE(1234, "Rakesh", "Ranchi", 5678,
1000)

serialization in Python to JSON

```
JSON_string = jsonpickle.encode(emp1);
print(JSON)
emp1.emp_result()
```

* Thread Programming

17/04/25

- Thread : A single flow of execution is known as thread and Multiple flow of execution is known as Multithreading.

A multithreading is much - much similar to Multitasking

• Multitasking

- Task / job / process : A running state of the program is known as task.

Task are of two types :

① single Tasking : we can perform single process at a time.

e.g :- cmd, Notepad

② Multi-tasking : At the same time user can perform multiple job.

Like → downloading, printing, editing the document at the same time.

- Multitasking is classified into two categories

1) process based

Multitasking \Rightarrow scheduling Algorithm (FCFS)

2) Thread based

Multitasking

Each process occupied its own memory

P_1, P_2, \dots, P_n

process	memory
P_1	"
P_2	"
P_3	"
P_n	"

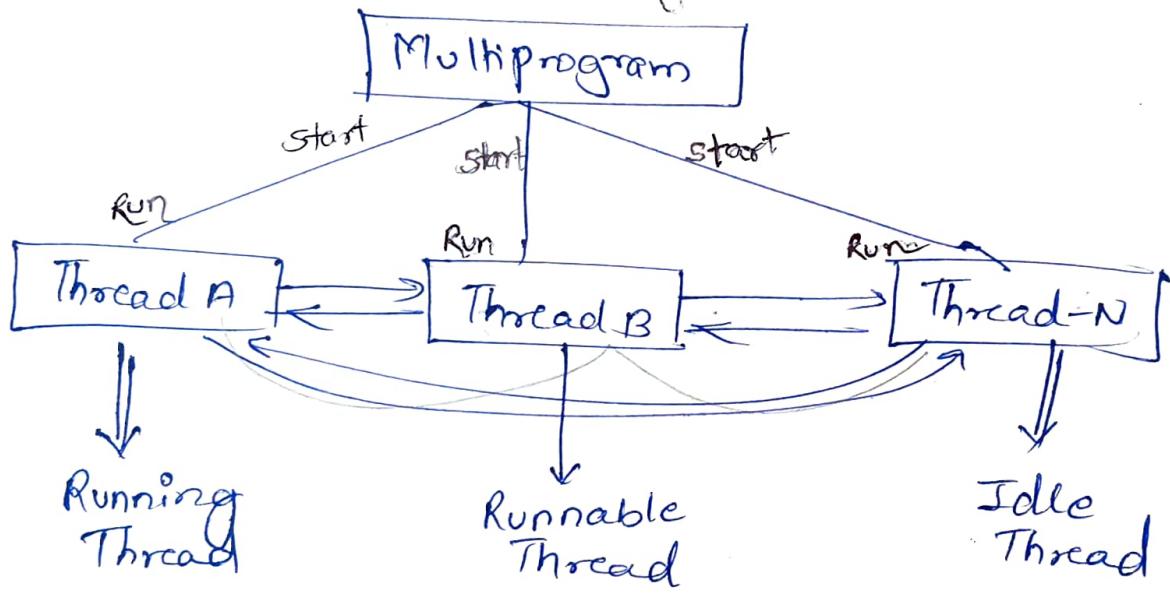
priority based
Thread parallelity

1) process based Multitasking :-

Each task is an independent of or has independent program and each task contains individual memory or its own memory for executing the program.

process based Multitasking work on the principle of scheduling Algorithm.

2) Thread based Multitasking :-



In thread based multitasking, A single process is divided into More than ~~n - subprograms~~ n - sub process and it will take single memory allocation for executing the program.

example :-

Suppose we have a hundred student in a coaching. each batch has 10 students then 10 batch will be created by one teacher. and take 4 hours then management decide hire another 3 teacher and distribute batch

among them. that means every teacher is a thread, and class room is a memory and each thread take classes parallelly or concurrently at the same time. This phenomena is known as Thread parallelism.

• Multithreading (Definition)

A multithreading is a conceptual programming where a main program or thread can be divided into packet (break into small pieces of program or fragmented) or sub program and all program will be executed concurrently or simultaneously or parallelly. This is known as Thread Parallelism or Multithreading.

19/04/25.

* Multithreading

creating thread

we can create Thread into two ways:

① By Thread class

② By Extending Thread class (By the help of Inheritance)

• By using thread class how to create a thread

step I: using API packages

from threading import *

or

[import threading]

Step II : To create a new thread. It will be possible by the help of thread object.

Create to newobj

Object.name = threading.Thread(target = (), args =)

OR

Object.name = Thread(target = ())

where args is optional

Step III : To execute and stop the thread object.

for execution

(a) Object.name.start()

for stop / end the thread

(b) Object.name.join()

Example

1.program: WA python Program to implement the thread class and print 1st 10 natural number using the help of function.

```
import threading  
or  
from threading import *
```

```
class Natural :
```

```
    def compute(self):
```

```
        num = 1
```

```
        while (num < 11):
```

```
            print(num)
```

OR

```
print(f"the Natural No. is =\n{self.num}")  
num = num + 1
```

obj1.Natural()

thread1 = threading.Thread(target=obj1.compute)

thread1.start()

thread1.join()

Q. Write a python program to implement thread by printing even number within given range.

from threading import *

import threading

class EvenNumber:

def display(self):

i=10

j=20

for num in range(i, j + 1):

if (num % 2 == 0):

print(num)

obj1 = EvenNumber()

Thread1 = threading.Thread(target=obj1.display)

Thread1.start()

Thread1.join() ↳ optional

Syntax of for loop :-

for counter in range(start, stop):

↑
increment/decrement
Optional
By default increment by 1

Q3 Write a python program to create two thread and print even number and odd number between 30 to 50 using user defined function in thread.

```
import threading
```

```
def Even():
```

```
    print("List of Even Number = ")
```

```
    for num in range(30, 51, 2):
```

```
        print(num, end = " ")
```

```
def Odd():
```

```
    print("List of Odd Number = ")
```

```
    for num1 in range(31, 49, 2):
```

```
        print(num1, end = " ")
```

```
thread1 = Thread(target = Even())
```

```
thread2 = Thread(target = Odd())
```

```
thread1.start()
```

```
thread2.start()
```

```
thread1.join()
```

```
thread2.join()
```

end = " "
30 ↑
32 give new line

Assignment

1. WAP Python program to implement Thread class and find the factorial of given number

2. WAP to find the factorial of given number using user defined function in case of thread.

↑ without Thread class

1

```
import threading  
class Factorial:  
    def compute(self):  
        num = 5  
        fact = 1  
        for i in range(5, 1 - 1):  
            num = num * i  
            fact = fact * i  
    obj.Factorial()  
thread1 = threading.Thread(target = obj.compute())  
thread1.start()  
thread1.join()
```

