
Dictionary

- It is an unordered collection of items where each item consist of a key and a value.
- It is mutable (can modify its contents) but Key must
- be unique and immutable.



`{'key': 'value'}`

Dictionary

Creating A Dictionary

It is enclosed in curly braces {} and each item is separated from other item by a comma(.). Within each item, key and value are separated by a colon (:). Passing value in dictionary at declaration is dictionary initialization.get() method is used to access value of a key

e.g.

```
dict = {'Subject': 'Informatic Practices', 'Class': '11'}
```

Accessing List Item

```
dict = {'Subject': 'Informatics Practices', 'Class': 11}  
print(dict)  
print ("Subject : ", dict['Subject'])  
print ("Class : ", dict.get('Class'))
```

OUTPUT

```
{'Class': '11', 'Subject': 'Informatics Practices'}  
('Subject : ', 'Informatics Practices') ('Class : ', 11)
```

Dictionary

Iterating / Traversing through A Dictionary

Following example will show how dictionary items can be accessed through loop.

e.g.

```
dict = {'Subject': 'Informatics Practices', 'Class': 11}
```

```
for i in dict:
```

```
    print(dict[i])
```

OUTPUT

11

Informatics Practices

Updating/Manipulating Dictionary Elements

We can change the individual element of dictionary.

e.g.

```
dict = {'Subject': 'Informatics Practices', 'Class': 11}
```

```
dict['Subject']='computer science'
```

```
print(dict)
```

OUTPUT

Dictionary

Deleting Dictionary Elements

del, pop() and clear() statement are used to remove elements from the dictionary.

del e.g.

```
dict = {'Subject': 'Informatics Practices', 'Class': 11}
print('before del', dict)
del dict['Class']    # delete single element
print('after item delete', dict)
del dict             #delete whole dictionary
print('after dictionary delete', dict)
```

Output

```
('before del', {'Class': 11, 'Subject': 'Informatics Practices'})
('after item delete', {'Subject': 'Informatics Practices'})
('after dictionary delete', <type 'dict'>)
```

Dictionary

pop() method is used to remove a particular item in a dictionary. clear() method is used to remove all elements from the dictionary.

e.g.

```
dict = {'Subject': 'Informatics Practices', 'Class': 11}  
print('before del', dict)  
dict.pop('Class')  
print('after item delete', dict) dict.clear()  
print('after clear', dict)
```

Output

```
('before del', {'Class': 11, 'Subject': 'Informatics  
Practices'}) ('after item delete', {'Subject': 'Informatics  
Practices'}) ('after clear', {})
```

Dictionary

Built-in Dictionary Functions

S.No.	Function & Description
1	<u>len(dict)</u> Gives the total length of the dictionary. It is equal to the number of items in the dictionary. dict = {'Name': 'Aman', 'Age': 37}; print ("Length : %d" % len (dict)) OUTPUT ->2
2	<u>str(dict)</u> Return a printable string representation of a dictionary
3	<u>type(variable)</u> If variable is dictionary, then it would return a dictionary type.

Dictionary

Built-in Dictionary Methods

S.No.	Method & Description
1	<code>dict()</code> - creates dictionary <code>x = dict(name = "Aman", age = 37, country = "India")</code> Here x is created as dictionary
2	<code>keys()</code> - returns all the available keys <code>x = dict(name = "Aman", age = 37, country = "India")</code> <code>print(x.keys())</code> OUTPUT-> <code>dict_keys(['country', 'age', 'name'])</code>
3	<code>values()</code> - returns all the available values <code>x = dict(name = "Aman", age = 37, country = "India")</code> <code>print(x.values())</code> OUTPUT-> <code>dict_values(['India', 37, 'Aman'])</code>

