Q:What is List? How will you reverse a list?

A list is a data structure in Python that is a mutable, or changeable, ordered sequence of elements. Each element or value that is inside of a list is called an item. Just as strings are defined as characters between quotes, lists are defined by having values between square brackets [ ]

reverse() Method. Every list in Python has a built-in reverse() method you can call to reverse the contents of the list object in-place. Reversing the list in-place means won't create a new list and copy the existing elements to it in reverse order.

Q:How will you remove last object from a list?

pop() You can use list. pop() method to remove the last element from the list. pop will raise index error if the list is empty.

Q:Suppose list1 is [2, 33, 222, 14, and 25], what is list1 [-1]?

Suppose list1 is [2, 33, 222, 14, 25], What is list1[-1]? Explanation: -1 corresponds to the last index in the list. 10

Q:Differentiate between append () and extend () methods?

Python append() method adds an element to a list, and the extend() method concatenates the first list with another list (or another iterable). When append() method adds its argument as a single element to the end of a list, the length of the list itself will increase by one.

Q:Write a Python function to get the largest number, smallest num and sum of all from a list.

```
lst = []
```

```
num = int(input('How many numbers: '))
for n in range(num):
    numbers = int(input('Enter number '))
    lst.append(numbers)
print("Maximum element in the list is :", max(lst), "\nMinimum
element in the list is :", min(lst)
```

Q:How will you compare two lists?

The list.sort() method sorts the two lists and the == operator compares the two lists item by item which means they have equal data items at equal positions. This checks if the list contains equal data item values but it does not take into account the order of elements in the list. This means that the list [1,2,3] will be equal to the list [2,1,3] according to this method of comparison.

Q:Write a Python program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings.

```
def match_words(words):
  ctr = 0

  for word in words:
    if len(word) > 1 and word[0] == word[-1]:
      ctr += 1
  return ctr

print(match_words(['abc', 'xyz', 'aba', '1221']))
```

Q:Write a Python program to remove duplicates from a list.

```
a = [10,20,30,20,10,50,60,40,80,50,40]
```

```
dup_items = set()
uniq_items = []
for x in a:
    if x not in dup_items:
        uniq_items.append(x)
        dup_items.add(x)

print(dup_items)
```

Q:Write a Python program to check a list is empty or not.

```
my_list = []
if not my_list:
    print("the list is empty")
```

Q:Write a Python function that takes two lists and returns true if they have at least one common member.

```
def common_data(list1, list2):
     result = False
     for x in list1:
         for y in list2:
             if x == y:
                 result = True
                 return result
print(common_data([1,2,3,4,5], [5,6,7,8,9]))
print(common_data([1,2,3,4,5], [6,7,8,9]))
```

Q:Write a Python function that takes two lists and returns true if they have at least one common member.

```
def common_data(list1, list2):
     result = False
     for x in list1:
        for y in list2:
            if x == y:
                result = True
                return result
print(common_data([1,2,3,4,5], [5,6,7,8,9]))
print(common_data([1,2,3,4,5], [6,7,8,9]))
```

Q:Write a Python program to generate and print a list of first and last 5 elements where the values are square of numbers between 1 and 30.

Q:Write a Python program to generate and print a list of first and last 5 elements where the values are square of numbers between 1 and 30.

```
def printValues():
    l = list()
    for i in range(1,21):
        l.append(i**2)
    print(l[:5])
    print(l[-5:])

printValues()
```

Q:Write a Python function that takes a list and returns a new list with unique elements of the first list.

```
def unique_list(l):
  x = []
  for a in l:
    if a not in x:
      x.append(a)
  return x

print(unique_list([1,2,3,3,3,3,4,5]))
```

Q:Write a Python program to convert a list of characters into a string.

```
s = ['a', 'b', 'c', 'd']
str1 = ''.join(s)
print(str1)
```

Q:Write a Python program to select an item randomly from a list.

```
import random
color_list = ['Red', 'Blue', 'Green', 'White', 'Black']
print(random.choice(color_list))
```

Q:Write a Python program to find the second smallest number in a list.

```
nums = []
print("Enter 10 Numbers (Elements) for List: ")
for i in range(10):
  nums.append(int(input()))

small = nums[0]
for i in range(10):
  if small>nums[i]:
    small = nums[i]

secondSmall = nums[0]
for i in range(10):
  if secondSmall>nums[i]:
    if nums[i]!=small:
      secondSmall=nums[i]

print("\nSecond Smallest Number is: ")
print(secondSmall)
```

Q:Write a Python program to get unique values from a list

```
my_list = [10, 20, 30, 40, 20, 50, 60, 40]
print("Original List : ",my_list)
my_set = set(my_list)
my_new_list = list(my_set)
print("List of unique numbers : ",my_new_list)
```

