What is difference between list and tuple?

List	**Tuple**
List is a mutable collection	Tuple is a immutable collection
After creating list changes can be done	After creating tuple changes cannot done
List is created using square brackets []	Tuple is created using parenthesis ()
List occupy more space, because of performing mutable operations	Tuple occupy less space because immutability
List cannot used as an element in hashable collections	Tuple can be used as an element in hashable collections
“list’ class or data type represents list object	“tuple” class or data type represents tuple object

Example:

```
# Python – Maximum and Minimum K elements in Tuple
# Input : test_tup = (3, 7, 1, 18, 9), k = 2
# Output : (3, 1, 9, 18)

test_tup=(3, 7, 1, 18, 9)
k=2
test_tup1=tuple(sorted(test_tup))
print(test_tup1)
result_tup=test_tup1[:k]+test_tup1[-k:]
print(result_tup)
```

Output

```
(1, 3, 7, 9, 18)
(1, 3, 9, 18)
```

```
#Create a List of Tuples with Numbers and Their Cubes – Python
'''
We are given a list of numbers and our task is to
create a list of tuples where each tuple contains
a number and its cube.
For example, if the input is [1, 2, 3],
the output should be [(1, 1), (2, 8), (3, 27)].'''

A=[1,2,3,4,5]
B=[]
```

```
for value in A:
    B.append((value,value**3))

print(A)
print(B)
C=[(value,value**3) for value in A]
print(C)

# Python – Sum of tuple elements
'''
The original tuple is : (7, 8, 9, 1, 10, 7)
The summation of tuple elements are: 42
'''

t1=(7, 8, 9, 1, 10, 7)
s=sum(t1)
print(t1)
print(s)

s=0
for value in t1:
    s=s+value
print(s)


# Python – Modulo of tuple elements

'''
The original tuple1 : (10, 4, 5, 6)
The original tuple2 : (5, 6, 7, 5)
The modulus tuple : (0, 4, 5, 1)
'''

t1=(10, 4, 5, 6)
t2=(5, 6, 7, 5)
res=tuple([t1[i]%t2[i] for i in range(len(t1))])
print(t1)
print(t2)
print(res)
```

```
t1=(10,4,5)
t2=(5,6,7,5)
if len(t1)<len(t2):
    res=tuple([t1[i]%t2[i] for i in range(len(t1))])
elif len(t1)>len(t2):
    res=tuple([t1[i]%t2[i] for i in range(len(t2))])
else:
    res=tuple([t1[i]%t2[i] for i in range(len(t1))])

print(t1)
print(t2)
print(res)


#Python – Row-wise element Addition in Tuple Matrix
'''
Input : test_list = [[(‘Gfg’, 3)], [(‘best’, 1)]]
cus_eles = [1, 2]
Output : [[(‘Gfg’, 3, 1)], [(‘best’, 1, 2)]]'''

test_list = [[('Gfg', 3)], [('best', 1)]]
output_list=[]
cus_eles = [1, 2]
i=0
for row in test_list:
    for t in row:
        l=list(t)
        l.append(cus_eles[i])
        i=i+1
        output_list.append([tuple(l)])

print(output_list)


# Update Each Element in Tuple List – Python
'''The task of updating each element in a tuple list
in Python involves adding a specific value to every
element within each tuple in a list of tuples'''

'''For example, given a = [(1, 3, 4), (2, 4, 6), (3, 8, 1)]
```

```
and ele = 4,
the goal is to add 4 to every element in each tuple,
resulting in [(5, 7, 8), (6, 8, 10), (7, 12, 5)].'''

a=[(1, 3, 4), (2, 4, 6), (3, 8, 1)]
ele=4
b=[]
for t in a:
    l=list(t)
    for i in range(len(l)):
        l[i]=l[i]+ele
    b.append(tuple(l))

print(a)
print(b)


# Python | Multiply Adjacent elements
'''
The original tuple : (1, 5, 7, 8, 10)
Resultant tuple after multiplication : (5, 35, 56, 80)
'''

t1=(1, 5, 7, 8, 10)
a=[]

for i in range(len(t1)-1):
    a.append(t1[i]*t1[i+1])
print(t1)
print(tuple(a))
```

String data type (OR) str class