2.

```
import threading  
def compute(self):  
    num = 5  
    fact = 1  
    for i in range(num, 1 - 1):  
        fact = fact * i  
thread1 = Thread(target = compute())  
thread1.start()  
thread1.join()
```

1 import threading
class factorial:
 def compute(self):
 num = 5
 fact = 1
 for i in range(5, 1 - 1):
 num = num * i
 fact = fact * i
 obj.factorial()
 thread1 = threading.Thread(target=obj.compute())
 thread1.start()
 thread1.join()

2. import threading
def compute(self):
 num = 5
 fact = 1
 for i in range(num, 1 - 1):
 fact = fact * i
 thread1 = Thread(target=compute())
 thread1.start()
 thread1.join()

23/04/20

Thread Programming

In python we can create thread into 3 ways -

- (1) By thread class (OOPs)
- (2) By function
- (3) By extending class (OOPs)

(3). By Extending class : we can implement the concept of method overriding in the implementation.
A extend class (Inheritance + Method overriding)

Polymorphism is a greek word ;
one name and multiple form

A word that have several meaning depending upon the demand.

Method overriding - Method overriding refers to the method in sub class with same name as method in super class

NOTE - python provide inbuilt class thread and we have to create a class by extending from a thread class. Inside the thread class python provide inbuilt function run() that means run() method will be inside the thread class and we have to override in run() method from super class to sub class.

In another word, we can say that a run() method is the heart and soul of thread programming

In case of extend class, these method will be control the entire behaviour of thread class and run() method must be invoked by another thread method which is start().

The run() method will override the thread execution logic when we call start() method.

NOTE - When we start a python program one thread run immediately and that thread is known as parent thread or Main thread. and we can inherit the properties of main thread to new created thread.

* thread - It is a main() thread and we can call its property by newly created class.

Example:-

extending Thread class

```
from threading import *
class Test(Thread):
    def run(self):
        for i in range(5):
            print("AMIT")
class Test1(Thread):
    def run(self):
        for i in range(5):
            print("Adv. Python")
```

```
obj1 = Test()
obj2 = Test1()
obj1.run()
obj2.run()
obj1.start()
obj2.start()
```

what should be O/P

AMIT
AMIT
↓
AMIT
Adv. Python
Adv. Python
:
Adv. python

Example 2: creating thread by extending class
implementing constructor (constructor overriding)

import

```
from Threading import *
class Test(Thread)
    def __init__(self, name):
        Thread.__init__(self)
        self.name = name
    def run(self):
        for i in range(5):
            print(self.name)
            time.sleep(1)
```

* parent class
constructor

```
class Test1(Thread):
    def __init__(self, name):
```

Thread. __init__(self)

self.name = name

def run(self):

for i in range(5):
print(self.name)

thread1 = Test("fawaz")

thread2 = Test("moh")

thread1.start()

thread2.start()

thread1.join()

thread2.join()

```
Thread. __init__(self)
    self.name = name
def run(self):
    for i in range(5):
        print(self.name)
```

```
thread1 = Test("Ramesh")
```

```
thread2 = Test1("MCA")
```

```
thread1.start()
```

```
thread2.start()
```

```
thread1.join()
```

```
thread2.join()
```

* Thread Communication by Queue Module. 26/04/25

Queue



Linear data structure (sequential data structure)

- ① producer - It will be insert the data element at one end \Rightarrow Rear end
- ② consumer - It will deleted the element at the other end \Rightarrow Front end

In other word, we can say that we can produced or consumed the data element with enqueue () and dequeue ()

python - we can use queue module in thread communication or thread object

for this purpose we can implement queue.Queue() class



It assigned on python API ~~from~~
~~from threading import Queue~~

we can use two method for performed operation on thread object

- ① put () \Leftarrow enqueue ()
- ② get () \Leftarrow dequeue ()

NOTE: There are two primitive operation performed in thread Queue

- ① insertion (producer) by `put()`
- ② deletion (consumer) by `get()` function.

Syntax of put()

`put(item, True / False)`
or
`put(item, block = True)` optional

Syntax get()

`obj1.get()`

Syntax of Queue

```
import threading, queue
q = queue.Queue()
def producer():
    q.put(item)
def consumer():
    q.get()
q.task_done()
```

By default, `queue.Queue` is FIFO, infinite, blocking and thread-safe.
by default no need to manually use locks.

threading.Thread
(target=producer).
start()

threading.Thread
(target=consumer).
start()

no limit
maxsize is 0
get() will start(wait)
if queue is empty

put() will block(wait)
if the queue is full

Example: write a python program to implement thread communication in case of `Queue` model for insertion and deletion of element.

|||||
Queue entry

|10/20/30/40|
enqueue (front)
dequeue (rear)

Program

```
from threading import Queue
```

```
def producer(my-queue):
```

```
    point (" Producer is running = ")
```

```
    num = int(input (" Enter the size of Queue  
    thread = " ))
```

```
    for i in range (1, num):
```

 ~~point (" Input = ")~~

```
    marks = float(input (" Enter student  
    marks = " ))
```

```
my-queue.put (marks)
```

```
my-queue.put (None)
```

```
point (" Producer end ")
```

```
consumer (my-queue)
```

```
point (" consumer is working = " ))
```

```
while True
```

```
item= my-queue.get()
```

```
if item is None:
```

```
break
```

```
print (" The item is = {item} ")
```

```
my-queue = Queue()
```

```
thread1 = Thread (target = producer ,  
arg = my-queue)
```

```
thread2 = Thread (target = consumer , arg = my-queue)
```

```
thread1.start()
```

```
thread2.start()
```

```
thread1.join()  
thread2.join()
```

Array

Array Module

Numpy \Rightarrow Numerical Python

Array :- It is a special variable which can store multiple element of same type.

Array in python is a special variable which can hold more than one value.