Q:Write a Python program to check whether a list contains a sub list

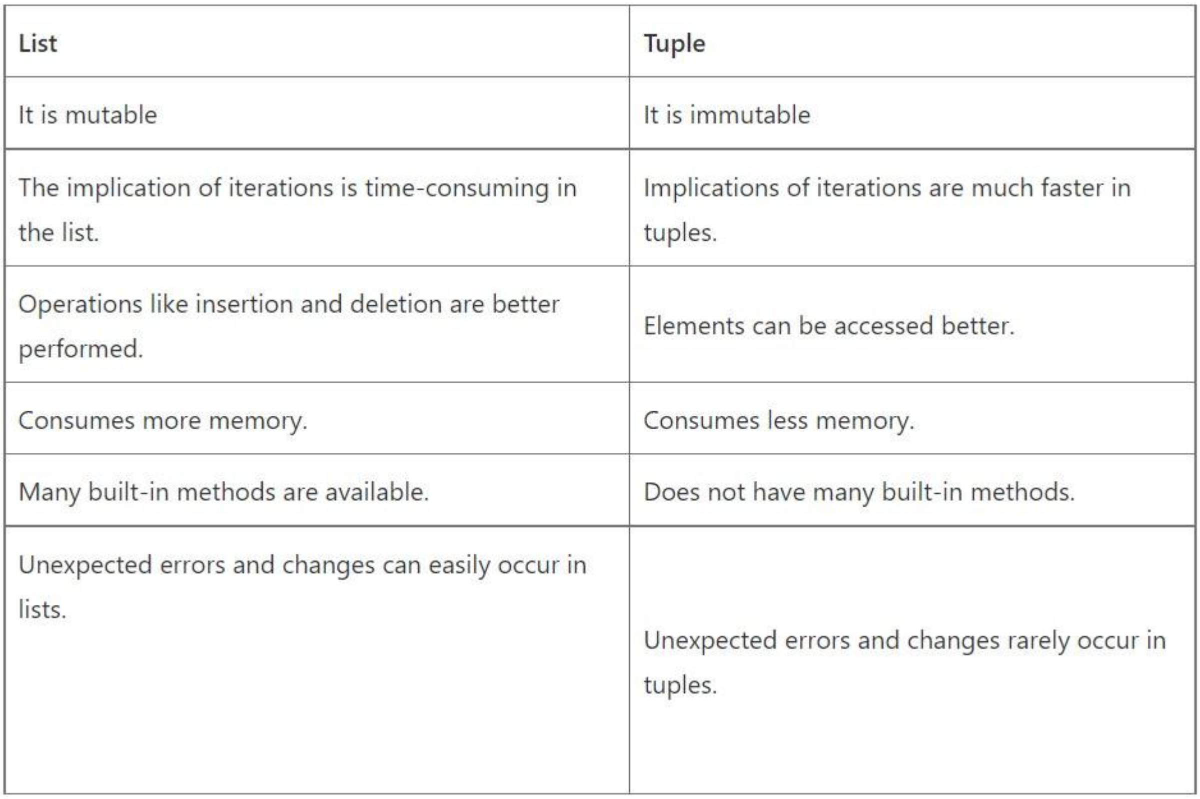
```
def is_Sublist(l, s):
        sub_set = False
        if s == []:
                sub_set = True
        elif s == l:
                sub_set = True
        elif len(s) > len(l):
                sub_set = False

        else:
                for i in range(len(l)):
                        if l[i] == s[0]:
                                n = 1
                                while (n < len(s)) and (l[i+n] == s[n]):
                                        n += 1

                                if n == len(s):
                                        sub_set = True

        return sub_set

a = [2,4,3,5,7]
```

```
b = [4,3]
c = [3,7]
print(is_Sublist(a, b))
print(is_Sublist(a, c))
```

Q:Write a Python program to split a list into different variables.

```
color = [("Black", "#000000", "rgb(0, 0, 0)"), ("Red", "#FF0000",
"rgb(255, 0, 0)"),
        ("Yellow", "#FFFF00", "rgb(255, 255, 0)")]
var1, var2, var3 = color
print(var1)
print(var2)
print(var3)
```

Q:What is tuple? Difference between list and tuple.



List	Tuple
It is mutable	It is immutable
The implication of iterations is time-consuming in the list.	Implications of iterations are much faster in tuples.
Operations like insertion and deletion are better performed.	Elements can be accessed better.
Consumes more memory.	Consumes less memory.
Many built-in methods are available.	Does not have many built-in methods.
Unexpected errors and changes can easily occur in lists.	Unexpected errors and changes rarely occur in tuples.

