

# Day8\_Dictionaries\_Built-in\_DS\_4

May 26, 2025

# Day 8 – Python Basics: Dictionaries

Today I explored **dictionaries** in Python — one of the most powerful built-in data structures. A dictionary is a collection of **key-value pairs**, enclosed in **curly braces {}** and separated by a **colon :**.

## Key Properties of Dictionaries:

- Dictionaries are **mutable** (we can add, update, or remove items).
  - Keys must be **unique** and **immutable** (like strings, numbers, or tuples).
  - Values can be **any data type**, including lists, tuples, or even other dictionaries.
- 

## Topics Covered:

- Creating dictionaries using {} or dict()
  - Using **integer**, **string**, and **mixed-type** keys
  - Accessing values using [] or .get()
  - Working with nested data (lists, tuples, and dictionaries as values)
  - Creating dictionaries from a sequence of keys using fromkeys()
  - Updating, adding, and removing items using:
    - update(), pop(), popitem(), del, and clear()
  - Copying dictionaries using copy()
  - Looping through keys and values using for loop
  - Checking membership using in
  - Using built-in functions: all(), any(), len(), id()
- 

**Real-life Example:** Dictionaries are often used to store structured data like:

```
employee = {  
    'Name': 'Akshay',  
    'ID': 12345,  
    'DOB': 1991,
```

```
    'Job': 'Analyst'  
}
```

You can nest lists or even other dictionaries as values, and update or access any part easily.

## 1 Create Dictionary

```
[1]: mydict = dict() # empty dictionary  
mydict
```

```
[1]: {}
```

```
[2]: mydict = {} # empty dictionary  
mydict
```

```
[2]: {}
```

```
[3]: mydict = {1:'one' , 2:'two' , 3:'three'} # dictionary with integer keys  
mydict
```

```
[3]: {1: 'one', 2: 'two', 3: 'three'}
```

```
[4]: mydict = dict({1:'one' , 2:'two' , 3:'three'}) # Create dictionary using dict()  
mydict
```

```
[4]: {1: 'one', 2: 'two', 3: 'three'}
```

```
[5]: mydict = {'A':'one' , 'B':'two' , 'C':'three'} # dictionary with character keys  
mydict
```

```
[5]: {'A': 'one', 'B': 'two', 'C': 'three'}
```

```
[6]: mydict = {1:'one' , 'A':'two' , 3:'three'} # dictionary with mixed keys  
mydict
```

```
[6]: {1: 'one', 'A': 'two', 3: 'three'}
```

```
[7]: mydict.keys() # Return Dictionary Keys using keys() method
```

```
[7]: dict_keys([1, 'A', 3])
```

```
[8]: mydict.values() # Return Dictionary Values using values() method
```

```
[8]: dict_values(['one', 'two', 'three'])
```

```
[9]: mydict.items() # Access each key-value pair within a dictionary
```

```
[9]: dict_items([(1, 'one'), ('A', 'two'), (3, 'three')])
```

```
[10]: mydict = {1:'one' , 2:'two' , 'A':['asif' , 'john' , 'Maria']} # dictionary
      ↪with
      mydict
```

```
[10]: {1: 'one', 2: 'two', 'A': ['asif', 'john', 'Maria']}
```

```
[12]: mydict = {1:'one' , 2:'two' , 'A':['asif' , 'john' , 'Maria'], 'B':('Bat' ,
      ↪'cat' , 'hat')}
      mydict
```

```
[12]: {1: 'one',
      2: 'two',
      'A': ['asif', 'john', 'Maria'],
      'B': ('Bat', 'cat', 'hat')}
```

```
[13]: mydict = {1:'one' , 2:'two' , 'A':{'Name':'asif' , 'Age' :20}, 'B':('Bat' ,
      ↪'cat' , 'hat')}
      mydict
```

```
[13]: {1: 'one',
      2: 'two',
      'A': {'Name': 'asif', 'Age': 20},
      'B': ('Bat', 'cat', 'hat')}
```

```
[14]: keys = {'a' , 'b' , 'c' , 'd'}
      mydict3 = dict.fromkeys(keys) # Create a dictionary from a sequence of keys
      mydict3
```

```
[14]: {'c': None, 'd': None, 'a': None, 'b': None}
```

```
[15]: keys = {'a' , 'b' , 'c' , 'd'}
      value = 10
      mydict3 = dict.fromkeys(keys , value) # Create a dictionary from a sequence of
      mydict3
```

```
[15]: {'c': 10, 'd': 10, 'a': 10, 'b': 10}
```

```
[16]: keys = {'a' , 'b' , 'c' , 'd'}
      value = [10,20,30]
      mydict3 = dict.fromkeys(keys , value) # Create a dictionary from a sequence of
      mydict3
```

```
[16]: {'c': [10, 20, 30], 'd': [10, 20, 30], 'a': [10, 20, 30], 'b': [10, 20, 30]}
```

```
[17]: value.append(40)
      mydict3
```

```
[17]: {'c': [10, 20, 30, 40],  
      'd': [10, 20, 30, 40],  
      'a': [10, 20, 30, 40],  
      'b': [10, 20, 30, 40]}
```

