**SCIFOR technologies assignment-1**

PYTHON

Datatypes

Question-Difference between List ,Tuple,Dictionary and Set

List :-

* Allows duplicate elements
* Changeable
* Ordered
* Symbole- [ ]
* Example- Mylist=[1,2,3]

print(Mylist)

Tuple :-

* Allows duplicate elements
* Unchangeable
* Ordered
* Symbol- ( )
* Example- t=(1,2,3)

print(t)

Dictionary :-

* No duplicate elements
* Changeable
* Unordered
* Symbole- { }
* Dictionaries are used to store data values in key:value pairs.
* Example- dict={"brand":"Ford","model":"Mustang","year":1964}

print(dict)

Set :-

* No duplicate elements
* Unchangeable
* Unordered
* Symbole- { }
* Example-set={"apple","banana","cherry"}

print(set)

Question- Difference between set and Frozenset

| SET | FROZENSET |
| --- | --- |
| 1. Mutable | 1. Immutable |
| 1. set( ) | 2. frozenset( ) |
| 1. Ordered | 3. Unordered |
| 1. Unique elements | 4. Unindexed collection of unique  elements |

Example of set:-

set={"apple","banana","cherry"}

print(set)

Example of frozenset:-

myFrozen = frozenset([1,2,3])

print(myFrozen)

Question- What are Decision-Making statements ?

## **if Statement**

| Statementment | Discription |
| --- | --- |
| if Statements | It consists of a Boolean expression which results are either TRUE or FALSE, followed by one or more statements. |
| if else Statements | It also contains a Boolean expression. The if the statement is followed by an optional else statement & if the expression results in FALSE, then else statement gets executed. It is also called alternative execution in which there are two possibilities of the condition determined in which any one of them will get executed. |
| Nested Statements | We can implement if statement and or if-else statement inside another if or if - else statement. Here more than one if conditions are applied & there can be more than one if within elif. |

Example:a = 15

if a > 10:

print("a is greater")

Output:a is greater

## **if else Statements**

Example:

a = 15

b = 20

if a > b:

print("a is greater")

else:

print("b is greater")

Output:b is greater

## **elif Statements**

elif - is a keyword used in Python replacement of else if to place another condition in the program. This is called chained conditional.

Example:

a = 15

b = 15

if a > b:

print("a is greater")

elif a == b:

print("both are equal")

else:

print("b is greater")

Output:

both are equal

We can write if statements in both ways, within parenthesis or without parenthesis

| Loop | Description |
| --- | --- |
| for Loop | This is traditionally used when programmers had a piece of code and wanted to repeat that 'n' number of times. |
| while Loop | The loop gets repeated until the specific Boolean condition is met. |
| Nested Loops | Programmers can use one loop inside another; i.e., they can use for loop inside while or vice - versa or for loop inside for loop or while inside while. |

## **for Loop**

##### **Example 01:**

for x in range (0,3) :

print ('Loop execution %d' % (x))

Output:

Loop execution 0

Loop execution 1

Loop execution 2

## **while Loop**

Example:

count =1

while count < 6 :

print (count)

count+=1

Output:

1

2

3

4

5

## **Nested Loops**

Example:

for g in range(1, 6):

for k in range(1, 3):

print ("%d \* %d = %d" % ( g, k, g\*k))

Output:

1 \* 1 = 1

1 \* 2 = 2

2 \* 1 = 2

2 \* 2 = 4

3 \* 1 = 3

3 \* 2 = 6

4 \* 1 = 4

4 \* 2 = 8

5 \* 1 = 5

5 \* 2 = 10

## **Loop Control Statements**

These statements are used to change execution from its normal sequence.

Python supports three types of loop control statements:

Python Loop Control Statements

| Control Statements | Description |
| --- | --- |
| Break statement | It is used to exit a while loop or a for a loop. It terminates the looping & transfers execution to the statement next to the loop. |
| Continue statement | It causes the looping to skip the rest part of its body & start re-testing its condition. |
| Pass statement | It is used in Python to when a statement is required syntactically, and the programmer does not want to execute any code block or command. |

### **Break statement**

Example:

count = 0

while count <= 100:

print (count)

count += 1

if count >= 3:

break

Output:

0

1

2

### **Continue statement**

Example:

for x in range(10):

if x % 2 == 0:

continue

print (x)

Output:

1

3

5

7

9

### **Pass Statement**

Example:

for letter in 'TutorialsCloud':

if letter == 'C':

pass

print ('Pass block')

print ('Current letter is:', letter)

Output:

Current letter is : T

Current letter is : u

Current letter is : t

Current letter is : o

Current letter is : r

Current letter is : i

Current letter is : a

Current letter is : l

Current letter is : s

Pass block

Current letter is : C

Current letter is : l

Current letter is : o

Current letter is : u

Current letter is : d

# Question:-Python Lambda

# A lambda function is a small anonymous function.

A lambda function can take any number of arguments, but can only have one expression.

The power of lambda is better shown when you use them as a function inside another function.

Syntax:- lambda arguments : expression

Example:-Add 10 to argument a, and return the result:

x = lambda a : a + 10

print(x(5))