Q:Write a Python program to create a tuple with different data types.

```
#Create a tuple with different data types
tuplex = ("tuple", False, 3.2, 1)
print(tuplex)
```

Q:Write a Python program to create a tuple with numbers.

```
#Create a tuple with numbers
tuplex = 5, 10, 15, 20, 25
print(tuplex)
#Create a tuple of one item
tuplex = 5,
print(tuplex)
```

Q:Write a Python program to convert a tuple to a string.

```
tup = ('e', 'x', 'e', 'r', 'c', 'i', 's', 'e', 's')
str =  ''.join(tup)
print(str)
```

Q:Write a Python program to check whether an element exists within a tuple.

```
tuplex = ("w", 3, "r", "e", "s", "o", "u", "r", "c", "e")
print("r" in tuplex)
print(5 in tuplex)
```

Q:Write a Python program to find the length of a tuple.

```
#create a tuple
tuplex = tuple("w3resource")
print(tuplex)
#use the len() function to known the length of tuple
```

```
print(len(tuplex))
```

Q:Write a Python program to convert a list to a tuple.

```
#Convert list to tuple
listx = [5, 10, 7, 4, 15, 3]
print(listx)
#use the tuple() function built-in Python, passing as parameter the list
tuplex = tuple(listx)
print(tuplex)
```

Q:Write a Python program to reverse a tuple.

```
#create a tuple
x = ("w3resource")
# Reversed the tuple
y = reversed(x)
print(tuple(y))
#create another tuple
x = (5, 10, 15, 20)
# Reversed the tuple
y = reversed(x)
print(tuple(y))
```

Q:Write a Python program to replace last value of tuples in a list.

```
l = [(10, 20, 40), (40, 50, 60), (70, 80, 90)]
print([t[:-1] + (100,) for t in l])
```

Q:Write a Python program to find the repeated items of a tuple.

```
#create a tuple
tuplex = 2, 4, 5, 6, 2, 3, 4, 4, 7
print(tuplex)
```

```
#return the number of times it appears in the tuple.
count = tuplex.count(4)
print(count)
```

Q:Write a Python program to remove an empty tuple(s) from a list of tuples.

```
L = [(), (), (",), ('a', 'b'), ('a', 'b', 'c'), ('d')]
L = [t for t in L if t]
print(L)
```

Q:Write a Python program to unzip a list of tuples into individual lists.

```
#create a tuple
l = [(1,2), (3,4), (8,9)]
print(list(zip(*l)))
```

Q:Write a Python program to convert a list of tuples into a dictionary.

```
#create a list
l = [("x", 1), ("x", 2), ("x", 3), ("y", 1), ("y", 2), ("z", 1)]
d = {}
for a, b in l:
    d.setdefault(a, []).append(b)
print (d)
```

Q:Write a Python script to sort (ascending and descending) a dictionary by value.

```
import operator
d = {1: 2, 3: 4, 4: 3, 2: 1, 0: 0}
print('Original dictionary : ',d)
sorted_d = sorted(d.items(), key=operator.itemgetter(1))
```

```
print('Dictionary in ascending order by value : ',sorted_d)
sorted_d = dict( sorted(d.items(),
key=operator.itemgetter(1),reverse=True))
print('Dictionary in descending order by value : ',sorted_d)
```

Q:Write a Python script to concatenate following dictionaries to create a new one.

```
dic1={1:10, 2:20}
dic2={3:30, 4:40}
dic3={5:50,6:60}
dic4 = {}
for d in (dic1, dic2, dic3): dic4.update(d)
print(dic4)
```

Q:Write a Python script to check if a given key already exists in a dictionary.

```
d = {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
def is_key_present(x):
  if x in d:
      print('Key is present in the dictionary')
  else:
      print('Key is not present in the dictionary')
is_key_present(5)
is_key_present(9)
```

Q:How Do You Traverse Through A Dictionary Object In Python?

There are two ways of iterating through a Python dictionary object. One is to fetch associated value for each key in keys() list. There is also items() method of dictionary object which returns list of tuples, each tuple having key and value.

