PAGE NO.:

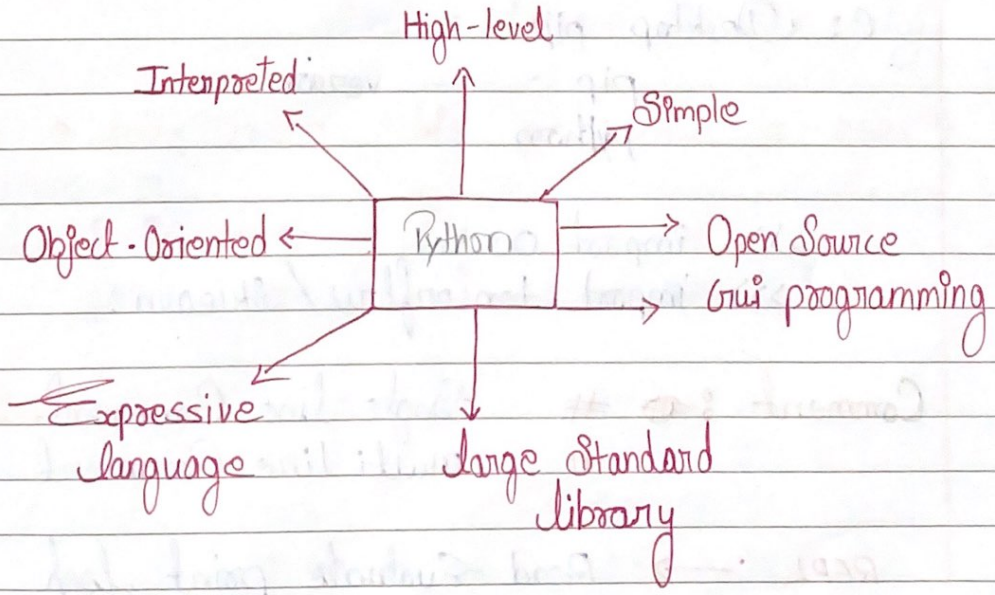
Python :—

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (Server-side),
- software development
- mathematics,
- system scripting

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured, object-oriented and functional programming.



Current version — 3.11

python.org ☑ Add python to my path

CMD

C:> Cash python

>>> exit ()

C:/> python --- version

IDLE (white) Shell (black)

→ Screen ←

Program :- >>> print ("Hello World")

modules

- → Built in (os, abc)
- → paid, user define
- → Extenally import
  - → pleast tenseflow sklearn

C: Desktop pip

pip ---version

python

>>> import os

>>> import tenserflow / sklearn.

Comment :- # Single line Comment

''' Multi line Comment

REPL ⟶ Read Evaluate print loop

Q.1 → Write a pr

Print ("

print ("

Output :-

Q.2 → to Write

num = 9

for 9 in

print

Output

PAGE NO:

```
>>> import os
>>> print ("Hello World")
>>> 3+4+8+9+2.      - 26.
```

Q.1 Write a poem ?

→

```
Print ("""twinkle twinkle little star
How I wonder what you are
up above the woodd so high
like a dimond in the sky """)

print ("twinkle. twinkle little star \n
How i wonder what you are \n")
```

Output :-

twinkle twinkle little star
How i wonder what you are
up above the world so high
like a dimond in the sky

Q.2 Write a 9 table and use to REPL ?

→

```
num = 9
for i in range (1,11):
    print (num,'x',i,'=',num * i)
```

Output

9x1 = 9
9x2 = 18
9x3 = 27
9x4 = 36
9x5 = 45
9x6 = 54
9x7 = 63
9x8 = 72
9x9 = 81
9x10 = 90

Q.3 play sound modul (download) ?

→ from ~~play sound~~ playsound . Import playsound
playsound (

Q.4 Write a python program can to print the content of directory using os moduls ?

Gedit :- print (" Hello World ")
Hello.py Desktop
cd Desktop
python Hello.py

output :- Hello World

a = 2
b = 2

print (a+b)
add.py
cd Desktop
python add.py

Output = 4

print (" . . . . .
. . . . .
. . .