In another word we can say that array in python is a data structure that store collection of elements of same data type. that means an array is a collection of similar data types that of can be accessed by a common name that means an array can be defined as storage or container or similar item in our continuous memory allocation. An array are useful for storing and manipulating the sequence of data such as list of number or character.

In python, we can create an array into two way :

- ① Array Module
- ② Numpy Library

1) Array Module :

The Array Module provide basic array function whereas numpy offer advance feature of Numerical python.

• Types of Array

Generally, we can use 3 types of array in Python

- ① 1-D Array
- ② 2-D Array
- ③ Boolean Array (True/False)

① 1D Array :

One-dimension One-D Array can be represented by single subscript [] with one variable and also uses open source inbuilt library array () .

* Declare and initialization of Array in python (How to declare the Array)

Step 1 : 1st of all we can use open source inbuilt library and package.

`from array import *`

or

`import array`

`import array
as arr`

array module is available in Python standard library

Step 2: Declare the syntax of one-d array

variable-name = array (datatype, [no. of elements])

example arr-num = array ('i', [Number])

or

variable-name = arr, array (datatype, [no. of elements])
p = arr.array ('i', [element+1])

Step 3: Body of the Array

PROGRAM: Write a python program to create 5 integers element in array and display them, also display 1st element, 3rd element and last element.

Rule :- An array always start with 0 as base address and with size - 1 (nth)

arrnum =

7	11	3	5	2
0	1	2	3	4

```
from array import *
```

import array as arr

```
arr-num = arr.array ('i', [7, 11, 3, 5, 2])  
print (arr-num)
```

```
for i in arr-num:  
    print(i)
```

```
print("the first element of given array =  
      [arr-num[0]]")  
  
print("the first element = [arr-num[0]]")  
print("the last element = [arr-num[9]]")
```

Q → write a python program to create a float element and add one element at the end of the list in given array.

arr-num = [7.1, 11.2, 3.15, 2.4, 7]

• append () :

With the help of append function we can insert one element at end of the list.

```
from array import *  
arr-num = array('d', [7.0, 11.0, 3.0, 5.0, 2.0])  
print("The No. of element = [arr-num]")  
arr-num.append(17)  
print("After Modification of element =  
      [arr-num]")
```

Lab Assignment

Thread and Program

- WAP to create a thread with the help of main thread and implement constructor overriding in case of student, information system, import threading
class Person:

```
def student (Person):
    def __init__ (self, name, age):
        self.name = self.name
        self.age = name, age
    class student (Person):
        def __init__ (self, name, sid, course):
            super (). __init__ (name, age)
            self.sid = self.sid
            self.course = sid, course
        def display (self):
            print ("Name:", self.name, "Age:", self.age)
            print ("ID:", self.sid, "Course:", self.course)
    def show (student):
        student.display ()
```

if

```
    name = "main"
    s = student ("Alice", 21, "S123", "ICA")
    t = threading.Thread (target=s.show, args=(s,))
    t.start()
    t.join()
```

Name: Alice, Age : 21, ID: S123, Course : ICA

2) Write Thread programming to find multiplication table with the help of user defined function.

import threading

def print_table(num):

```
print("Multiplication Table for %d:" % num)
for i in range(1, 11):
    print("%d * %d = %d" % (num, i, num * i))
if __name__ == "__main__":
    number = int(input("enter a number to print its multiplication table:"))
    t = threading.Thread(target = print_table,
                         args = (number,))
    t.start()
    t.join()
```

Output:-

enter a number to print its multiplication table: 5
Multiplication Table for 5:

$$5 \times 1 = 5$$

$$5 \times 2 = 10$$

$$5 \times 3 = 15$$

$$5 \times 4 = 20$$

$$5 \times 5 = 25$$

$$5 \times 6 = 30$$

$$5 \times 7 = 35$$

$$5 \times 8 = 40$$

$$5 \times 9 = 45$$

$$5 \times 10 = 50$$

- 3) define Thread. Architecture in Python

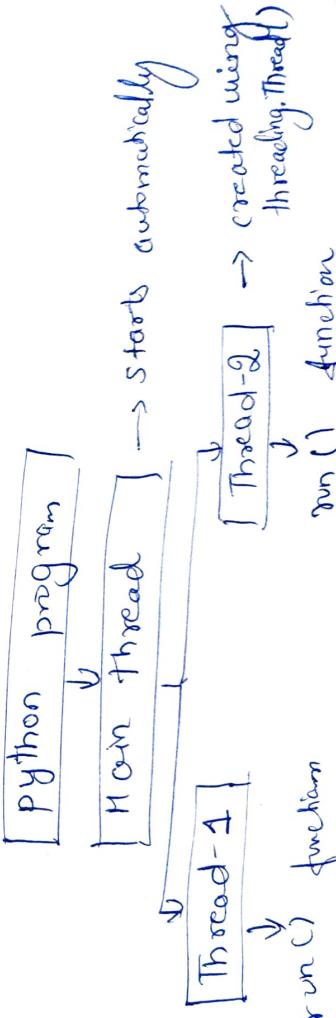
Thread Architecture in Python (Short version)

- Python uses the threading module to run multiple threads (smaller units of process) concurrently.
- The main thread starts when the program begins.
- New child threads can be created using the Thread class
 - Each thread runs a function using start() and completes with join()

example

```
from threading import Thread
def task():
    print("Thread is running")
t = Thread(target=task)
t.start()
t.join()
```

- Python has a Global Interpreter Lock (GIL), so threads are ideal for I/O-bound tasks, not CPU-heavy ones
- visual diagram



Q) write a python thread program to find the addition of two numbers using the global variable.

import threading

result = 0

def add () :

global result

a = int (input ("Enter first number : "))

b = int (input ("Enter second number : "))

result = a + b

print ("Addition = {result}")

t = threading.Thread (target = add).

t.start ()

t.join ()

output

Enter first number : 10

Enter second number : 20

Addition = 30

W A multi threading program
constructor overloading from Main Thread to
sub Thread

```
from threading import *
import time
class Test(Thread):
    def __init__(self, name):
        Thread.__init__(self)
        self.name = name
    def run(self):
        for i in range(5):
            print(self.name)
            time.sleep(1)
class Test1(Thread):
    def __init__(self, name):
        Thread.__init__(self, name)
        self.name = name
    def run(self):
        for i in range(5):
            print(self.name)
            time.sleep(1)
obj1 = Test("Vibha")
obj1.start()
obj2 = Test1("Aani")
obj2.start()
```

