Experiment No : 05

Aim : Write python programs to understand different List and Tuple operations using Built-in function.

Description :

List :

A list is a collection of arbitrary objects, somewhat akin to an array in many other programming languages but more flexible. Lists are defined in Python by enclosing a comma-separated sequence of objects in square brackets ([])

Example : List=[44,55,88]

Tuple :

Tuples are identical to lists in all respects, except for the following properties:

Tuples are defined by enclosing the elements in parentheses (()) instead of square brackets ([]).

Tuples are immutable.

Example : Tuple=(44,55,99)

Implementation :

Code :

List :

```
a=[1,2,3,4,5]
#list constructor
b=list(("cricket", "hockey"))
print(a,b)
a=[1,2,3,4,5]
#duplicates allowed
b=list(("cricket", "hockey", "cricket"))
# boolean item
c=[True, False]
print(a,b)
# no of items in the list
print(len(a))
print(len(b))
# any data type
x=[1,"cricket", True, 2, False]
print(x)
#list indexed
print(a[2])
#range of items
print(a[1:3])
print(a[:3])
```

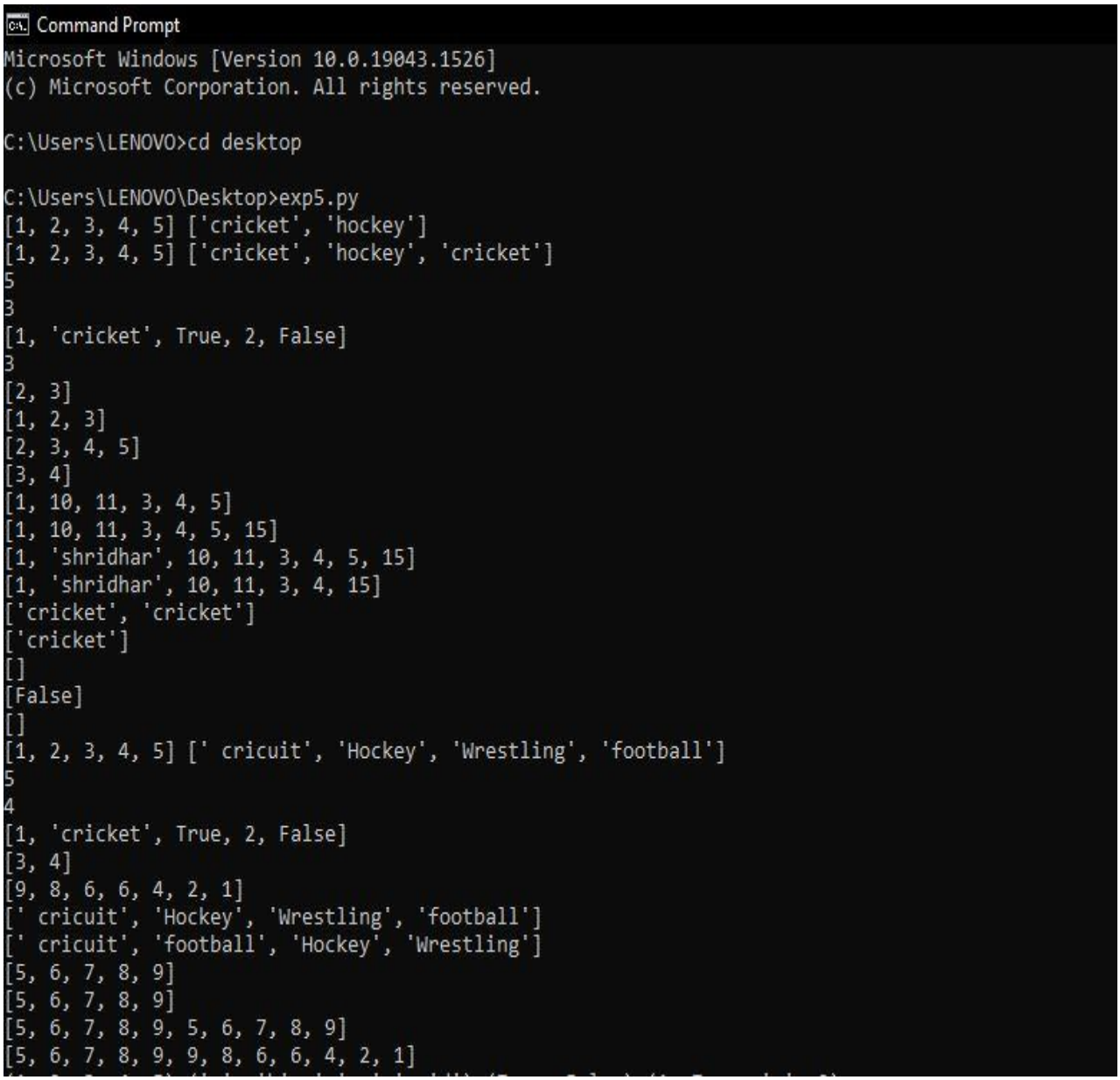
```
print(a[1:])
print(a[-3:-1])
# change list item values
a[1]=9
#print(a)
a[1:2]=[10,11]
print(a)
#add list items
a.append(15)
print(a)
a.insert(1,"shridhar")
print(a)
# remove list items
a.remove(5)
print(a)
b.remove("hockey")
print(b)
#remove particular item with pop function(only specifiy index value)
b.pop(0)
print(b)
b.pop()
# last index value will be removed
print(b)
del c[0]
```

```
print(c)
#delete entire list
c=[4,58,55]
del c
c=[4,58,55]
# empty entire list
c.clear()
print(c)


#sort lists


a=[1,2,3,4,5]
#duplicates allowed
b=list((" cricuit", "Hockey" , "Wrestling" ,"football"))

# boolean item
c=[True, False]
print(a,b)
# no of items in the list
print(len(a))
print(len(b))
# any data type
x=[1,"cricket", True, 2, False]
```

```
print(x)
print(a[-3:-1])
z=[9,6,1,4,2,8,6]
# desending order list  display
z.sort(reverse=True)
print(z)
# sort function case sensetive
a=[1,2,3,4,5]
b=list((" cricuit", "Hockey" , "Wrestling" ,"football"))
c=[True, False]
b.sort()
print(b)
#CASE INSENSETIVE SORTING
b.sort(key=str.lower)
print(b)
# copy lists
y=[5,6,7,8,9]
t=y.copy()
print(t)
y=[5,6,7,8,9]
g=list(y)
print(g)
#join lists
h=y+g
```

```
print(h)

y.extend(z)

print(y)
```

