Day 4 – USER INPUTS, LISTS

Python User Input:

• The input() function allows user input

• You can assign the user input to a variable

Example:

In [2]:

```
x = input('Enter your Favourite cricter:')
print('Your favourite cricketer is ' + x)
```

```
Enter your Favourite cricter:pandya
Your favourite cricketer is pandya
```

Python Lists:

• The list type is a container that holds a number of other objects, in a given order

• The list type allows you to add and remove objects from the sequence.

Building a list:

In [3]:

```
A = ['a','b','c']
B = [1,2,3]
```

Accessing a list:

• The number of items in the list, L[i] returns the item at index i (the first item has in-dex 0), and L[i:j] returns a new list, containing the objects between i and j.

• Index of list starts from 0

```
item = L[index]

Example: second_element = A[1]

seq = L[start:stop]
```

Example:

In [4]:

```
print(A[1:2])
```

```
['b']
```

• If you pass in a negative index, Python adds the length of the list to the index. L[-1] can be used to access the last item in a list.

• For normal indexing, if the resulting index is outside the list, Python raises an Index-Error exception.

• You can also add items to an existing sequence. The append method adds a single item to the end of the list, the extend method adds items from another list (or any se-quence) to the end, and insert inserts an item at a given index, and move the remain-ing items to the right.

```
L.append(item)
L.extend(sequence)
L.insert(index, item)
```

• You can also remove items. The del statement can be used to remove an individual item, or to remove all items identified by a slice. The pop method removes an indi-vidual item and returns it, while remove searches for an item, and removes the first matching item from the list.

```
del L[i]
del L[i:j]
item = L.pop() # last item
item = L.pop(0) # first item
item = L.pop(index)
L.remove(item)
```

• To get the smallest or largest item in a list, use the built-in min and max functions:

```
lo = min(L)
hi = max(L)
```

• The sort method sorts a list in place.

```
L.sort()
```

• To get a sorted copy, use the built-in sorted function

```
out = sorted(L)
```

Python Tuples:

• A tuple in Python is similar to a list. The difference between the two is that we cannot change the elements of a tuple once it is assigned whereas we can change the elements of a list.

• A tuple can have any number of items and they may be of different types (integer, float, list, string, etc.).

In [5]:

```
my_tuple = (1,2,3)
print(my_tuple)
```

```
(1, 2, 3)
```

- Tuple with mixed datatypes:

In [6]:

```
my_tuple = (1, "Hello", 3.4)
print(my_tuple)
```

```
(1, 'Hello', 3.4)
```

- Accessing tuple elements using indexing

In [7]:

```
my_tuple = ('p','e','r','m','i','t')
print(my_tuple[0])
```

```
p
```

- Unlike lists, tuples are immutable (They cannot be changed)

Exercise:

1. Write a program to create a list of n integer values and do the following
   - Add an item in to the list (using function)
   - Delete (using function)
   - Store the largest number from the list to a variable
   - Store the Smallest number from the list to a variable

In [11]:

```
n = int(input("The value of n is:"))
nums = list(range(0, n))
print(nums)
nums.append(n)
print("The appended list is ", nums)
del nums[3]
print("The list is after deleting is ", nums)
max_num = max(nums)
min_num = min(nums)
print("The max num is ", max_num)
print("The min num is ", min_num)
```

```
The value of n is:5
[0, 1, 2, 3, 4]
The appended list is  [0, 1, 2, 3, 4, 5]
The list is after deleting is  [0, 1, 2, 4, 5]
The max num is  5
The min num is  0
```

2. Create a tuple and print the reverse of the created tuple

In [13]:

```
tuple = ('a','a','h','a','n','a')
print(tuple[::-1])
nums_tuple = (1, 4, 9, 16, 25)
print(nums_tuple[::-1])
```

```
('a', 'n', 'a', 'h', 'a', 'a')
(25, 16, 9, 4, 1)
```

3. Create a tuple and convert tuple into list

In [18]:

```
alpha = ('a', 'b', 'c', 'd')
print(alpha)
alpha = list(alpha)
print(alpha)
nums_tuple = (1, 4, 9, 16, 25)
nums_tuple = list(nums_tuple)
type(nums_tuple)
```

```
('a', 'b', 'c', 'd')
['a', 'b', 'c', 'd']
```

Out[18]:

```
list
```

Completed Day 1's notes & exercises

THANK YOU!

Check out My Repository at https://github.com/AakankshaJarode/BestEnlist_Python_Internship.git (https://github.com/AakankshaJarode/BestEnlist_Python_Internship.git)

Chech out My LinkedIn Page at https://www.linkedin.com/in/aakanksha-jarode-1b0195179 (https://www.linkedin.com/in/aakanksha-jarode-1b0195179)