SEPTEMBER-29

Slicing:

- It contains 3 values
- Slicing happens based on position

  [start position: stop position: step size]
- Default start position is 0
- Default stop position is end
- Default step size is 1 1. Print elements from a to true l1 =

['a',2,'e',9.5,'besant',True,'Tech',34,99.9] l1[0:6:1] output:

['a', 2, 'e', 9.5, 'besant', True]

2. Print odd position elements

l1[1::2] output:

[2, 9.5, True, 34]

3.To print all the elements of the list

L1[:]

Output: ['a',2,'e',9.5,'besant',True,'Tech',34,99.9]

4.To print last three values

L1[-3:]

Output:

['Tech', 34, 99.9]

5.Print all the elements whose position is divisible by 3 l1[3::3] output:

[9.5, 'Tech']

6. Write a program to find the max item from list without using max function.

[4,6,1,9,2] Code:

l2 = [4,6,1,9,2]

max_num = l2[0] for

i in l2:

if(i>max_num):

max_num = i

print(max_num)

Output: 9 Sorting:

Default is ascending

l2 = [4,6,1,9,2]

l2.sort() l2 output:

[1, 2, 4, 6, 9]

To print in descending:

l2.sort(reverse=True)

l2 output:

[9, 6, 4, 2, 1]

7. l3 = [1,2,3,3,3,4,4,5,6,7,8,9,9] Remove

all duplicates from the list Code:

l3 = [1,2,3,3,3,4,4,5,6,7,8,9,9]

new_l3 = [] for i in l3:   if i

not in new_l3:

new_l3.append(i)

print(new_l3) output:

[1, 2, 3, 4, 5, 6, 7, 8, 9] 8. list1 =

[5,20,15,20,25,50,20] remove all

occurrences of item 20 code-1:

list1 = [5,20,15,20,25,50,20]

for i in list1:   if(i==20):

```
list1.remove(20)   print(list1)

output: [5, 15, 25, 50] Code-

2:

list1 =  [5,20,15,20,25,50,20]
while 20 in list1:

list1.remove(20)

print(list1) output:

list1 =  [5,20,15,20,25,50,20]
```

9.   l1   =   [1,2,3,4,5]   l2   =

[4,5,6,7,8]

perform union and intersection on 2 given list code:

```
l1 = [1,2,3,4,5]

l2 = [4,5,6,7,8]

intersection = []

for i in l1:     if i

in l2:

        intersection.append(i)

print(intersection) output:

[4, 5] Code:

l1 = [1,2,3,4,5]

l2 = [4,5,6,7,8]

union = [] for i

in l1:     if i not

in l2:

union.append(i
```

) for i in l2:

if i not in

union:

union.append(i

) print(union)

output:

[1, 2, 3, 4, 5, 6, 7, 8]

10. Remove empty string from the list of string

list1 = ["arun","","kamala","","john"] code:

list1 = ["arun","","kamala","","john"]

for i in list1:   if(i==""):

list1.remove("") print(list1) output:

['arun', 'kamala', 'john']

Tuple:

- It is represented by tuple() or ()
- Its ordered,immutable,allow duplicates
- It is also Heterogeneous in nature

To create a tuple- t1 = (23,'Indu',78.9,False,'karan') o/p:

(23, 'Indu', 78.9, False, 'karan')

Merging two tuples: t1 =

(23,'Indu',78.9,False,'karan') t2 =

(24,5.3,10+1j) t1=t1+t2 t1

o/p: (23, 'Indu', 78.9, False, 'karan', 24, 5.3, (10+1j))

To print every element with its index: for i in

enumerate(t1):   print(i) output: (0, 23)

```
(1, 'Indu')

(2, 78.9)

(3, False)

(4, 'karan')

(5, 24)

(6, 5.3)

(7, (10+1j))
```

- Tuple is immutable so we can’t do data manipulation.

Programs:

1. t1 = (2,5,8,1,4)
   Create a new tuple containing the square of the above tuple.
   Code:
   ```
   t1   =   (2,5,8,1,4)   ns   =
   tuple(x**2   for   x   in   t1)
   print(ns)
   ```
   o/p:
   ```
   (4, 25, 64, 1, 16)
   ```
2. create odd,even,prime number tuple from 1 to 20 number.
   Code: numbers = range(1, 21) odd_numbers = tuple(n
   ```
   for n in numbers if n % 2 != 0) even_numbers = tuple(n
   for n in numbers if n % 2 == 0) def is_prime(n):     if n
   < 2:
         return False     for i in
   range(2, int(n**0.5) + 1):        if n
   % i == 0:            return False
      return True

   prime_numbers = tuple(n for n in numbers if is_prime(n))
   ```

print("Odd numbers :", odd_numbers) print("Even numbers:", even_numbers)
print("Prime numbers:", prime_numbers)

output:
Odd numbers : (1, 3, 5, 7, 9, 11, 13, 15, 17, 19)
Even numbers: (2, 4, 6, 8, 10, 12, 14, 16, 18, 20)
Prime numbers: (2, 3, 5, 7, 11, 13, 17, 19)

3. Print elements from a to true t1 =

   ('a',2,'e',9.5,'besant',True,'Tech',34,99.9)

   code:

t1 = ('a', 2, 'e', 9.5, 'besant', True, 'Tech', 34, 99.9)

end = t1.index(True) result = t1[:end + 1]

print(result)

output: ('a', 2, 'e', 9.5, 'besant', True)

4. Print odd position elements t1[1::2]
   o/p: (2, 9.5, True, 34)
5. To print all the elements of the tuple t1[:]
   o/p: ('a', 2, 'e', 9.5, 'besant', True, 'Tech', 34, 99.9)
6. To print last three values t1[-3:]
   o/p: ('Tech', 34, 99.9)
7. Print all the elements whose position is divisible by 3 t1[3::3] (9.5, 'Tech')

8. Write a program to find the max item

   from tuple without using max function.

   (4,6,1,9,2) Code:

t1 = (4, 6, 1, 9, 2)

max_item = t1[0] for

num in t1:

if num > max_item: max_item

= num print("Maximum item is:",

max_item) output:

Maximum item is: 9

9. t3=(1,2,3,3,3,4,4,5,6,7,8,9,9) Remove all duplicates from the tuple Code:
   t3 = (1,2,3,3,3,4,4,5,6,7,8,9,9)

   list1 = [] for
   i in t3:   if i
   not in list1:
   list1.appen
   d(i)

   unique_tuple = tuple(list1)
   print(unique_tuple) o/p:
   (1, 2, 3, 4, 5, 6, 7, 8, 9)