* Variable / Datat

Variable :-

a = 30
b = " MCA "
c = 34.8

Class = 40

→ Variable
in a

Variable :-
Keyword :-
Identifiers

print (a+b)
add.py
Cd Desktop
python add.py

Output = 4

print (""" ..... multiline
.....
..... """)

* Variable / Datatype [Variable = Case sensetive]

Variable :- Keywords
Identifiers

a = 30 Integer
b = " MCA " String
c = 34.8 float

Class = 40 false = " MCA " None = 48.3

→ Variable is the name given to a memory location in a program.

Variable :- Container store a value
Keyword :- reserve words in python
Identifiers :- class, function or variable

Datatype :— 1 Integer
2 float
3 string
4 Boolean
5 None

Eg:— a = 71 — class <int>
b = 8.1 — class <float>
c = "MCA" — class <string>

```
a = 71
print (type (a))
```

Rules for Defing a variable (Also apply to other identify)

1. A variable name can contain alphabates, digits and underscore.
2. A variable name can only start with an alphabet & underscore.
3. A variable name can not start with a digit.
4. No white space is allowed to be used a variable name.

```
a1 = 10
a_ = 15
_a = 11
10 = 10
@ 0
a-1 $ ghi = 10
```

5 Variable are

2. write a prim

```
Num = int (
flag = 0
if num > 1
    for i in
    if (nu
        flag
        brea
if flag:
    prin
else:
    p
```

Output :—

Example

```
abc
a = 12
b = 2.
c = "
c1 = "
c2 = "
d =
N =
```

5 Variable are case sensetive

Q. write a prime Number ?

```
Num = int ( input ("enter a number :"))
flag = 0
  if num > 1 :
     for i in range (2, num):
      if (num % i) == 0:
         flag = 1
         break
  if flag:
        print (num, " is not a prime no. ")
  else :
        print (num, " is a prime no. ")
```

Output :- python prime.py
Enter a number : 1
(1, "is a prime number")
Enter a number : 9
9, " is not a prime number ")

Example of few variable are

```
abc   a-1   -01
a = 12
b = 2.8
C = "Sakshi"
C1 = " Maya'
C2 = "'silky"'
d = True
N = None
```

print (" The value of a is ", a its type is type(a))
print (type (b))
print (type (c))
print (type (c1))
print (type (c2))
print (type (d))
print (type (N))

* Operators in python

1 Arithmatic operator → +, -, *, /, %
2 Assigment operator → ++, --, *=, /=, == etc
3 Comparision operator → ==, >, <, >=, != etc
4 logical operator → AND, OR, NOT

# code for arithmatic operator
a = 20
b = 4
print (a+b)
print (a-b) # or we can write print ("a+b=", a+b)
print (a*b)
print (a/b)
print (a%b)

# Code for assignment operator.
a = 12
b = 8
a += b
print (" value of a = ", a)
a -= b

```
print ("value of a =", a)
a *= b
print ("value of a =", a)
a /= b
print ("value of a =", a)
```

Code for Comparision operator

```
a = 10
b = 20
print ("A is greater no =", a>b)
print ("B is greater no =", a<b)
print ("A is greater than equal to B =", a>=b)
print ("A is less than equal to B =", a<=b)
print ("A is not equal to B =", a!=b)
```

Code for logical operator

```
a = 30
b = 15
print ("AND operator =", a>5 and b<7)
print ("OR operator =", a>5 or b<7)
print ("NOT Operator =", not (b>6 or b>4))
```

Output :-

```
24
16
80
5
0
('Value of a =', 20)
('Value of a =', 12)
('Value of a =', 96)
('Value of a =', 12)
```

('A is greator no = ', false)
('B is greator no = ', True)
('A is greator than equal to B = ', false)
('A is less than equal to B = ', True)
('A is not equal to B = ', True)
(' AND Operator = ', False)
(' OR Operator = ', True)
(' NOT Operator = ', False)

```
a = input ("enter name here")
print (a)
```
enter name here
MCA

Input tag = string value.