WAP Python programming to declare the object base class and derived class in case overriding method and show its effect in case of Employee and Faculty information.

```
class Employee:  
    def __init__(self, nm, sal):  
        self.name = nm  
        self.salary = sal  
  
    def getName(self):  
        return self.name  
  
    def getSalary(self):  
        print(self.salary)  
  
class Faculty(Employee):  
    def __init__(self, nm, sal, inc):  
        super().__init__(nm, sal)  
        self.inc = inc  
  
    def getSalary(self):  
        return self.salary + self.inc  
  
e1 = Employee("Rajesh", 9000)  
e1.getName()  
e1.getSalary()  
  
f1 = Faculty("AMIT", 10000, 1000)  
f1.getSalary()  
f1.getSalary()
```

print("The Total salary is = .{s1.getSalary()})

Python programming to declare the object base class and derived class in case overriding method and show its effect in case of Employee and Faculty information.

Write a Multithread program with the help of Main thread and override run method with the help of start method

```
from threading import *
import time
class Test(Thread):
    def run(self):
        for i in range(5):
            print("Amit")
            time.sleep(2)
    class Test2(Thread):
        def run(self):
            for i in range(5):
                print("RANCHI")
                time.sleep(1)
obj1 = Test()
obj1.start()
obj2 = Test2()
obj2.start()
```

Output

Amit	RANCHI
RANCHI	RANCHI
Amit	RANCHI
RANCHI	RANCHI
Amit	RANCHI
RANCHI	Amit
Amit	RANCHI
RANCHI	Amit

WAP in python to implement the concept
of overriding in case of arithmetic operations
like (+, -, *, %, /) using OOPs

```
class Arith:  
    def calc(self, a, b):  
        pass  
  
    class Addition(Arith):  
        def calc(self, a, b):  
            s = a + b  
            print("sum : ", s)  
  
    class Subtraction(Arith):  
        def calc(self, a, b):  
            s = a - b  
            print("sum : ", s)  
  
    class Multiplication(Arith):  
        def calc(self, a, b):  
            m = a * b  
            print("multiply : ", m)  
  
    class Modulus(Arith):  
        def calc(self, a, b):  
            m = a % b  
            print("Modulus : ", m)  
  
    class Division(Arith):  
        def calc(self, a, b):  
            d = a / b  
            print("division : ", d)
```

obj1 = Addition()
obj2 = Subtraction()
obj3 = Multiplication()
obj4 = Modulus()
obj5 = Division()

obj1 . calc (20, 30)
obj2 . calc (40, 10)
obj3 . calc (24, 3)
obj4 . calc (50, 3)
obj5 . calc (24, 4)

obj :=

sum := 50
sub := 30
Multiply := 72
Module := 2
Division := 2.66666666666665

Numpy Assignment (Python)

02/05/25

WAP to input 5 numbers in Numpy array and display it in reverse order

```
import numpy as np
```

```
arr_num = []
n = int(input("Enter the size of array := "))
```

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

```
    val = int(input("Enter the Element := "))
```

```
    arr_num.append(val)
```

```
x = np.array(arr_num)
```

```
print("----Output----") / np.reverse(arr_num)
```

```
print(x)
```

```
print("Reverse output")
```

```
print(result)
```

```
print("----Numpy---")
```

```
for x1 in x:
```

```
    print(x1)
```

```
print("----Reverse order---")
```

```
for x in result:
```

```
    print(x) ✓
```

WAP to input 3*3 matrix using Numpy array

```
import numpy as np
mat1 = []
row = int(input("Enter the Row"))
col = int(input("Enter the Col"))
for i in range(row):
    a = []
    for j in range(col):
        val = int(input("Enter the No ="))
        a.append(val)
    mat1.append(a)
arr = np.array(mat1)
print("-----Matrix Numpy -----")
for i in range(3):
    for j in range(3):
        print(arr[i][j], end=" ")
```

WAP to input 5 number in Numpy array and display it.

```
import numpy as np
arr-num = []
n = int(input("Enter the size of array :="))
for i in range(n):
    val = int(input("Enter the element :="))
    arr-num.append(val)
x = np.array(arr-num)
print ("---Output---")
print(x)
print("--- Numpy ---")
for x1 in x:
    print(x1)
```

Subtraction of Matrix by user input

```
print("Enter the Row and column size of
First Matrix : ", end = " ")
rowOne = int(input())
colOne = int(input())
print("Enter the Row and column size of
Second Matrix : ", end = " ")
rowTwo = int(input())
colTwo = int(input())
```

If rowOne == rowTwo and colOne == colTwo :

mat1 = []

```
print("Enter 9 element for first Matrix : ", end = " ")
```

```
for i in range (rowOne):
    mat1.append ([ ])
for j in range (colOne):
    num = int (input ())
    mat1[i].append (num)

mat2 = []
print ("Enter g Elements for second Matrix:", end="")
for i in range (rowTwo):
    mat2.append ([ ])
    for j in range (colTwo):
        num = int (input ())
        mat2[i].append (num)

mat3 = []
for i in range (rowOne):
    mat3.append ([ ])
for j in range (colTwo):
    sub = mat1 [i][j] - mat2 [i][j]
    mat3[i].append (sub)

print ("Subtraction Result:")
for i in range (rowOne):
    for j in range (colOne):
        print (mat3[i][j], end=" ")
    print ()

else:
    print ("In Order Mismatched!")
```

Transpose Matrix by user input

```
rows = int(input("Enter the Number of rows:"))
columns = int(input("Enter the Number of columns:"))

print("Enter the elements of Matrix:")
matrix = [[int(input()) for i in range(columns)]
          for j in range(rows)]

print("Your Matrix is ---")
for n in matrix:
    print(n)

result = [[0 for i in range(rows)] for j
          in range(columns)]

for r in range(rows):
    for c in range(columns):
        result[c][r] = matrix[r][c]

print("Transpose matrix is:")
for r in result:
    print(r)
```

O/p :- Enter the number of rows : 3
Enter the number of columns : 2
Enter the elements of Matrix :

1
2
3
4
5
6

---Your matrix is ---

[1, 2]
[3, 4]
[5, 6]

Transpose matrix is :

[1 3 5]
[2 4 6]

