

Q3 (i) Features of Python -

- ① Easy to learn, write and more readable.
- ② It is interpreted not compiled.
- ③ It is open-source language under GNU public License
- ④ Python program → Bytecodes → Python virtual machine → machine language
Python is platform independent.

Scanned with CamScanner

papergrid

Date: / /

Byte codes - Some low level language to be understood by some virtual machine

- ⑤ Free
- ⑥ Variable declaration do not require type.

Scanned with CamScanner

(ii)

(a.)

$$5 \% 10 \text{ and } -25 > 1 * 8 / 15$$

Ans. False

(b.)

$$(7 - 4 * 2) * 10 / 5 + 2 + 15$$

Ans. 19.6

(c.)

$$7 * 2 / 19 \% 3$$

Ans 2



Scanned with CamScanner

Q4.

Q1. Python Strings :-

String is a collection of characters, in python string are immutable, i.e. we cannot change a string.

Defination / declaration :- we can use 'abc', "abc", abc to represent string object.

Eg:-

```
>>> str='abc def'
```

```
>>> str
```

'abc def'

String Indexing :- In python we can use positive indexes as well as negative indexes to access individual character of string.

0	1	2	3	4	5	6
a	b	c	d	e	f	

→ -6 -5 -4 -3 -2 -1



Example :-

>>> str[0]

a

>>> str[-7]

>>> len(str) # to display length
of string

7

String Slicing :- To display only
part of string, we
can use slicing.

Syntax:-

>>> str[< starting Index > : < one more
than the ending Index >]

Ex:- for printing def we can use the
below slicing.

>>> str[4:7]

def

>>> str[1:7]

abc def

>>> str[2:]

c def



Using + operator for string concatenation

>>> "abc" + "def"

O/P abcdef

Using * operator to repeat a string

>>> '*' * 5

Q2. Lists:- A list is a heterogeneous collection of python objects, it can also be index, sliced like string, but list are Mutable (i.e. we can change elements of list). It is represented in square brackets.

```
>>> list1 = [1, 2, 3, 4, 4]
>>> list2 = ['a', 'b', 1, 2, 4.5,
           True]
>>> list1[0] = 10
>>> list1
[10, 2, 3, 4, 4]
```

positive	0	1	2	3	4	+ve index
Index	1	2	3	4	4	last index
Negative	-5	-4	-3	-2	-1	
Index						do " "

```
>>> list1[1:]
[2, 3, 4, 4]
```

```
>>> len(list1)
```

5



#concatenate 2 list.

>>> L₁ = [1, 2, 3]

>>> L₂ = [3, 2, 1]

>>> L₁ + L₂

[1, 2, 3, 3, 2, 1]

Removing elements from list

>>> L = [1, 2, 3, 4]

>>> L.pop() # pops the last

element

>>> L

[1, 2, 3]

>>> L.pop[2] # Pop element

at index 1.

2

>>> L

[1, 3]

Remove function :- Removes the

first instance of the
element from list.

>>> t = [1, 2, 1, 4, 5, 1]

```
>>> L2.remove(1)
```

```
>>> L2  
[2, 1, 4, 5, 1]
```

- count all instances of element

```
>>> L2.count(1) # count instances  
2
```

- delete a list

```
>>> del L2
```

- Add elements in lists (add at end)

```
>>> L3 = [1, 2, 3]
```

```
>>> L3.append(4)
```

```
>>> L3  
[1, 2, 3, 4]
```

To copy a list

```
>>> L4 = L3.copy()
```

List can contain other lists

>>> L₁ = [1, 2, 3]

>>> L₂ = [4, 5, 6]

>>> L₃ = [L₁, L₂]

>>> L₃

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

>>> len(L₃)

2

>>> L₃[0]

[1, 2, 3]

>>> L₃[1]

[4, 5, 6]

>>> L₃[0][1]

2

>>> L₃.append("abc")

>>> L₃

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

>>> L₃[2][1]

'b'