

PYTHON DICTIONARY

Python Dictionaries

- ☐ The python dictionary is called as associative data structure.
- ☐ Python dictionary is a mutable data type.
- ☐ To create dictionary use curly braces '{ }' in between use (,) for separate items and use (:) colon for key and its value.
- ☐ Dictionary will have keys and its values
- ☐ There are three ways to create python dictionaries,

Example 1:

```
>>>Num={1:"one",2:'two',3:'three'}
```

Here '1' is called as key and 'one' is called as value

Example 2: >>>temp= { }

```
>>>temp['sun']=20
```

```
>>>temp['mon']=22
```

```
>>>temp['tue']=26
```

```
>>>temp['wed']=28
```

```
>>>temp
```

```
{'sun':20,'mon':22,'tue':26,'wed':28}
```

Example:3

- ☐ Use dict keyword to create empty dictionary
- ☐ >>>dict={ }

And also

Dictionary :

- A dictionary is a collection data type.
- It is unordered, changeable and indexed.
- In Python, dictionaries are written with curly brackets,
- They have keys and values.

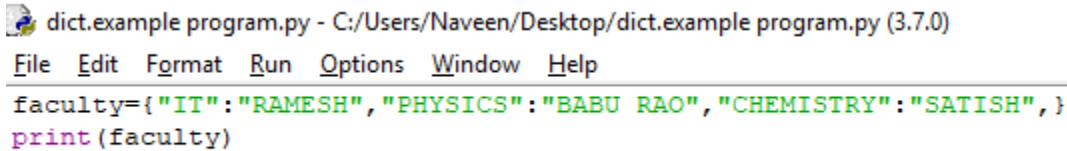
Creating a dictionary-

Creating a dictionary is as simple as placing items inside curly braces { } separated by a comma.

An item has a key and the corresponding value expressed as a pair, key: value.

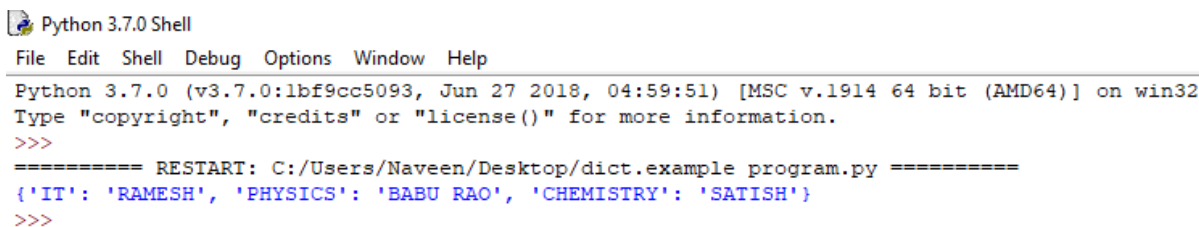
While values can be of any data type and can repeat, keys must be of immutable type (string, number or tuple with immutable elements) and must be unique.

EXAMPLE PROGRAM:



```
dict.example program.py - C:/Users/Naveen/Desktop/dict.example program.py (3.7.0)
File Edit Format Run Options Window Help
faculty={'IT':"RAMESH", "PHYSICS":"BABU RAO", "CHEMISTRY":"SATISH",}
print(faculty)
```

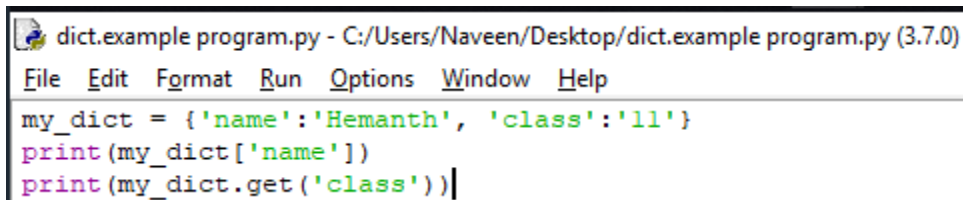
The output of the above program is as follows:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Naveen/Desktop/dict.example program.py =====
{'IT': 'RAMESH', 'PHYSICS': 'BABU RAO', 'CHEMISTRY': 'SATISH'}
>>>
```

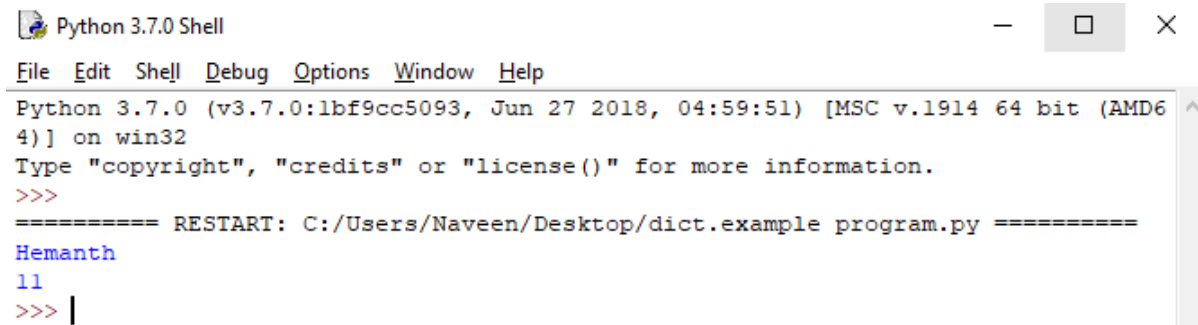
In the above program, (IT,PHYSICS,CHEMISTRY) are keys and (RAMESH,BABU RAO,SATISH) are values.

NOTE:-To access the elements in a dictionary, the following python program helps in guiding us.