WAP Python program to input 6 numbers and find maximum and minimum among them using Array module.

```
from Array import *
arr-num = []
for i in range(6):
    num = int(input("Enter 6 element"))
    arr-num.append(num)
largest = arr-num[0]
min = arr-num[0]
for i in range(1, len(arr-num)):
    if arr-num[i] > largest:
        largest = arr-num[i]
        print("Largest = ", arr-num[i], largest)
    elif arr-num[i] < min:
        min = arr-num[i]
print("Smallest element", min)
```

O/P:- enter 6 elements 65
enter 6 elements 32
enter 6 elements 14
enter 6 elements 23
enter 6 elements 87
enter 6 elements 25
Largest = 87
Smallest element - 14

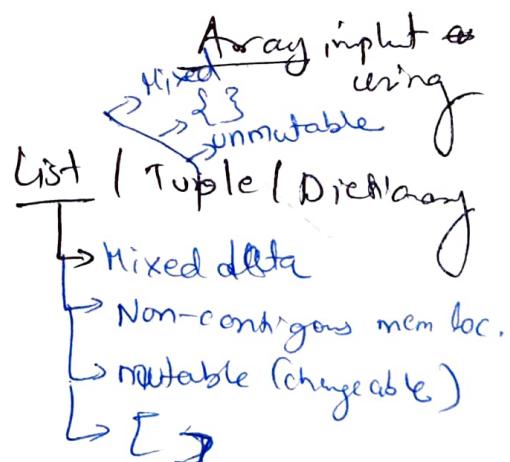
Numpy (Numerical Python)

Numpy is open-source python library mainly used for science and engineering. It is universal standard for working with numerical data in python. It is a core of scientific python and py data ecosystem. In another word we can say that numpy is a part of array unit mainly used for multi-dimensional array and matrix data structure. That means we can say that numpy can be used to perform varieties of mathematical operation on array. It add powerful datastructure to python that efficient calculation with array and matrix. By the help of numpy we can perform the linear algebra problem and Fourier transformation problem etc.

Why Numpy similar data

↑
Nd array: array object
↑
nth dimensional array

- ① fast 50x
- ② scientific calc - linear algebra, Fourier transform
- ③ contiguous memory alloc
- ④ complex data structure



In python we have list that serve the purpose of array but they are slow to the process and numpy aim to provide an array object that is upto

50 time faster than python list. numpy array are stored at one continuous place in memory unlike like list. so process can be easily access and manipulate very efficiently due to this reason we can use numpy.

• Syntax of numpy : pip installer download

doesn't support

Numpy is open-source library \Rightarrow Package numpy

~~numpy~~

Pythond 3 → apps

Step I: import numpy
or

import numpy as np

or

from numpy import *

from array import *

Step II: create an empty array
variable name = []

e.g: arr_numpy = []

Step III: body of array

point & point if

↑
no need to
use concatenate

Example (program)

Write a python program to implement an array/ numpy array to # input number of element by the user and display them.

arr_num = [0 1 2 3 4
 | 7 | 9 | 3 | 11 | 37]

Without numpy

```
from array import *
import numpy as np
arr_num=[]
num=int(input("Enter the size of Array="))
for i in range(num):
    val=int(input("Enter the element ="))
    x=arr_num.append(val)
```

print(f" The element of Array={arr_num}")
OR
print(f" The element is ={arr_num[i]}")

OR

{ for x1 in x :
 print(x1) } → o/p
 7
 9
 3
 11
 37

optional

o/p 7 9 3 11 37

Q. write a program to implement
numpy to input the number
the help of the user and print
positive number or sum of the
an array or
of element by
sum of the
negative number

```
from array import *
import numpy as np
arr_num = []
num = int(input("Enter the size of Array = "))
for i in range(num):
    val = int(input("Enter the element"))
    arr_num.append(val)
print(f"-- output = {arr_num} -- b = 0")
print(f" The element of Array = {arr_num}")
if for i in range(num):
    if (arr_num[i] >= 0):
        print(arr e = e + arr[i])
        b = b + arr_num[i]
    else:
        n = n + arr_num[i]
```

```
b = 0
n = 0
print(f" the sum of positive no. = {b}")
print(f" the sum of negative no. = {n}")
```

Q. Write a python program to implement Numerical python to input number of element by user and display it in matrix form.

Inputting

Enter no. of rows = 3
 Enter no. of column = 3
 Enter the element
 1
 2
 3
 4

Output matrix

1	2	3
4	5	6
7	8	9

```
from array import *
R = int(input("Enter no. of rows = "))
C = int(input("Enter no. of column = "))
matrix = []
print("Enter the element")
for i in range(R):
    a = []
    for j in range(C):
        a.append(int(input()))
    matrix.append(a)
for i in range(R):
    for j in range(C):
        print(matrix[i][j], end=" ")
print()
```

Q. Write a python program to implement Numerical python to input number of element by user of two matrix and find addition of matrix

mat 1

Enter no. of row = 3
 Enter no. of column = 3
 Enter the element
 1
 2

mat 2
 Enter no. of row
 11 12 13
 14 15 16
 17 18 19
 Enter no. of column

mat 3 = mat 1 + mat 2

12	14	16
13	20	22
14	28	28

```

from array import *
R = int(input("Enter no. of row = "))
C = int(input("Enter no. of column = "))
matrix = []
print("Enter the element Rowsize = ")
for i in range(R):
    a = []
    for j in range(C):
        a.append(int(input()))
    matrix.append(a)
for i in range(R):
    for j in range(C):
        print(matrix[i][j], end=" ")
    print()

```

Assignment

Q. Write a python program to implement numpy to reverse of an array that means first element become a last element using array slicing.

import numpy as np

original_array = np.array([10, 20, 30, 40, 50])

reversed_array = original_array[::-1]

print("Original array : ", original_array)

print("Reversed array : ", reversed_array)

Array Slicing -

It is a technique used in python (and NumPy) to extract a portion of an array or reverse its elements without explicitly looping.