Assignment No. 1

1. Write a python program to add two number ?

```
x = 5
y = 10
print (x+y)
```
Output => 15

2. Write a python program to find reminder when a number is divided by 2.

```
a = int (input ("enter a no."))
print ("reminder = ", a%2)
```

Output
enter a no. = 5
('remainder = ', 1)

3. Check the th
function
a = input
print ("
print (typ
c = input
print ("
print (ty
b = input
b = int (
print ("
print (

Output

4. Use Comp
variable
a = 50
b = 65
print

Output

3 Check the thype of variable assigned using input() function.

```
a = input ("Enter name here :")
print ("a=", a)
print (type(a))
c = input ("Enter number")
print ("c=", c)
print (type(c))
b = input ("Enter number")
b= int (b)
print ("b=", b)
print (type (b))
```

Output

```
Enter name here : Saraf
a = Saraf
class < string >
Enter number 5
c=5
class <string>
Enter number
b = 23
Class <int>
```

4 Use Comparision operator to find out whether a given variable 'a' or not. take a = 50 and b = 65.

```
a = 50
b= 65
print ("Greater no. is", a>b)
```

Output Greater no. is 65

5 Write a python program to find average of two no. entered by the user.

→
```
a = int (input (" Enter 1st no. "))
b = int (input (" Enter 2nd no. "))
print (" Average of a & b = ", (a+b)/2)
```

Output → Enter 1st no. 73
Enter 2nd no. 51
Average of a & b = 62

6 Write a python program to calculate square of a number entered by the user.

→
```
a = int (input (" Enter a number "))
print (" square = ", a * a)
```

Output → Enter a number = 5
square = 25

* String

```
a = ' Sakshi '
b = " Sakshi "
c = ''' Sakshi '''
```

```
x = " Good Morning "
y = " mca "
print (x+y)
```

Output - Good Morning mca

String is data-type python is a sequence

we can primarly write a string in the flowing three

a 'Silky' Single ' ' string
b "silky" double " " string
c """Silky""" Triple """ """ string

* String Concatation :— x = "Good Afternoon "
y = "Silky"
print = (x+y)
Output :- Good Afternoon Silky

* String Indexing :— a = "Meera"

0	1	2	3	4
M	E	E	R	A
-5	-4	-3	-2	-1

1 print (a[0:4]) → MEER
2 print (a[1:3]) → EE
3 print (a[:4]) → MEER
4 print (a[0:]) → MEERA
5 print (a[4]) → A
6 print (a[5]) → Error string index out of bound
7 a[4] = "E" STR object does not support item.

a = "Meera"
print (a[-5]) = M # same as a[0]
print (a[-1]) = a # same as a[4]

print (a[-6]) = Error
print (a[-5:-1]) = meer # same as a (0:4)
print (a[:-2]) = mee # same as a (:3)
print (a[1:80]) = eera

* Datetime

```
import time
import datetime
x = datetime.datetime.now()
print (x.strftime("%c"))
```

Output : Mon Dec 5 10:39:25 2022

* Matrix

```
x = [ [1, 2, 3] ,
      [4, 5, 6] ,
      [7, 8, 9] ]

y = [ [10, 11, 12] ,
      [13, 14, 15] ,
      [16, 17, 18] ]

result = [ [0,0,0] ,
           [0,0,0] ,
           [0,0,0] ]
for i in range (len(x)):
  for j in range (len(x[0])):
    result[i][j] = x[i][j] + y[i][j]
for r in result:
    print(r)
```

Output

[11
[17,
[23

* String Slicing

geeks

Consider the

0	1
M	E
-1	-2

The index pr
in python
a followi

S = a [

First index

a [0:3]
a [1:3]

Other adv
a [:
a [1:

Output

[11, 13, 15]
[17, 19, 21]
[23, 25, 27]

* String Slicing :-

A string in python can be slice for geating a part of the string.

Consider the following string

0	1	2	3	4
M	E	E	R	A
-1	-2	-3	-4	-5

The index in a string start from 0 to [length-1] in python in order to slices a string we use a following syntax.

S = a [ind . start . ind . end]

First index included — Last index is not included

a [0:3] = Mee ⟶ Charcter from 0 to 2.
a [1:3] = ee ⟶ Charcter from 1 to 2

Other advance slicing technique :-

a [: 3] = Mee
a [1 :] = eera

PAGE NO

Negative Indexes :—

NI can also be used a show in the figure (1) -1 Crosspond to the (length - 1) inded. -2 Crosspond to the (length - 2) in a (-4 : -1) = eer.