## 2 Accessing Items

```
[18]: mydict = {1:'one' , 2:'two' , 3:'three' , 4:'four'}  
mydict
```

```
[18]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
[19]: mydict[1] # Access item using key
```

```
[19]: 'one'
```

```
[20]: mydict.get(1) # Access item using get() method
```

```
[20]: 'one'
```

```
[21]: mydict1 = {'Name':'Asif' , 'ID': 74123 , 'DOB': 1991 , 'job' : 'Analyst'}  
mydict1
```

```
[21]: {'Name': 'Asif', 'ID': 74123, 'DOB': 1991, 'job': 'Analyst'}
```

```
[22]: mydict1['Name'] # Access item using key
```

```
[22]: 'Asif'
```

```
[23]: mydict1.get('job') # Access item using get() method
```

```
[23]: 'Analyst'
```

## 3 Add, Remove & Change Items

```
[24]: mydict1 = {'Name':'Akshay' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Germany'}  
mydict1
```

```
[24]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1991, 'Address': 'Germany'}
```

```
[25]: mydict1['DOB'] = 1992 # Changing Dictionary Items  
mydict1['Address'] = 'Delhi'  
mydict1
```

```
[25]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1992, 'Address': 'Delhi'}
```

```
[26]: dict1 = {'DOB':1995}
      mydict1.update(dict1)
      mydict1
```

```
[26]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1995, 'Address': 'Delhi'}
```

```
[27]: mydict1['Job'] = 'Analyst' # Adding items in the dictionary
      mydict1
```

```
[27]: {'Name': 'Akshay',
      'ID': 12345,
      'DOB': 1995,
      'Address': 'Delhi',
      'Job': 'Analyst'}
```

```
[28]: mydict1.pop('Job') # Removing items in the dictionary using Pop method
      mydict1
```

```
[28]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1995, 'Address': 'Delhi'}
```

```
[29]: mydict1.popitem() # A random item is removed
```

```
[29]: ('Address', 'Delhi')
```

```
[30]: mydict1
```

```
[30]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1995}
```

```
[31]: del[mydict1['ID']] # Removing item using del method
      mydict1
```

```
[31]: {'Name': 'Akshay', 'DOB': 1995}
```

```
[32]: mydict1.clear() # Delete all items of the dictionary using clear method
      mydict1
```

```
[32]: {}
```

```
[33]: del mydict1 # Delete the dictionary object
      mydict1
```

```
-----
NameError                                Traceback (most recent call last)
Cell In[33], line 2
      1 del mydict1 # Delete the dictionary object
----> 2 mydict1
```

```
NameError: name 'mydict1' is not defined
```

## 4 Copy Dictionary

```
[34]: mydict = {'Name': 'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Hilsinki'}  
mydict
```

```
[34]: {'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Hilsinki'}
```

```
[35]: mydict1 = mydict # Create a new reference "mydict1"
```

```
[36]: id(mydict) , id(mydict1) # The address of both mydict & mydict1 will be the same  
↪ same
```

```
[36]: (2788251765824, 2788251765824)
```

```
[37]: mydict2 = mydict.copy() # Create a copy of the dictionary
```

```
[38]: id(mydict2) # The address of mydict2 will be different from mydict because mydic
```

```
[38]: 2788251859968
```

```
[39]: mydict['Address'] = 'Mumbai'
```

```
[40]: mydict
```

```
[40]: {'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Mumbai'}
```

```
[41]: mydict1 # mydict1 will be also impacted as it is pointing to the same dictionary
```

```
[41]: {'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Mumbai'}
```

```
[42]: mydict2 # Copy of list won't be impacted due to the changes made in the original  
↪ original
```

```
[42]: {'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Hilsinki'}
```

## 5 Loop through a Dictionary

```
[44]: mydict1 = {'Name': 'Akshay' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Finland' }  
mydict1
```

```
[44]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1991, 'Address': 'Finland'}
```

```
[46]: for i in mydict1:  
    print(i , ':' , mydict1[i]) # Key & value pair
```

Name : Akshay  
ID : 12345  
DOB : 1991  
Address : Finland

```
[47]: for i in mydict1:  
       print(mydict1[i]) # Dictionary items
```

Akshay  
12345  
1991  
Finland

## 6 Dictionary Membership

```
[48]: mydict1 = {'Name': 'Akshay' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Finland' }  
mydict1
```

```
[48]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1991, 'Address': 'Finland'}
```

```
[49]: 'Name' in mydict1 # Test if a key is in a dictionary or not.
```

```
[49]: True
```

```
[50]: 'Akshay' in mydict1 # Membership test can be only done for keys.
```

```
[50]: False
```

```
[51]: 'ID' in mydict1
```

```
[51]: True
```

```
[52]: 'Address' in mydict1
```

```
[52]: True
```

## 7 All / Any

The all() method returns: ### True - If all keys of the dictionary are true ### False - If any key of the dictionary is false The any() function returns True if any key of the dictionary is True. If not, any() returns False.

```
[53]: mydict1 = {'Name': 'Akshay' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Finland' }  
mydict1
```

```
[53]: {'Name': 'Akshay', 'ID': 12345, 'DOB': 1991, 'Address': 'Finland'}
```

```
[54]: all(mydict1) # Will Return false as one value is false (Value 0)
```

```
[54]: True
```