Syntax : array [start : stop : step]

Example :

- print(arr[1:4])
- print(arr[::2])
- print(arr[:: -1])

Output : [20 30 40]

Output : [10 30 50]

Output : [50 40 30 20 10]

program

write a python program to implement Array/Numpy to input number of row and column for matrix and perform addition.

```

from array import *
import numpy as np
R = int(input("Enter the number of rows of
    1st matrix ="))
C = int(input("Enter the number of columns of
    1st matrix ="))
print("Enter the elements of 1st matrix =")
mat1 = []
for i in range(R):
    a1 = []
    for j in range(C):
        a1.append(int(input())))
    mat1.append(a1)
# display the 1st matrix
print("1st Matrix -1")
print(mat1[i][j], end = " ")
print()
R1 = int(input("Enter the Range of 2nd matrix"))
C1 = int(input("Enter the col of 2nd matrix"))
print("1st Enter the element")
mat2 = []
for i in range(R1):
    a2 = []
    for j in range(C1):
        a2.append(int(input()))
    mat2.append(a2)
    
```

```

# display the 2nd matrix
print ("A matrix-2 ")
print (mat2[i][j] end = " ")
print ()

print ("A ADDITION OF TWO MATRIX=")
if R1 == R2 and C1 == C2:
    for i in Range (R):
        for j in range (C):
            print (mat1 [i][j] + mat2 [i][j],
                  end = " ")
    print ()
else:
    print ("Addition is not possible")

```

NOTE: The order of 1st matrix should be equal to order of 2nd matrix.

program 2

wAP python program to implement array module to input number of rows and column for 3*3 matrix and find transpose of matrix

mat1

1	2	3
4	5	6
7	8	9

Transpose

1	4	7
2	5	8
3	6	9

from array import *

R = int(input ("Enter the no. of rows = "))
C = int(input ("Enter the no. of columns = "))
print ("Enter the 9 elements of matrix z")
mat2 = []

```

for i in range (R):
    mat3 = []
    for j in range (C):
        mat3.append(int(input())))
    mat.append(mat3)
or
np.array(mat)
print ("A matrix = ")
for i in range (R):
    for j in range (C):
        print (mat[i][j], end = " ")
    print ()
print ("Computation of Transpose = ")
for i in range (C):
    for j in range (R):
        print (mat[i][j], end = " ")
    print ()

```

Assignment

WAP a numpy program to create 3×3 matrix with the help of a vector value 2 to 10.

```

import numpy as np
vector = np.arange(2, 11)
matrix = vector.reshape(3, 3)
print ("3x3 Matrix:")
print (matrix)

```

Q:- 3x3 matrix:

```

[[2 3 4]
 [5 6 7]
 [8 9 10]]

```

Numerical python (Array) unit

Q → write a program to implement numpy array to input number of element by the user and find sum of (+)ve number and sum of (-)ve number

```
From array import *
import numpy as np
arr-num = []
sump = 0
sumn = 0
num = int(input("Enter the size of Array"))
for i in range(num):
    val = int(input("Enter the element ="))
    arr-num.append(val)
    result = np.array(arr-num)
    print(f"the array element = {result}")
for i in range(val):
    if arr-num[i] > 0:
        sump = sump + arr-num[i]
        print(f"The sum of [+ve] = {sump}")
    else:
        sumn = sumn + arr-num[i]
        print(f"The sum of [-ve] = {sumn}")
```

Q → WAP to implement numpy to input 10 nos of elements and display them with the help of array

```
from array import *  
import numpy as np  
arr,num=[ ]
```

```
num=int(input("Enter the 1st element ="))  
for i in range(num):  
    val=int(input("Enter the element ="))
```

Q → WAP to input 5 number in Numpy array and display it is reverse order

```
import numpy as np  
arr-num=[ ]
```

```
n=int(input("Enter the size of array:"))  
for i in range(n):
```

```
    val=int(input("Enter the element ="))  
    arr-num.append(val)  
x=np.array(arr-num)  
result=np.flip(arr-num)
```

```
print("--output--")
```

```
print(x)
```

```
print("Reverse output")  
print(result)
```

```
print("-- Numpy --")  
for x1 in x:  
    print(x1)
```

```
print("-- Reverse order --")  
for x in result:  
    print(x)
```

Lab Assignment

03/05/25

Numerical package

- ▷ WAP a python program to implement NumPy as input number of rows and columns in the form matrix and find sum of rows and sum of column

```
import numpy as np
from array import array
R = int(input("Enter the rows of matrix"))
C = int(input("Enter the column of matrix"))
mat1 = []
print("Enter the elements of matrix")
for i in range(R):
    a1 = []
    for j in range(C):
        a1.append(int(input()))
    mat1.append(a1)
print("sum of row and sum of column")
print(np.sum(mat1, axis=1))
print(np.sum(mat1, axis=0))
```

Assignment: Numpy computation of Matrix addition

from array import *

R = int(input("Enter the rows of 1st matrix"))
C = int(input("Enter the columns of 1st matrix"))
mat1 = [[]]

mat2 = [[]]

print("Enter the elements of 1st matrix")
for i in range(R):

a = []

for j in range(C):

a.append(int(input()))

mat1.append(a)

print("matrix -1")

for i in range(R):

for j in range(C):

print(mat1[i][j], end=" ")

print("")

R = int(input("Enter the rows of 2nd matrix"))
C = int(input("Enter the columns of 2nd matrix"))
mat2 = [[]]

print("Enter the elements of 2nd matrix")
for i in range(R):

a = []

for j in range(C):

a.append(int(input())))

mat2.append(a)

print("matrix 2")

for i in range(C):

print(mat2[i][j], end=" ")

print("")

print ("Addition of matrix 1")

for i in range (4) :

 for j in range (4) :

 print (mat [i][j], end = " ")

Output

Enter the rows of 1st matrix 2

Enter the column of 1st matrix 2

Enter the elements of 1st matrix

1
2
3
4

matrix -1

1 2
3 4

Enter the rows of 2nd matrix 2

Enter the column of 2nd matrix 2

Enter the elements of 2nd matrix

5
6
7
8

matrix 2

5 6
7 8

Addition of matrix

6 8
10 12

WAPP to input size of rows and column as well as elements of array and compute transpose of array

from array import *

R = int(input("Enter the no. of rows of matrix"))

C = int(input("Enter the no. of columns of matrix"))

mat = [[]]

print ("Enter the elements of matrix")

for i in range (R):

 for j in range (C):

 a1.append (int (input ()))

mat1.append (a1)

print ("matrix")

for i in range (R):

 for j in range (C):

 print (mat1 [i][j], end = " ")

print ("Transpose")

for i in range (C):

 for j in range (R):

 print (mat1 [j][i], end = " ")

print ("")