```
dict.example program.py - C:/Users/Naveen/Desktop/dict.example program.py (3.7.0)
File Edit Format Run Options Window Help
my_dict = {'name':'Hemanth', 'class':'11'}
print(my_dict['name'])
print(my_dict.get('class'))
```

***The output is as follows:**



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Naveen/Desktop/dict.example program.py =====
Hemanth
11
>>> |
```

Python Dictionary Methods :

Methods that are available with dictionary are tabulated below.

Method	Description
<code>.clear()</code>	Removes all the elements from the dictionary
<code>.copy()</code>	Returns a copy of the dictionary
<code>.fromkeys()</code>	Returns a dictionary with the specified keys and values
<code>.get()</code>	Returns the value of the specified key
<code>.items()</code>	Returns a list containing a tuple for each key value pair
<code>.keys()</code>	Returns a list containing the dictionary's keys
<code>.pop()</code>	Removes the element with the specified key
<code>.popitem()</code>	Removes the last inserted key-value pair
<code>.setdefault()</code>	Returns the value of the specified key. If the key does not exist: insert the key, with the specified value
<code>.update()</code>	Updates the dictionary with the specified key-value pairs
<code>.values()</code>	Returns a list of all the values in the dictionary

Built-in Functions with Dictionary :

Built-in functions like `all()`, `any()`, `len()`, `cmp()`, `sorted()`, etc. are commonly used with dictionary to perform different tasks.

Built-in Functions with Dictionary

<i>FUNCTION</i>	<i>DESCRIPTION</i>
<u>all()</u>	Return True if all keys of the dictionary are true (or if the dictionary is empty).
<u>any()</u>	Return True if any key of the dictionary is true. If the dictionary is empty, return False.
<u>len()</u>	Return the length (the number of items) in the dictionary.
<u>cmp()</u>	Compares items of two dictionaries.
<u>sorted()</u>	Return a new sorted list of keys in the dictionary.

Example programs for each method in python dictionary:-

```
Sport={"cricket":'11',"ice-hockey":'20',"kabaddi":'7',"baseball":'9',"basket ball":'5'},
```

```
dict.get()
```

This method is used to get the specified value of a dictionary by entering the corresponding key in the bracket.

e.g.

```
sport.get("cricket")
```

#output

'11'

```
dict.items()
```

This method is used to return a list containing a tuple for each key value pair.

5

e.g.

dict.items()

#output

dict_items([('cricket','11'),('ice-hockey','20'),('kabaddi','7'),('baseball','9'),('basket ball','5')])

dict.keys()

This method is used to return all the keys of a dictionary.

e.g.

sport.keys()

dict_keys(['cricket','ice-hockey', 'kabaddi', 'baseball', 'basket ball'])

dict.pop()

This method is used to remove the specified item in a dictionary.

e.g.

sport.pop("ice-hockey")

#output

'20'

sport

#output

{ 'cricket':'11', 'kabaddi':'7', 'baseball':'9', 'basket ball':'5', }

dict.popitem()

6

This method is used to remove the last item in the dictionary.

e.g.

```
sport.popitem()
```

#output

```
( 'basket ball', '5' )
```

sport

#output

```
{ 'cricket': '11', 'kabaddi': '7', 'baseball': '9' }
```

```
dict.update()
```

This method is used to update the dictionary with specified key value.

e.g.

```
sport.update({ "volleyball": "6" })
```

sport

#output

```
{ 'cricket': '11', 'kabaddi': '7', 'baseball': '9', 'volleyball': '6' }
```

```
dict.values()
```

This method is used to return a list of all values in a dictionary.

e.g.

```
sport.values()
```

#output

7

```
dict_values(['11','7','9','6'])
```

```
dict clear()
```

This method is used to clear or erase the entire dictionary.

e.g.

```
sport.clear()
```

#EMPTY OUTPUT

```
fromkeys()
```

This method is used to return a dictionary with the specified keys and values.

e.g.

```
sport=('cricket','football','bandy')
```

```
players=11
```

```
thisdict=dict.fromkeys(sport,player)
```

#output

```
{'cricket':11,'football':11,'bandy':11}
```

Built-in Functions with Dictionary

any():-Return True if any key of the dictionary is true. If the dictionary is empty, return False.

e.g.

```
mydict = {0 : "Apple", 1 "orange"}
```

```
x = any(mydict)
```

#output

True

all: Return True if all keys of the dictionary are true (or if the dictionary is empty).

e.g.

```
mydict = {0 : "Apple", 1 "orange"}
```

```
y = all(mydict)
```

#output

False

len():-Return the length (the number of items) in the dictionary.

e.g.

```
dict = {'Tim':18,'Cook':12,'Roy':22}
```

```
print("Length : %d" % len (dict))
```

#output

Length : 4