Output :



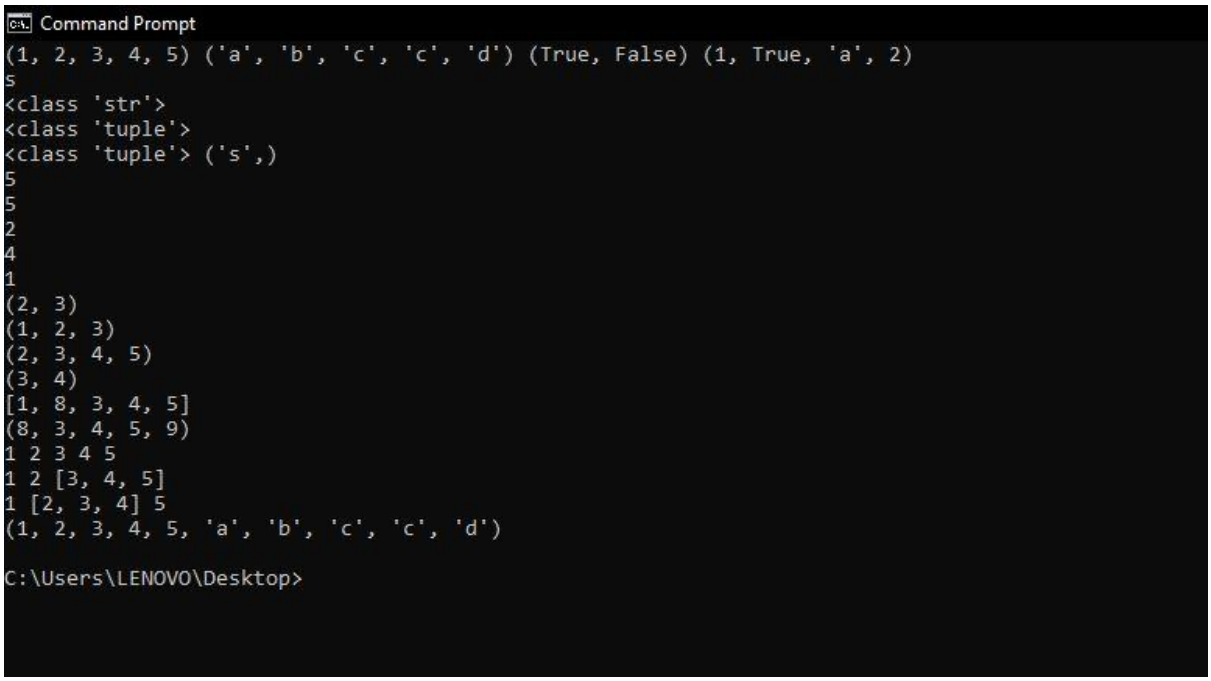
Tuple :

```
#tuble is also (to store multIple items in a single variable)
a=(1,2,3,4,5)
# TUPLE CONSTRUCTOR
b=tuple(("a","b","c", "c" ,"d"))
c=(True,False)
x=(1,True,"a",2)
print(a,b,c,x)
#tuple print in round bracket
#tuple have one item
z=("s")
print(z)
# check its type
z=("s")
print(type(z))
# but it will not print like tuple on command prompt
#it is  tuple.Tuple is ordered.
z=("s",)
print(type(z))
#print tuple.Tuple is ordered.Tuple is unchangeble.duplicates allowed
z=("s",)
print(type(z),z)
# No of items in tuple
```

```
print(len(a))

print(len(b))

print(len(c))

print(len(x))

#access tuple items. index work here

print(a[0])

# display range of index values

print(a[1:3])

print(a[ :3])

print(a[1: ])

print(a[-3:-1])


# indirect method of changing tuples  items , remove items .convert tuple in list. apply methods (like in list) .then convert list in tuples.

l=list(a)

l[1]=8

print(l)

l.append(9)

l.remove(1)

d=tuple(l)

print(d)

#packing . wnen you create tuple .it means packing items in tuple. Unpacking means removing  packing(trying to extract the values to the variables)

#variables(no of items in tuple=no of variables)
```

```
v,n,m,s,e=a

print(v,n,m,s,e)

#less no of variables compared to items.

v,n,*m=a

print(v,n,m)

v,*n, m=a

print(v,n,m)

# join tuples

f=a+b

print(f)

del f


#delete tuple

#print error. f no longer exist
```

Output :



Conclusion : Therefore we have successfully implemented different List and Tuple operations using Built-in function in python.