WAP to implement Numpy and enter number of elements and find sum of + and sum of -ve using numerical package.

```
from numpy import *
import numpy as np
arr-num = []
sump = 0
sumn = 0
```

```
num = int(input("Enter the size of array"))
for i in range(num):
    val = int(input("Enter An element"))
    arr-num.append(val)
    print(f"\n The element {i} = {array-num[i]}")
for i in range(num):
    if array-num[i] > 0:
        sump = sump + array-num[i]
    else:
        point(f"\n sumpt array-{num[i]}")
```

else :

```
sumn = sumn + val
print(f"\n sumn {sumn}")
```

WAPP to implement NumPy or Array object
to input number of rows and columns for
matrix and sum of Diagonal in matrix

```
import numpy as np
from array import array
R = int(input("Enter the rows of matrix"))
C = int(input("Enter the cols of matrix"))
mat = []
for i in range(R):
    arr = []
    for j in range(C):
        arr.append(int(input("Enter the elements of matrix")))
    mat.append(arr)
print(mat)
```

```
R = int(input("Enter the rows of matrix"))
C = int(input("Enter the cols of matrix"))
mat = []
for i in range(R):
    arr = []
    for j in range(C):
        arr.append(int(input("Enter the elements of matrix")))
    mat.append(arr)
print(mat)

for i in range(R):
    for j in range(C):
        print(mat[i][j], end=" ")
    print()

print("The Diagonal of matrix")
n = np.array(mat)
print(np.trace(n))
```

Matrix
from this
complex

WAP in Python using cmath unit for complex numbers arithmetic operation and find exponential, Trigonometric function, magnitude and basic operation.

```
import cmath  
  
# Define complex number  
z1 = 2 + 3j  
z2 = complex(-1, 1)  
  
# Basic operation  
print ("Addition : ", z1 + z2)  
print ("Subtraction : ", z1 - z2)  
print ("Multiplication : ", z1 * z2)  
print ("Division : ", z1 / z2)  
  
# Magnitude and phase  
print ("Absolute value (magnitude) of z1 : ", abs(z1))  
print ("Phase (angle) of z1 : ", cmath.phase(z1))  
  
# Exponent and logarithmic functions  
print ("Exponential of z1 : ", cmath.exp(z1))  
print ("Logarithm of z1 : ", cmath.log(z1))  
  
# Trigonometric functions  
print ("Sine of z1 : ", cmath.sin(z1))  
print ("Cosine of z1 : ", cmath.cos(z1))  
print ("Square root of z1 : ", cmath.sqrt(z1))
```

Numpy

10/05/25

Program 1: Write a python program to implement numpy / Array object for 3*3 matrix and find sum of the rows and column using Numerical package.

Rowwise	<table border="1" style="border-collapse: collapse; width: 100px; height: 40px;"> <tr> <td style="width: 33px; height: 13px;"></td><td style="width: 33px; height: 13px; border-right: 1px solid black;">2</td><td style="width: 33px; height: 13px; border-right: 1px solid black;">3</td></tr> <tr> <td style="width: 33px; height: 13px; border-right: 1px solid black;">1</td><td style="width: 33px; height: 13px; border-right: 1px solid black;">5</td><td style="width: 33px; height: 13px; border-right: 1px solid black;">6</td></tr> <tr> <td style="width: 33px; height: 13px; border-right: 1px solid black;">2</td><td style="width: 33px; height: 13px; border-right: 1px solid black;">8</td><td style="width: 33px; height: 13px; border-right: 1px solid black;">9</td></tr> <tr> <td style="width: 33px; height: 13px; border-right: 1px solid black;">7</td><td style="width: 33px; height: 13px;"></td><td style="width: 33px; height: 13px;"></td></tr> </table>		2	3	1	5	6	2	8	9	7		
	2	3											
1	5	6											
2	8	9											
7													
Columnwise	[6 15 24] \Rightarrow Rowwise [12 15 18] \Rightarrow columnwise												

Note :-

- Row wise can be represented by axis=1
column wise can be represented by axis=0 in numpy
- for summation of rowwise / columnwise array we can use Numpy Open library function sum().

```
import numpy as np
from array import *
```

```
R = int(input("Enter the Number of Rows"))
C = int(input("Enter the Number of columns"))
mat1 = []
print("Enter the elements of matrix = ")
for i in range(R):
    for j in range(C):
        for j in range(C):
```

```
a1.append(int(input()))
```

```
mat1.append(a1)
```

```
print("Matrix =")  
for i in range(R):  
    for j in range(C):  
        print(mat1[i][j], end = " ")  
    print()
```

```
result = np.array(mat1)  
print("Sum of the rows and sum of column")
```

```
print(np.sum(mat1, axis=1))  
print(np.sum(mat1, axis=0))  
OR
```

```
print(np.sum(result, axis=1))  
print(np.sum(result, axis=0))
```

Q. Write a python program to implement Numerical package in case of matrix programming and find sum of diagonal of 3*3 matrix using array object.

Inputting :

1	2	3
4	5	6
7	8	9

Output:

Sum of diagonal = 15

```
import numpy as np  
from array import *
```

```
R = int(input("Enter the Number of Rows = "))  
C = int(input("Enter the Number of columns = "))  
mat1 = [[ ]]
```

```

print("Enter the 9 elements of the matrix")
for i in range(9):
    a1 = []
    for j in range(9):
        a1.append(int(input()))
    mat1.append(a1)

print("Matrix")
for i in range(9):
    for j in range(9):
        print(mat1[i][j], end=" ")
    print()

print("Diagonal of matrix")
n=0; m=0
for i in range(9):
    for j in range(9):
        if (i==j):
            n=mat1[i][j]+n
        elif (i+j == 9):
            m=mat1[i][j]+m
    print()

print("1st Matrix before diagonal = ")
print(mat1)
print()
m = np.array(mat1)
print(np.trace(m))

```

*// NOTE: with the help of trace() function of numpy we can sum of diagonal in case of matrix.

WA numpy program to create 3x3 matrix with value in 12 to 20.

