tuple set dict Assingment

February 11, 2023

1 Q.1

'''1) Ordering: The elements in a tuple have a defined order, and this order is preserved.

2) Immutable: Once a tuple is created, its elements cannot be changed. This means that tuples are "read-only".

3) Heterogeneous: Tuples can contain elements of different data types, such as integers, strings, and other objects.

4) Indexable: The elements in a tuple can be accessed using their index, starting from 0.

4)Iterable: Tuples can be iterated over using a loop, just like lists.

yes, tuples are immutable.'''

Q.2 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.

'"'Tuples have two built-in methods in Python: count() and index(). 1) count(): This method returns the number of times a specified element appears in the tuple. Here's an example:"'

[11]:
```
num=(1,2,3,4,5,4,3,4,5,6,6,4,3,2,1,3)
```

[9]:
```
type(num)
```

[9]:
```
tuple
```

[10]:
```
num.count(3)
```

[10]:
```
4
```

'"'2) index(): This method returns the index of the first occurrence of a specified element in the tuple. Here's an example: python"'

[13]:
```
num
```

[13]:
```
(1, 2, 3, 4, 5, 4, 3, 4, 5, 6, 6, 4, 3, 2, 1, 3)
```

[16]:
```
num.index(4)
```

[16]:
```
3
```

”“Tuples have only two built-in methods because they are intended to be simple, lightweight, and efficient data structures. The idea is to keep them as basic as possible and let the programmer add more functionality through other data structures or functions if needed. Lists, on the other hand, have more built-in methods because they are intended to be more versatile and flexible data structures.” ’

2 Q.3 Which collection datatypes in python do not allow duplicate items? Write a code using a set to remove

duplicates from the given list. 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]

”“In Python, there are one collection data types that do not allow duplicate items: sets” ’

[18]:
```
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]
```

[21]:
```
unique_list=set(List)
print(unique_list)
```

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

[ ]:
```

```

3 Q.4 Explain the difference between the union() and update() methods for a set. Give an example of

each method.

”“The union() and update() methods are used to combine two or more sets in Python 1) union:- This method returns a new set that contains all the elements from the original set, as well as any elements in the set(s) being combined. The original set is not modified.” ’

[25]:
```
set1={1,2,3}
set2={2,3,4,}
set3=set1.union(set2)
print(set3)
```

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

#2.update :- This method updates the original set with the elements from the set(s) being combined. The original set is modified in-place.

[26]:
```
set1.update(set2)
print(set1)
```

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

[27]:
```
set1
```

[27]:
```
{1, 2, 3, 4}
```

3.1 Q5. What is a dictionary? Give an example. Also, state whether a dictionary is ordered or unordered.

”‘A dictionary is a collection data type in Python that stores key-value pairs. A dictionary is similar to a list, but instead of using integer indices to access elements, you use keys, which can be any immutable type (such as strings, numbers, or tuples).”’

[28]:
```
name={"Kishor":21,"sushil":23,"sanket":25}
```

[29]:
```
type(name)
```

[29]:
```
dict
```

[30]:
```
print(name)
```

```
{'Kishor': 21, 'sushil': 23, 'sanket': 25}
```

”‘Dictionaries in Python are unordered. This means that the items in a dictionary have no defined order, and they can be in any order when you retrieve them. If you need to preserve the order of items in a collection, you should use a list or another data structure that is ordered, such as an ordered dictionary or a list of tuples”’

3.2 .6 Can we create a nested dictionary? If so, please give an example by creating a simple one-level

nested dictionary.

“ ““Yes, we can create a nested dictionary in Python. A nested dictionary is a dictionary that contains one or more dictionaries as its values. Here’s an example”“ ”

[32]:
```
dict1={"name":["kishor","sushil","sanket","ashish"],"age":{21,23,34,15}}
```

[33]:
```
dict1
```

[33]:
```
{'name': ['kishor', 'sushil', 'sanket', 'ashish'], 'age': {15, 21, 23, 34}}
```

[34]:
```
dict1.keys()
```

[34]:
```
dict_keys(['name', 'age'])
```

[38]:
```
dict.items(dict1)
```

[38]:
```
dict_items([('name', ['kishor', 'sushil', 'sanket', 'ashish']), ('age', {34, 23,
21, 15})])
```

3.3 Q.7 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’]

[39]:
```
d = {}
d.setdefault('topics', ['Python', 'Machine Learning', 'Deep Learning'])
print(d)
```

```
{'topics': ['Python', 'Machine Learning', 'Deep Learning']}
```

’‘you can use the setdefault() method to create a key in a dictionary and set its value if the key doesn’t exist.”’

[41]:
```
dict1 = {'language' : 'Python', 'course': 'Data Science Masters'}
dict1.setdefault('topics', ['Python', 'Machine Learning', 'Deep Learning'])
print(dict1)
```

```
{'language': 'Python', 'course': 'Data Science Masters', 'topics': ['Python',
'Machine Learning', 'Deep Learning']}
```

3.4 Q.8 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. dict1 = {‘Sport’: ‘Cricket’ , ‘Teams’: [‘India’, ‘Australia’, ‘England’, ‘South Africa’, ‘Sri Lanka’, ‘New Zealand’]}

[42]:
```
dict1 = {'Sport': 'Cricket' , 'Teams': ['India', 'Australia', 'England', 'South␣
↪Africa', 'Sri Lanka', 'New Zealand']}
```

[43]:
```
dict1.keys()
```

[43]:
```
dict_keys(['Sport', 'Teams'])
```

[47]:
```
dict1.values()
```

[47]:
```
dict_values(['Cricket', ['India', 'Australia', 'England', 'South Africa', 'Sri
Lanka', 'New Zealand']])
```

[49]:
```
dict.items(dict1)
```

[49]:
```
dict_items([('Sport', 'Cricket'), ('Teams', ['India', 'Australia', 'England',
'South Africa', 'Sri Lanka', 'New Zealand'])])
```

[ ]:
```

```