Slicing with skip value :—

We can provider skip value as a part of are slice like this

b = "amazing"
print (b (1:6:2]) = magin
2 = mzn
print (b[1:6:3]) = mi

* String funtion :—

Some of the mostly use function to perform operation on menuplate string are :-

1 len () :—

This function return the leangth of the string.

print (len ("amazon"))
a = "amazon"
print (len (a))

2 String Endswith () :-

variable string c
If string is "
wise false.

a = " once upo
point (a. end

3 String . Count () : -

of any chanct

a = "Once upor
b = "meera"
print (b. count
print (b. Coun

4 String of Capital

first cha

5 String of lowe

Letter of

5 String . Upper ()

of

PAGE NO.

2 String Endswith () :-

This function tells weather the variable string endswith the string "time or not" If string is "time" it return true other wise false.

```
a = "Once upon a time"
print (a.endswith ("time"))
```

3 String.Count () :-

Counts the total number of occurrence of any character of string.

```
a = "Once upon a time upon"
b = "meera"
print (b.count ("e")) = 2
print (a.Count ("upon")) = 2
```

4 String of Capitalize() :-

The function Capitalize the first character of a given string.

5 String of lower () :-

Returns a string with every Letter of original in lower case.

5 String.Upper() :-

Returns a string with every letter of original in upper case.

7. String.find(word):-
This function find the wood and return the index the first occurrence of that woods the string.

8. String.replace(old word, New Word):-
This function replace every old word with new word in the entire string.

9. Escape Sequence charcter:-
Sequence of Charcter after backlash. escape sequence Charcter Comparise of more than one Charcter but represence when use within the ~~esth~~ string.

'\n', '\t' '\\' '\'
newline tab backlash

* Slice String program *

1. Write a python program to display a user enter name followed by the Good afternoon using input ~~below name and date~~ function.

->
```
name = input ("enter your name -")
print ("Good afternoon", name)
```

Output - Enter your name = Sakshi
Good afternoon Sakshi

2 Write a program to fill in letter template given below name and date. Silky

Letter = ''' Dear </Name/>
your are selected !.
</Date/> '''
12/12/22

→ letter = ''' Dear </Name/>
your are Selected !
</Date/> '''
name = input (" enter your name : ")
date = input (" enter date : ")
letter = letter . replace ("</Name/>", name)
letter = letter . replace ("</Date/>", Date)
print (letter)

Output

Enter your name : Sakshi Ranka
Enter date : 12/12/22
Dear Sakshi Ranka
you are selected !
12/12/22

3 WAP to Didect double space in a string.
" Once ---- upon

→ a = " Once upon a time ";
print (a.count (" "))

Output :- 5

4 Replace the double space from problem three with single space.

→
```
p = "Once upon  a  time "
print (p.replace ("  ", "___"))
```

Output Once upon___a___time

5 WAP to formate the following letter using escape sequence character.

letter = "Dear Meera, this is mca class, thanks|"

→
```
print ("Dear sakshi \n\t this is mca class \n\t\t\t
        thanks| ")
```

Output
Dear Meera
This is mca class
Thanks

* List & tuple

List :—

python list are containers to store a set of value of any datatype with in square bracket [].

```
f = [ "Apple" , 8 , false , " MCA " ]
      ↓       ↓     ↓       ↓
    String  Integer Boolean string
```

1 Create the list using square bracket

a = [1, 2, 3, 4, 5]

2 print the List
print (a)

3 Access using
0 1 2
a = [1, 2, 3

A list can be
print (a

Eg :— l [
prin
prin

print
prin
pri

4 Change the

a = "Meera"
a[1] = "o
b = [" l
print (b
b(2) =
print (b

k = ["
k[0] =

problem three with

ne "
!", "____ "))

time

letter using escape

Is mca class, thanks|"
is is mca class \n \t \t
")

lass
Thanks

ers to store a set
with in square

, " MCA "]
↓
string
bracket

2 print the List
print (a)

