

\* Class Assessment 1

(28/30)

marks

Q 1] Explain the difference between list & tuple in python  
Provide an example for each.

→ Lists:-

- 1] Lists are mutable - meaning you can modify their <sup>content</sup>.
- 2] They are defined using square brackets [ ]
- 3] Lists are generally used for collections of items where the order & the ability to change the element are important.

Example :-

L = ["Hello", 1, 2, 3, 4, "World", 5]

Tuple :-

- ①
- 1] Tuples are immutable - you cannot change their content.
  - 2] Tuples are suitable for fixed collection where the order of elements matters but you don't want the data to be modified.

Example :-

T = (1, 2, 3, "Apple", 5, "Banana")

Q 2] Describe the purpose of the set data type in python. Provide an example to illustrate its use.

→

Purpose :- 1] It is used to represent an unordered collection of unique elements

2] Primary purpose of a set is to perform mathematical set operation like union, intersection, difference & symmetric difference efficiently

3] Sets are mutable

e.g.  $a = \{1, 2, 3, 4, 5\}$

$b = \{3, 4, 5, 6, 7\}$

a. add (8)

b. remove (7)

sets can be used for tasks such as combining, finding common elements identifying difference & more

Q 3] What is the key difference between a float & an integer data type in python? Give an example where using a float would be more appropriate.

→ 'Integer ("int") :- Represent whole number without decimal point.

e.g. :- 1, 5, 100

• Float :- Represent number with decimal points or number expressed in exponential form

e.g. :- 3.14, 0.5, 2e3

Example

Distance = 10

Time = 2.5

Speed = distance / time

Point = ("speed", speed, "m/s")

Q 4] How does dictionary data type in python differ from lists & tuples. Provides an example of dictionary & explain its structure.

- i) The difference dictionary data type in python differs from list & tuples in that it is an unordered collection of key value pairs, where each key must be unique.
- ii) Dictionaries are defined must be using curly bracket {}.
- iii) It useful for storing & retrieving data in a way that is easily accessible & efficient

For example :-

Personal info = {"Name": "Ruchika", "Age": 21}

Structure :-

- i) Personal info has key such as Name, Age the values associated with type keys provide information about the personal (e.g Ruchika, 21)
- ii) Keys & values are separated by colons (':') with each key-value pair.

Q 5] What is doc string & use of this string in python

→ A doc string in python is string literal used as a comment at the beginning of module, Function, class or method definition. Its primary purpose is to provide documentation or information about the purpose & usage of the code.

\* Use :-

- i) Documentation
- ii) Accessibility
- iii) Interactive Development.

Q 6] Explain the purpose of the // operator in python. Provide example to illustrate its use.

→ // :- Its purpose is the Floors division operator. It performs division & return the largest integer that is less than or equal to the result. It discards the fractional part of the division result & produces an integer quotient.

e.g i— 5 = 10 // 3

Print (5)

In e.g '5' assigned the value of 3 because the floor division of 10 by 3 is 3 with remainder 1.

Q.7] Differentiate bet<sup>n</sup> the == & is operator in python provides examples demonstrate their usages.

== Operator

1) The '==' operator checks the values of two operands use equal

2) It compares the content of the object

a = [1, 2, 3]

b = [1, 2, 3]

result = a == b

Print (result)

a == b return 'True'

is operator

1) The 'is' operator checks if the operators refer to the same object in memory

2) It reset for object identity not just equalify the content

a = [1, 2, 3]

y = [1, 2, 3]

result = a is y

Print (result)

a is y return False

Q.8] What is the use of += operator in python? Provide an example to demonstrate its functionality

→ The '+=' operator in python is used an assignment operator & it is a shorthand for updating the value of variable by adding another value to it. it commonly used for in place addition & assignment.

Count = 5

Count += 2

Print (count)

Output = 7

'+=' operator add the value '2' to variable count

Q9] Discuss the role of the 'in' operator in python. Provide an example of how it can be used.

- i) The 'in' operator in python is used to test whether a specified value is present in a sequence, such as a string, list, tuple or dictionary.
- ii) It returns the Boolean value (True & False) based on the existence of the specified element in the sequence.

e.g message = "Good morning Shravani"

s = "Shravani" in message

Point (s)

Output: True

because Shravani is present in string message

Q10] Explain the concept of the ternary operator ( $x$  if condition else  $y$ ) in python. Provide example scenarios where it can be employed.

- i) The ternary operator in python, also known as the conditional expression, provides a concise way to write a simple 'if-else' statement in a single line.
- ii) Its syntax is  $x$  if condition else  $y$ . The expression evaluates to ' $x$ ' if condition is true otherwise evaluate to ' $y$ '.

e.g.

a = 10

b = 15

IF a > b;

Print, (large number = a)

else :

Print (larger number = b)

Q11 What is the purpose of the if statement in Python? Provide an example demonstrating the use of an if statement

→ The 'if' statement in python is used for conditional execution code. It allows to specify a block of code to be executed only if a certain condition is true. If the condition is false, the code block is skipped or an alternative block may be executed if an 'else' clause is present.

For example:-

```
Num = int (input ("Enter a number"))
```

```
| 112 / if number > 0 :
```

```
    print ("The entered num is Positive)
```

```
elif number < 0 :
```

```
    print ("The entered num is negative)
```

```
else :
```

```
    print ("The entered num is zero)
```

Q12 Describe the difference bet<sup>⑩</sup> while & for loop in python  
Give an example for each loop system

→

While loop

i) The 'while' loop in python is used to repeatedly execute a block of code as long as specified condition is true

ii) It keeps iterating until the condition becomes False

e.g:

```
count = 1
while count <= 5 :
    print (count)
```

For loop

i) The 'for' loop in python is used to iterate over a sequence or either iterable object

ii) It executes a block of code for each element <sup>in a</sup> sequence e.g.

```
colors = ["red", "blue"]
```

```
for color in colors:
    print (color)
```

Explain the significance of the break statement in python  
provide a scenario where using break is appropriate  
The 'break' statement in python is used to exit loop  
prematurely, before its normal termination condition  
is met, when encountered 'break' immediately terminates  
the loop & the program continues with the next statement  
after the loop

e.g.

```
input = input ("Enter 'exit' to stop the loop:")
if input.lower() == 'exit':
    print ("Exiting the loop")
    break
print ("Continuing the loop...")
```

Q 14] Discuss the role of the continue statement in python  
provide a code snippet demonstrating its use

→ i) The 'continue' statement in python is used to skip  
the rest of the code inside loop for the current iteration  
& proceed to the next iteration. It is particularly  
useful when you want to bypass certain condition or  
code block without permanently terminating the entire  
loop.

e.g.

```
for num in range(1, 11):
    if num % 2 == 0:
        print ("Skipping even num: (Num)")
        continue
    print ("Processing odd num: (Num)")
```

- 15] How does the else clause in loop contribute to control flow in python? Provide an example illustrating the use of the else clause in a loop
- 1) The 'else' clause in a loop in python is executed when the loop condition becomes false. It is not executed if the loop is terminated by a 'break' statement.
- 2) The else clause in a loop provides a way to execute a block of code after the loop has completed its normal iteration

e.g.

For num in range(2,11):

    for i in range(2, int(num\*\*0.5)+1):

        if num % i == 0:

            print ("Num is not a Prime Number")

            break

    else :

        print ("Num is a Prime number")