Name :- Pratik Satish More　　Batch No :- D[illegible]9
College :- D.A.C.O.E. Karad　　22/20
Date :- 27/12/2023

Page No.
Date

Class assisment-1

Q1] Explain the difference between a list and a tuple in python. Provide an example for each.

List	Tuple
(i) Lists are mutable, meaning their elements can be modified after creation	(i) Tuple are immutable, meaning their elements cannot be changed.
(ii) Empty list can be created by using empty square bracket.	(ii) Empty tuple can be created by using empty simple bracket.
(iii) Ex. a = [1, 2, 3, 'Mango']	(iii) Ex. a = (1, 2, 3, 'Orange')

2

Q2] Describe the purpose of the set data type in python. Provide an example to illustrate its use.

→ The set data type in python is used to store an unordered collection of unique elements. Sets are particularly useful when you need to eliminate duplicate values and perform mathematical set operations like union, intersection and difference efficiently.

Example

```
set 1 = {'apple', 'orange', 'banana', 'apple'}
set 2 = {'Orange', 'kiwi', 'grape'}
Fruits = set 1 union set 2
Print ("Fruits": , Fruits)
```

9

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

→ The key difference between a float and an integer data type in python is the integers represent whole numbers without any decimal points, while floats represents number with decimal points or numbers in exponential form.

Example.

```
my - integer = 5
my_ float = 3.14
# using a float for precision
 radius = 4.5
area = 3.14 * radius ** 2
print ("Area of the circle:" , area)
```

Q 4] How does the dictonary data type in python differ from lists and tuples ? Provide an example of a dictonary and explain its structure

→ Lists are ordered and mutable and the Tuples are ordered and immutable but the Dictionaries are unordered and mutable

Example

```
Person info = {
      'name: John',
         'age': 30,
      'city' : New York'
}
Point ("Name":, Personal- info ['name']
```

Page No. Date

```
Print ("Age:", personal-info['age'] )
Print("City:", personal-info['city'])
```

Q5] What is doc string and use of this string in pythone.

→ A docstring (document string) in python is a string literal used to provide documentation for a module, class, function or method. It is placed at the beginning of a code block and is enclosed in triple quotes (single or double). Docstrings serve as documentation and can be accessed using the 'help()' function or tools like sphinx for generating documentation.

Q6] Explain the purpose of the // operator in Python. Provide an example to illustrate its use.

→ In Python the '//' operator is the floor division operator. It performs division between two numbers and returns the largest integer less than or equal to the result. It essentially discards the decimal part of the division, providing the floor value.

Example

```
result = 10 // 3
print (result)
```

Q7] Differentiate between the == and is operators in python. Provide examples to demostrate their usage.

→ In python, the '==' operator is used for value equality, comparing the values of objects while the 'is' operator is used for identity comparison checking if two objects refer to the same memory location.

Example.

Using '=='

```
a= [1,2, 3]
b = [1,2,3]
c = a


Print (a==b)     # True
Print (a is b)   # False
```

Q8) What is the use of the += operator is python Provide an example to demonstrate its functionality

→ In Python, the '+=' operator is an assigonment operator used to add the right operand to the left operand and then assign the result to the left operand. It is a shorthand for performing addition and assignment in a single step.

Example.

```
count = 10
count += 3
print (count)
```

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

→ In Python the 'in' operator is used to test if a value exists within a sequence such as a string, list, tuple or any other iterable It returns a Boolean value, 'True' if the specified element is found in the sequence and 'False' otherwise.

Example.

```
Fruits =['apple', 'banana', 'orange', 'grape']
banana = 'banana' in fruits
print (banana)
  # True.
```

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

→ The ternary operator, also known as the conditional expression is a concise way to express a conditional statement in a single line. It has the following syntax: 'x if condition else y'. It evaluates the condition and if it is 'True', it returns the value of 'x'; otherwise, it returns the value of 'y'.

7

Example:

```
number = 7
result = "Even" if number % 2 == 0 else "odd"
print (result)

  odd
```

Q.12] What is the difference between while and for loops in python. Give an example for each loop type.

→ While loop:

A 'while' loop repeatedly executes a block of code as long as a specified condition is 'True'. The loop continues untile the condition becomes 'False'.

Example.

```
counter = 1
while counter <= 5:
        print (counter)
        counter += 1
```

Ans 1, 2, 3, 4, 5

For loop:

A 'For' loop is used for iterating over a sequence (such as a list, tuple, string or range) or other iterable objects. It iterates over each item in the sequence.

Example

```
Fruits=['apple', 'banana', 'orange']
for Fruit in fruits:
        print (Fruit)
```

Q 11] What is the purpose of the if statement in python? Provide an example demonstrating the use of an if statement.

Page No.
Date

→ The 'if' statement in Python is used to execute a block of code only if a specified condition is 'True'. It allows for conditional branching, enabling different parts of the code to be executed based on whether a particular condition is met.

Example

```
 number = -7
if  number > 0 :
    print ("Tebe number is positive")
elif  number = = 0 :
     print ("The number is zero.")
else:
    print ("The number is negative.")
```

2

Q13 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 or break out of a loop prematurely. It is typically used ~~in~~ within loop.

```
numbers = [1, 3, 5, 8, 9, 10, 12, 15]
For num  in  numbers :
     if  num % 2 == 0 :
         print (F"The first even number in the
                   list is : {num}")
         break.
```

2

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

→ The 'continue' statement in python is used to skip the rest of the code inside a loop for the current iteration and move on to the next iteration

```
numbers=[1,2,3,4,5,6,7,8,9]
for num in numbers:
    if num % 2 != 0:
       continue.
       print (F" Even number : {num}")
Even number = 2 , 4 , 6 , 8
```

Q15] How does the else ~~ed~~ clause in a loop contribute to the flow in Python ? Provide an example illustrating the use of the else clause in a loop.

→ In Python loops the 'else' clause is executed when the loop condition becomes 'False'. However if the loop is terminated by a 'break' statement the 'else' clause is skipped.

```
 number = 11
 for i in range (2, int(number** 0.5 )+1):
     if number % i == 0:
          print (F"{number} is not a prime number!")
          break
else:
     print (F"{number} is a prime number.")
```