3 Access using index
0 1 2 3 4
a = [1, 2, 3, 4, 5]

A list can be index just like a string
print (a(2)) = 3
0 1 2
Eg :- l [7, 10, " apple "]
print. (l(2)) = Apple
print (l(40)) = Error list Index out of range.

print (l(0:2)) = 7, 10
print (l(-2)) = 10
print (l(:2)) = 7, 10

4 Change the value of list

a = "Meera"
a[1] = "a" ⟶ Error
b = [" 1, 2, 3, 4, 5, 6, 7 "]
print (b)
b(2) = 98
print (b) ⟶ [1, 2, 98, 4, 5, 6, 7]

K = [" MCA", 1, 2, 3, 4,]
K[0] = 1

* List Method :—
Consider the following list

l = [1, 8, 7, 2, 21, 15]

1 l.sort() :- Update the list to.

l.sort() ⟶ [1, 8, 7, 2, 21, 15]

2 l.reverse() :- update the list to

l.reverse() ⟶ [15, 21, 2, 7, 8, 1]

3 l.append() :-

l.append (40) ⟶ [1, 8, 7, 2, 21, 15, 40]
value

4 l.insert (3, 50) :— [1, 8, 7, 50, 21, 15]
Index

5 l.pop (2) :— [1, 8, 2, 21, 15]
Index

6 l.Remove (15) :— [1, 8, 7, 2, 21]
Value

Tuples & python :—

Create a tuple using
t = (1, 2, 3, 4)
print (t[0])
Output - 1
a() ⟶ empt
a=(1,) ⟶ tuple

Out

Once define a tupl

t (1, 2, 3, 4)
t[0] = 34

* Tuples Meth

→ Consider the t

0 1 2
a = (1, 7, 2

1 a.Count() :—

1 occurr

a.Count(1) =
a.Count(2) =

PAGE NO:

Tuples & python :—

Cannot change

A tuple is a immutable datatype in python.

Create a tuple using ( ) :

```
t = (1,2,3,4)
print (t[0])
```

Output - 1

a () —> Empty tuple

a=(1,) —> tuple ——> a=1 print (type[a])

a=(1,) print (type[a])

output :— <Class 'int'>

<Class 'tuple'>

Once define a tuple element can't be alter or manuplated

```
t (1,2,3,4)
t[0] = 34 . Error!! tuple object does not support
```

* Tuples Method

→ Consider the following tuples.

```
      0 1 2 3 4 5
a = ( 1,7,2,1,2,1 )
```

1 a.Count () :—

A.Count (1) will return no. of times 1 occurer in a

a.Count (1) = 3 a.Count (7) = 1

a.Count (2) = 2

Q a.index() :-

a.index(i) will return the index of first occurrence of i in a

a.index(7) = 1
a.index(1) = 0

dist / Tuple

1 Write a program to store seven fruits in a list entered by the user ?

→ F1 = input("Enter fruit Number 1 : ")
F2 = input("Enter fruit Number 2 : ")
F3 = input("Enter fruit Number 3 : ")
F4 = input("Enter fruit Number 4 : ")
F5 = input("Enter fruit Number 5 : ")
F6 = input("Enter fruit Number 6 : ")
F7 = input("Enter fruit Number 7 : ")
Myfruitlist = [F1, F2, F3, F4, F5, F6, F7]
print(myfruitlist)

Output :-

Enter fruit Number 1 : Pineapple
Enter fruit Number 2 : Watermelon
Enter fruit Number 3 : Green Apple
Enter fruit Number 4 : Orange
Enter fruit Number 5 : Mango
Enter fruit Number 6 : Grapes
Enter fruit Number 7 : Pear
['Pineapple', 'Watermelon', 'Green Apple', 'Orange', 'Mango', 'Grapes', 'Pear']

2 WAP to accept marks them and sort

→ M1 = int(input("Ma
M2 = int(input("Ma
M3 = int(input("Ma
M4 = int(input("Ma
M5 = int(input("Ma
M6 = int(input("Ma
Sortedmarks = [M1,
Sortedmarks.sort
print(Sortedmarks

Output :-

Marks of
Marks of
Marks of
Marks of
Marks of
Marks of
[56, 65,

3 Check that a tup