Q:How Do You Check The Presence Of A Key In A Dictionary?

given key already exists in a dictionary. has_key() method returns true if a given key is available in the dictionary, otherwise it returns a false. With the Inbuilt method has_key() , use if statement to check if the key is present in the dictionary or not

Q:Write a Python script to print a dictionary where the keys are numbers between 1 and 15.

```
d=dict()
for x in range(1,16):
    d[x]=x**2
print(d)
```

Q:Write a Python program to check multiple keys exists in a dictionary

```
student = {
  'name': 'Alex',
  'class': 'V',
  'roll_id': '2'
}
print(student.keys() >= {'class', 'name'})
print(student.keys() >= {'name', 'Alex'})
print(student.keys() >= {'roll_id', 'name'})
```

Q:Write a Python script to merge two Python dictionaries

```
d1 = {'a': 100, 'b': 200}
d2 = {'x': 300, 'y': 200}
d = d1.copy()
d.update(d2)
print(d)
```

Q:Write a Python program to map two lists into a dictionary

```
keys = ['red', 'green', 'blue']
values = ['#FF0000','#008000', '#0000FF']
color_dictionary = dict(zip(keys, values))
print(color_dictionary)
```

Q:Write a Python program to combine two dictionary adding values for common keys.

o d1 = {'a': 100, 'b': 200, 'c':300}
o d2 = {'a': 300, 'b': 200,’d’:400}

```
from collections import Counter
d1 = {'a': 100, 'b': 200, 'c':300}
d2 = {'a': 300, 'b': 200, 'd':400}
d = Counter(d1) + Counter(d2)
print(d)
```

Q:Write a Python program to print all unique values in a dictionary.

```
L = [{"V":"S001"}, {"V": "S002"}, {"VI": "S001"}, {"VI": "S005"},
{"VII":"S005"}, {"V":"S009"},{"VIII":"S007"}]
print("Original List: ",L)
u_value = set( val for dic in L for val in dic.values())
print("Unique Values: ",u_value)
```

Q:Why Do You Use the Zip () Method in Python?

Python's zip() function creates an iterator that will aggregate elements from two or more iterables. You can use the resulting iterator to quickly and consistently solve common programming problems, like creating dictionaries

Q:Write a Python program to find the highest 3 values in a dictionary

```
from heapq import nlargest
my_dict = {'a':500, 'b':5874, 'c': 560,'d':400, 'e':5874, 'f': 20}
three_largest = nlargest(3, my_dict, key=my_dict.get)
print(three_largest)
```

Q:Write a Python function to calculate the factorial of a number (a non-negative integer)

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)
n=int(input("Input a number to compute the factiorial : "))
print(factorial(n))
```

Q:Write a Python function to check whether a number is in a given range

```
def test_range(n):
    if n in range(3,9):
        print( " %s is in the range"%str(n))
    else :
        print("The number is outside the given range.")
test_range(5)
```

Q:Write a Python function to check whether a number is perfect or not.

```
def perfect_number(n):
    sum = 0
    for x in range(1, n):
        if n % x == 0:
            sum += x
```

```
  return sum == n
print(perfect_number(6))
```

Q:Write a Python function that checks whether a passed string is palindrome or not

```
def isPalindrome(string):
    left_pos = 0
    right_pos = len(string) - 1

    while right_pos >= left_pos:
        if not string[left_pos] == string[right_pos]:
            return False
        left_pos += 1
        right_pos -= 1
    return True
print(isPalindrome('aza'))
```

Q:How can you pick a random item from a list or tuple?

```
pi=22/7
degree = float(input("Input degrees: "))
radian = degree*(pi/180)
print(radian)
```

Q:Write a Python program to calculate the area of a trapezoid

```
base_1 = 5
base_2 = 6
height = float(input("Height of trapezoid: "))
base_1 = float(input('Base one value: '))
base_2 = float(input('Base two value: '))
area = ((base_1 + base_2) / 2) * height
print("Area is:", area)
```

Q:Write a Python program to calculate the area of a parallelogram

```
base = float(input('Length of base: '))
height = float(input('Measurement of height: '))
area = base * height
print("Area is: ", area)
```

Q:Write a Python program to calculate surface volume and area of a cylinder

```
pi=22/7
height = float(input('Height of cylinder: '))
radian = float(input('Radius of cylinder: '))
volume = pi * radian * radian * height
sur_area = ((2*pi*radian) * height) + ((pi*radian**2)*2)
print("Volume is: ", volume)
print("Surface Area is: ", sur_area)
```

Q:Write a Python program to returns sum of all divisors of a number

```
def sum_div(number):
    divisors = [1]
    for i in range(2, number):
        if (number % i)==0:
            divisors.append(i)
    return sum(divisors)
print(sum_div(8))
print(sum_div(12))
```

Q:Write a Python program to find the maximum and minimum numbers from the specified decimal numbers.

```
from decimal import *
data = list(map(Decimal, '2.45 2.69 2.45 3.45 2.00 0.04
7.25'.split()))
```

```
print("Maximum: ", max(data))
print("Minimum: ", min(data))
```