Python Programming

RGM College of Engineering & Technology (Autonomous)

Department of Computer Science & Engineering

Academic Year : 2020-2021

DICTIONARY DATA TYPE - 3

Guido Van Rossum

Learning Mantra

If you really strong in the basics, then remaining things will become so easy.

Agenda:

1. Important functions/methods of Dictionary

1. popitem()
2. keys()
3. values()
4. items()
5. copy()

6. popitem():

- It removes an arbitrary item(key-value) from the dictionary and returns it.

Eg:

```
d={100:"karthi",200:"saha",300:"sri"}
print(d)
print(d.popitem())
print(d.popitem())
print(d)
print(d.pop(400))
```

```
{100: 'karthi', 200: 'saha', 300: 'sri'}
(300, 'sri')
(200, 'saha')
{100: 'karthi'}
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-29-4185b7c5bad9> in <module>
      4 print(d.popitem())
      5 print(d)
----> 6 print(d.pop(400))          # KeyError

KeyError: 400
```

If the dictionary is empty then we will get **KeyError.**

Eg:

```
d ={}

print(d.popitem())        #KeyError: 'popitem(): dictionary is empty‘
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-28-14f741a4e5d5> in <module>
      1 d ={}
----> 2 print(d.popitem())

KeyError: 'popitem(): dictionary is empty'
```

Eg:

d={100:"karthi",200:"saha",300:"sri"}

```
print(d)
print(d.popitem())
print(d.popitem())
print(d.popitem())
print(d.popitem())
print(d)
```

```
{100: 'karthi', 200: 'saha', 300: 'sri'}
(300, 'sri')
(200, 'saha')
(100, 'karthi')
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-30-17881d89d74e> in <module>
      4 print(d.popitem())
      5 print(d.popitem())
----> 6 print(d.popitem())
      7 print(d)

KeyError: 'popitem(): dictionary is empty'
```

7. keys():

- It returns all keys associated with dictionary.

Eg:

```
d={100:"karthi",200:"saha",300:"sri"}
print(d.keys())
for key in d.keys():
    print(key)
```

Output:

```
dict_keys([100, 200, 300])
100
200
300
```

8. values():

- It returns all values associated with the dictionary.

Eg:

```
d={100:"karthi",200:"saha",300:"sri"}
print(d.values())
for key in d.values():
    print(key)
```

Output:

```
dict_values(['karthi', 'saha', 'sri'])
karthi
saha
sri
```

9. items():

- It returns list of tuples representing key-value pairs like as shown below.

[(k,v),(k,v),(k,v)]

Eg:

```
d={100:"karthi",200:"saha",300:"sri"}

list = d.items()

print(list)
```

Output:

```
dict_items([(100, 'karthi'), (200, 'saha'), (300, 'sri')])
```

Eg:

```
d={100:"karthi",200:"saha",300:"sri"}
for k,v in d.items():
    print(k,"--",v)
```

Output:

```
100 -- karthi
200 -- saha
300 -- sri
```

10. copy():

- This method is used to create exactly duplicate dictionary(cloned copy).

Eg:

```
d={100:"karthi",200:"saha",300:"sri"}

d1=d.copy()

print(d1)

print(d)
```

Output:

```
{100: 'karthi', 200: 'saha', 300: 'sri'}

{100: 'karthi', 200: 'saha', 300: 'sri'}
```

Any question?

If you try to practice programs yourself, then you will learn many things automatically

Spend few minutes and then enjoy the study

Thank You