

## Dictionary Data Type

04 June 2025 15:22

- A **dictionary** in Python is similar to a pocket dictionary where each word (key) is mapped to its meaning (value).
- In Python, a dictionary is a collection of **key-value pairs**, enclosed within {}.

Syntax: variable name={key:Value1,key2:value2.....key n: value n}

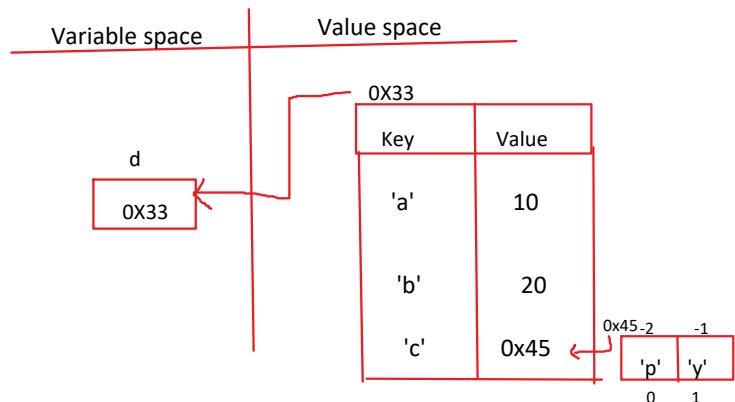
- Keys and values are separated by a **colon (:)**.
- Each key-value pair is separated by a **comma (,)**.
- The complete collection is called a **dictionary**.
- The default (empty) dictionary is {}, which is internally considered as **False** in boolean context.
- **Keys must be immutable** (e.g., string, number, tuple).

Example 1: d={'a':10,'b':20,'c':30,'a':58}

In above example key repeated again in case of duplication the old value will get overridden by new value. Key should be unique.

Example2: d={'a':10,'b':20,'c':'py'}

```
type(d)
Output:<class 'dict'>
len(d)
Output:3
bool(d)
Output: True
bool({})
Output: False
d['c'][1]
Output:- 'y'
```



Example 3: e=[[1,2]:8]

```
e
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    e=[[1,2]:8]
TypeError: unhashable type: 'list'.
```

### Accessing, Modifying, Adding, and Removing Data from the dictionary:

Example 4:

```
f={'a': 'angle', 'b': 'beautiful', 'c': 'cat', 'd': 'dog'}
f
Output:{'a': 'angle', 'b': 'beautiful', 'c': 'cat', 'd': 'dog'}
```

By using key we can access or fetch the data from the dictionary.

Syntax: variable name[key]

```
Example:-f['a']
Output:' angle'
f['c']
Output: 'cat'
```

By using key we can modify the values in dictionary.

Syntax: variable name[key]=new value

```
Example:-f['c']=10+6j
Output:
{'a': 'angle', 'b': 'beautiful', 'c': (10+6j), 'd': 'dog'}
```

We can add new key value pair by using below syntax.

Syntax: variable name[new key]=new value

---

Example:- `f['e']=100`

`f`

Output:-`{'a': ' angle', 'b': 'beautiful', 'c': (10+6j), 'd': 'dog', 'e':100}`

We can remove the values from dictionary by using `pop()`.

Syntax: Variable name. `pop(key)`

Example: `f.pop('d')`

Output: `'dog'`

`f`

Output:`{'a': ' angle', 'b': 'beautiful', 'c': (10+6j), 'e':100}`