**Creation of List:**

# empty list

my\_list = []

# list of integers

my\_list = [1, 2, 3]

# list with mixed data types

my\_list = [1, "Hello", 3.4]

**Nested List:**

my\_list = ["mouse", [8, 4, 6], ['a']]

**Indexing in List:**

# List indexing

my\_list = ['p', 'r', 'o', 'b', 'e']

# Output: p

print(my\_list[0])

# Output: o

print(my\_list[2])

# Output: e

print(my\_list[4])

# Nested List

n\_list = ["Happy", [2, 0, 1, 5]]

# Nested indexing

print(n\_list[0][1])

print(n\_list[1][3])

# Error! Only integer can be used for indexing

print(my\_list[4.0])

**To search whether an element is present in the list or not.**

list1=[4,1,2,5,3]

found=0

search=int(input("Enter Search number:"))

for i in range(0,len(list1)):

if search==list1[i]:

found=1

break

if found==1:

print("The Element is found in position:",i)

else:

print("The element is not found")

**List Comprehension example 1:**

[i\*2 for i in [1,2,3,4]]

**List Comprehension example 2:**

[i\*\*3 for i in [1,2,3,4] if i>2]

**List Comprehension example 3:**

numbers = list(range(1,10))

squares =[n\*n for n in numbers]

odd\_squares = [n\*n for n in numbers if n%2==1]

print (numbers)

print (squares)

print (odd\_squares)

**List Membership Test:**

my\_list = ['p', 'r', 'o', 'b', 'l', 'e', 'm']

# Output: True

print('p' in my\_list)

# Output: False

print('a' in my\_list)

# Output: True

print('c' not in my\_list)

Tuple Membership Test:

# Membership test in tuple

my\_tuple = ('a', 'p', 'p', 'l', 'e',)

# In operation

print('a' in my\_tuple)

print('b' in my\_tuple)

# Not in operation

print('g' not in my\_tuple)

**Tuple Accessing:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Mouse | 8 | 4 | 6 | 1 | 2 | 3 |

Nested Tuple

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | | | 2 | | |
| 0 | 1 | 2 | 0 | 1 | 2 |
| Mouse | 8 | 4 | 6 | 1 | 2 | 3 |
| 0 | 1 | | | 2 | | |
| 0 | 1 | 2 | 0 | 1 | 2 |
| Mouse | 8 | 4 | 6 | 1 | 2 | 3 |

my\_tuple = ("mouse", [8, 4, 6], (1, 2, 3))

my\_tuple[0] Output: ‘mouse’

my\_tuple[1] Output: [8,4,6]

my\_tuple[2] Output: (1,2,3)

my\_tuple[1][0] Output: 8

my\_tuple[1][1] Output: 4

my\_tuple[1][2] Output:6

my\_tuple[2][0] Output:1