Dictionary

Built-in Dictionary Methods

S.No.	Method & Description
4	<code>items()</code> - return the list with all dictionary keys with values. <code>x = dict(name = "Aman", age = 37, country = "India") print(x.items())</code> OUTPUT-> <code>dict_items([('country', 'India'), ('age', 37), ('name', 'Aman')])</code>
5	<code>update()</code> -used to change the values of a key and add new keys <code>x = dict(name = "Aman", age = 37, country = "India")</code> <code>d1 = dict(age= 39) x.update(d1,state="Rajasthan") print(x)</code> OUTPUT- <code>{'country': 'India', 'age': 39,'name': 'Aman','state': 'Rajasthan'}</code>

Dictionary

Built-in Dictionary Methods

S.No.	Method & Description
6	del -used to remove key <pre>x = dict(name = "Aman", age = 37, country = "India") del x['age'] print(x) OUTPUT->{'country': 'India', 'name': 'Aman'}</pre> del x -> will remove complete dictionary
7	fromkeys() – is used to create dictionary from keys <pre>keys = {'a', 'e', 'i', 'o', 'u' } value ="Vowel" vowels = dict.fromkeys(keys, value) print(vowels) OUTPUT-> {'i': 'Vowel', 'u': 'Vowel', 'e': 'Vowel', 'a': 'Vowel', 'o': 'Vowel'}</pre>

Dictionary

Built-in Dictionary Methods

S.No.	Method & Description
8	<code>copy()</code> - returns a shallow copy of the dictionary. <code>x = dict(name = "Aman", age = 37, country = "India")</code> <code>y=x.copy()</code> <code>print(y)</code> <code>print(id(x))</code> <code>print(id(y))</code> OUTPUT - >{'country': 'India', 'age': 37, 'name': 'Aman'} 33047872 33047440
9	<code>popitem()</code> – removes last item from dictionary <code>x = dict(name = "Aman", age = 37, country = "India")</code> <code>x.popitem()</code> <code>print(x)</code> OUTPUT-> {'age': 37, 'name': 'Aman'}

Dictionary

Built-in Dictionary Methods

S.No.	Method & Description
10	<code>setdefault()</code> method returns the value of the item with the specified key. If the key does not exist, insert the key, with the specified value. <pre>x = dict(name = "Aman", country = "India") y=x.setdefault('age',39) print(y) OUTPUT-> 39</pre>
11	<code>max()</code> – returns key having maximum value <pre>Tv = {'a':100, 'b':1292, 'c' : 88} Keymax = max(Tv, key=Tv.get) print(Keymax) OUTPUT-> b</pre>
12	<code>min()</code>- returns key having minimum value

Dictionary

Built-in Dictionary Methods

S.No.	Method & Description
13	sorted- sort by key or value <pre>dict1 = {'b':100, 'a':12, 'c' : 88} y = sorted(dict1.items(),key=lambda x: x[1],reverse=True) print(y)</pre> OUTPUT-> [('b', 100), ('c', 88), ('a', 12)]

Dictionary

count the number of times a character appears in a given string using a dictionary

```
input_string = "python" frequencies = {}  
for char in input_string:  
    if char in frequencies:  
        frequencies[char] += 1  
    else:  
        frequencies[char] = 1  
  
print ("Per char frequency in '{}' is :\n {}".format(input_string, str(frequencies)))
```

OUTPUT

Per char frequency in 'python' is :

{'i': 1, 'h': 1, 'k': 1, 'm': 1, 'o': 1, 'n': 2, 'p': 1, 's': 1, 't': 1, 'v': 1, 'y': 2, '.': 2}

Dictionary

create a dictionary with names of employees, their salary and access them

```
employees =  
{ 'Aman': {'salary': '10000'}, 'Mayur': {'salary': '51000'}}  
employee1 = employees['Aman']  
print(employee1)
```

OUTPUT

```
{'salary': '10000'}
```

Dictionary

Questions.

1.Create dictionary to store 4 student details with rollno,name,age field.Search student in list.

2.Create dictionary for month and noofdays for a year. User is asked to enter month name and system will show no of days of that month.