```
a = {}
print(a)
print(type(a))
```

```
{}
<class 'dict'>
```

```
# DICTIONARY
#! Dicitionary is a combination of keys and values.
#! Keys and values in a doctionary are collectively known as items.
#! Changeable and do not allow duplicates.
#! Dictionaries are written with curly brackets, and have keys and
values.

a = {1:1,2:4,3:9,4:16,5:25}
print(a)                  # Output => {1: 1, 2: 4, 3: 9, 4: 16, 5: 25}
print(type(a))            # Output => <class 'dict'>
print(len(a))             # Output => 5

print(a.keys())           # Output => dict_keys([1, 2, 3, 4, 5])
print(a.values())         # Output => dict_values([1, 4, 9, 16, 25])
print(a.items())          # Output => dict_items([(1, 1), (2, 4), (3,
9), (4, 16), (5, 25)])
```

```
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25}
<class 'dict'>
5
dict_keys([1, 2, 3, 4, 5])
dict_values([1, 4, 9, 16, 25])
dict_items([(1, 1), (2, 4), (3, 9), (4, 16), (5, 25)])
```

```
# Accessing values with the help of keys.
print(a[5])
# Output => 25
print(a[3])
# Output => 9
```

```
25
9
```

```
# Updating values with the help of keys.
a[5] = 125
print(a)              # Output => {1: 1, 2: 4, 3: 9, 4: 16, 5: 125}

a[2] = 8
a[3] = 27
a[4] = 64
# All values are updated.
print(a)              # Output => {1: 1, 2: 8, 3: 27, 4: 64, 5: 125}

print(a.keys())       # Output => dict_keys([1, 2, 3, 4, 5])
```

```
print(a.values())    # Output => dict_values([1, 8, 27, 64, 125])
print(a.items())     # Output => dict_items([(1, 1), (2, 8), (3, 27),
(4, 64), (5, 125)])
```

```
{1: 1, 2: 4, 3: 9, 4: 16, 5: 125}
{1: 1, 2: 8, 3: 27, 4: 64, 5: 125}
dict_keys([1, 2, 3, 4, 5])
dict_values([1, 8, 27, 64, 125])
dict_items([(1, 1), (2, 8), (3, 27), (4, 64), (5, 125)])
```

```
# Adding values in a dictionary.
a[6] = 216
print(a)            # Output => {1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6:
216}
a[7] = 343
a[8] = 512
print(a)            # Output => {1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6:
216, 7: 343, 8: 512}
```

```
{1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6: 216}
{1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6: 216, 7: 343, 8: 512}
```

```
#get function => To access values with the help of keys.
print(a.get(4))     # Output => 64
print(a.get(7))     # Output => 343

#copy
b = a.copy()
print(b)            # Output => {1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6:
216, 7: 343, 8: 512}

#clear
a.clear()
print(a)            # Output => {}

#del => Deletes from the root
del a
del a         # Already deleted so, it shows an error.
                      # While 'clear' function doesn't show an error
after using again and again.
```

```
64
343
{1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6: 216, 7: 343, 8: 512}
{}

---------------------------------------------------------------------
-----
NameError                                 Traceback (most recent call
last)
c:\Users\PKVidyarthi\Desktop\Data Science\Notes\Code\Dictionary.ipynb
```

```
Cell 6 in <cell line: 15>()
     <a
href='vscode-notebook-cell:/c%3A/Users/PKVidyarthi/Desktop/Data
%20Science/Notes/Code/Dictionary.ipynb#ch0000005?line=12'>13</a> #del
=> Deletes from the root
     <a
href='vscode-notebook-cell:/c%3A/Users/PKVidyarthi/Desktop/Data
%20Science/Notes/Code/Dictionary.ipynb#ch0000005?line=13'>14</a> del a
---> <a
href='vscode-notebook-cell:/c%3A/Users/PKVidyarthi/Desktop/Data
%20Science/Notes/Code/Dictionary.ipynb#ch0000005?line=14'>15</a> del a

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