tuple assignment

March 5, 2023

Q1. What are the characteristics of the tuples? Is tuple immutable? sol: A tuple is an another data type in python which store objects just like like lists. Yes tuples are immutable.

Q2. What are the two tuple methods in python? Give an example of each method. Give a reason why tuples have only two in-built methods as compared to Lists. sol: There are only two tuple methods count() and index() that a tuple object can call. 1. count() 2. index() examples:

[6]:
```
a = (1,1,1,2,3,"rik")
a.count(1)
```

[6]: 3

[8]:
```
a = (1,1,1,2,3,"rik")
a.index("rik")
```

[8]: 5

Q3. Which collection datatypes in python do not allow duplicate items? Write a code using a set to remove duplicates from the given list. sol : set datatype doesn't allow the duplicacy of items in it. exapple :

[25]:
```
set1 = {"rik","rik","sam",1,1,2,3,3,5}
print(set1)
set1.
```

```
{1, 2, 3, 5, 'sam', 'rik'}
```

[29]:
```
List = [1, 1, 1, 2, 1, 3, 1, 4, 2, 1, 2, 2, 2, 3, 2, 4, 3, 1, 3, 2, 3, 3, 3, 4,␣
↪4, 1, 4, 2, 4, 3, 4, 4]
set2 = set(List)
print(set2)
```

```
{1, 2, 3, 4}
```

Q4. Explain the difference between the union() and update() methods for a set. Give an example of each method.

sol: update: The update() method updates the current set, by adding items from another set (or any other iterable). union : The union() method returns a set that contains all items from the original set, and all items from the specified sets.

[33]:
```
##union:
x= {"a", "b", "c"}
y= {"f", "d", "a"}
z= x.union(y)
print(z)
```

```
{'d', 'b', 'c', 'a', 'f'}
```

[34]:
```
##update
x= {"a", "b", "c"}
y= ["f", "d", "a"]
x.update(y)
print(x)
```

```
{'b', 'd', 'c', 'a', 'f'}
```

Q5. What is a dictionary? Give an example. Also, state whether a dictionary is ordered or unordered. sol: A dictionary is a datatype collection which is ordered, changeable and do not allow duplicates. Dictionaries are used to store data values in key:value . Yes, it is ordered collection.

[43]:
```
'''example:'''
dict1 = {"a":"123","b":1235}
```

[43]:
```
dict
```

Q6. Can we create a nested dictionary? If so, please give an example by creating a simple one-level nested dictionary. Sol: yes we can. example:

[48]:
```
dict1 = {"a":123,"b":1235,"c":{"a":12365}}
```

Q7. Using setdefault() method, create key named topics in the given dictionary and also add the value of the key as this list ['Python', 'Machine Learning', 'Deep Learning'] sol :

[59]:
```
dict1 = {'language' : 'Python', 'course': 'Data Science Masters'}
a = dict1.setdefault("topics", ["Python", "Machine Learning", "Deep Learning"] )
```

[60]:
```
dict1.get("topics")
```

[60]:
```
['Python', 'Machine Learning', 'Deep Learning']
```

Q8. What are the three view objects in dictionaries? Use the three in-built methods in python to display these three view objects for the given dictionary. sol:The three view objects in dictionaries are : 1. key 2.value 3. item

[70]:
```
###for displaying keys:
dict1 = {'language' : 'Python', 'course': 'Data Science Masters'}
print(dict1.keys())
```

```
dict_keys(['language', 'course'])
```

[71]:
```
###for displaying values:
dict1 = {'language' : 'Python', 'course': 'Data Science Masters'}
print(dict1.values())
```

```
dict_values(['Python', 'Data Science Masters'])
```

[72]:
```
###for displaying items:
dict1 = {'language' : 'Python', 'course': 'Data Science Masters'}
print(dict1.items())
```

```
dict_items([('language', 'Python'), ('course', 'Data Science Masters')])
```

[ ]:
```

```