Range value :- [12 13 14 15 16 17 18 19 20]

↳ convert

```
[ 12  13  14  
 15  16  17  
 18  19  20 ]
```

```
import numpy as np  
vector = np.arange(12, 20)  
matrix = vector.reshape(3, 3)  
print ("3x3 Matrix :")  
print (matrix)
```

or

```
import numpy as np  
arr_num = np.arange(12, 20).reshape(3, 3)  
print (arr_num)
```

H.W

Assignment -1

WA numpy program to find common value between two array

```
import numpy as np  
array 1 = np.array([11, 13, 12, 16, 19])  
array 2 = np.array([13, 19, 21, 26])  
common_values = np.intersect1d  
array 1, array 2  
print ("common values : ", common_values)
```

```
arr_num1 = [11, 13, 12, 16, 19]  
arr_num2 = [13, 19, 21, 26]
```

common value = [13, 19]

Numerical Processing

(Numpy Random Number)

Q - what do you mean by random number
 Random Number cannot produce output logically.
 It produce or generate different output in every execution.

Random Number can be produced or generated by algorithm known as Pseudo Random Number or Pseudocode Random Number.

NOTE : using Pseudo Random Number can be utilized in Numpy via module known as Random module.

* Requirement of Random Number or Pseudo Random Number

- 1) Numpy
- 2) Matplotlib lib or Matlab for data visualization and plotting purpose.
- 3) Seaborn plotting library.

* How to install Random Number or Pseudo Terminal →
 pip install numpy
 numpy install Random
 pip install seaborn
 pip install matplotlib lib
 OR
 file → setting → project → python interpret → install numpy and add them.

NOTE :-

1. For using Random number in numpy we must include package.
 - ① from numpy import random
 - ② import numpy as np
2. For using Random number in numpy than must utilised open source library of numpy. For integer random number we can use a function random.randint()

3. For Floating point random number random.rand()
- OR
random.uniform()

Q1. program - write a program in python to implement numpy and generate integer number between 0 to 100.

```
from numpy import random
import numpy as np
arr-num = np.random.randint(100)
```

OR

```
arr-num = np.random.randint(100)
```

```
print("The integer Random Number generation = ")
OR
print(arr-num)
```

It will be produced output at every execution will be unique or different.

It will be produced output at every execution will be unique or different.

Q3. write a python program to implement numpy and generate five (5) integer random number between 0 to 100 using np.

```
from numpy import random  
import numpy as np  
arr-num = random.randint(low=0, high=100, size(5))  
print ("f" "integer Random Number generation  
by the help of ID = arr-num")
```

Q3. WAP in python to implement numpy and generate five (5) integer random number between 0 to 100 using loop.

```
from numpy import random  
import numpy as np  
for i in range (5):  
    OR
```

```
for i in range (5):  
    arr-num= np.random.randint(low=0, high=100)  
    print ("f" "the random integer generation by  
loop = arr-num")
```

Q4. WAP to implement numpy and generate five (5) random number between 20 to 30 using loop.

```
from numpy import random  
import numpy as np  
for i in range (5):  
    arr-num = random.uniform(20, 30):  
    OR  
    arr-num = random.uniform (low : 20, high : 30))  
    print (arr-num)
```

2-D Array

```
from numpy import random
import numpy as np
for i in range(3):
    arrnum = random.rand(2,3)
```

Output -

```
[ [ 0.11371626  0.85316265  0.56680049]
  [ 0.6131036   0.80193664  0.23251893]
  [ 0.42254165  0.91918022  0.48284487]
  [ 0.5835652   0.0468331  0.87474555] ]
```

1-D Array

from numpy import random
import numpy as np
for i in range(5):
 arrnum = random.rand(1)

Value = 0 to 1

Output -

```
0.657520103256245
0.0174
0.4289
0.3683
0.3280
```

Ques. Use algo + import lib (numpy)

17/05/25

Pseudo Random Number

Q1. Write a python program to implement Numerical processing and generate 1-D array with the help of given set of array value.

```
from numpy import random  
import numpy as np
```

```
arr_num = np.array ([1, 2, 3, 4, 5, 6, 7, 8, 9])
```

print "The original array element = arr-num"
~~result = np.random.choice (arr-num, size = 4)~~
~~print "the convolution of 1-D of given~~
~~array array = result"~~

Op :- the original array elst = [1-9]
the convion of 1D of
given array = [2 8 1 7]

Random.choice — with the help of the library we can generate the random number in given set of array.

array () — with the help of this library we can create the array either be vector or matrix.

Q2. Write a python program to implement Numerical processing and generate 2-D array with the help of given set of array

```
import numpy as np  
import random
```

```
arr-num = np.array([[-1, 2, 3],  
[4, 5, 6],  
[7, 8, 9]])
```

```
print("original array =")  
print(arr-num)
```

```
result = np.random.choice(arr-num, size=3)  
print("result =")  
print(result)
```

Random number

- ① Not produced output logically
- ② At every execution it will be produced different result.

It will take two line coding.
① random.seed()

for same set of random no.

random.seed()
It will be implement the seed()

- ④ random.seed()
- seed function() ensures to generate same set of random number at every execution.

Q-> W/NP to implement Numerical python to generate same set of random number

op:- enter the seed value:5

import numpy as np

from numpy import random

arr_num = int(input("Enter the seed value ="))

random.seed(arr_num)

print(random.random(), end = " ")

* Controlling Execution

early static binding → late binding or
Dynamic execution

→ predefined output

logic will be same but
output will be different



Dynamic execution: In python involves running code that constructed or modified during the program runtime. That means in another word we can say that dynamic execution refers to the ability to execute the part that is generated or modified at runtime.

Controlling Execution

19/05/25

Dynamic Execution

Static Execution

1. Modification and program construction will be possible at runtime.

Program construction and modification can't be possible at run time.

2. For using or performing operation in dynamic execution, we can use open source inbuilt library of python.

There are 4 methods can be utilized for dynamic execution in python.

1) eval()

2) exec()

3) import lib

4) Meta programming.

NOTE:

That means we can say that Dynamic execution refers to the ability that is generated or modified or constructed the program code.

2) exec():

It is inbuilt python library that generate the dynamic execution of python code. It takes either string containing code to be execute or python object with optional local and global director

Syntax - exec (object [global, [locals]])

↓
source of the program

exec () function provides powerful library / features that allow for dynamic execution that generate or constructed at run time.

NOTE :

1. We can use exec () for single line statement.
2. we can also use exec () for multiple line statement.

• Example 1 :- (single line statement.)

```
arr_num = input("Enter python program for  
dynamic execution =")  
exec(arr_num)
```

when we will run the program -

Enter the python program :

```
[print(arr_num) for arr-num in range(0,5)]
```

Output :-
0
1
2
3
4