DICTIONARY

A dictionary is an unordered collection of items. It consists of Key-Value pairs separated by commas and those key-value pairs are called items.

Dictionaries are commonly used for mapping keys to values, allowing for efficient lookup and retrieval of data based on a unique key.

A dictionary is represented by curly braces "{ }" separated by commas (,) and expressed as a pair (key: value).

```
d={"name": "Sunny" }
```

The values can be of any data type and can repeat, keys must be of immutable type and must be unique.

Creating Dictionary:

```
d={ }
print(type(d))
O/P: <class dict>
```

Dictionary with Key-Value pair:

```
d={1:"keerthi","s.no": 22.0}
```

Accessing elements from the dictionary:

Indexing is used with other data types to access values a dictionary uses keys.

```
Eg: d={1:"keerthi","s.no": 22.0}
    print(d['s.no']}
    O/P: 22.0
```

Methods in Dictionary:

keys(): Return a new object of the dictionary's key

```
Eg: d={"name":"Kweet","phone":[80,08],"color":"White"}
    print(d.keys())
    O/P: dict_keys["name","phone","color"]
```

Values(): Returns a new object of the dictionary's values

```
Eg: d={"name":"Kweet","phone":[80,08],"color":"White"}
    print(d.Values())
    O/P: dict_values["Kweet",[80,08],"White"]
```

items(): Return a new object of the dictionary items in (key,Value) format.

```
Eg: d={"name":"Kweet","phone":[80,08],"color":"White"}
```

print(d.items())
O/P: dict_items([(‘name’:’Kweet’),(‘phone’:[80,08]),(‘color’:’white’)])

get(key[d]): Returns the value of the key. If the key does not exist,return d

Eg: d={“name”:”Kweet”,”phone”:[80,08],”color”:”White”}
print(d.get(“name”))
O/P: Kweet

update([other]): Updates the dictionary with the key/value pairs from other,Over writing existing keys.

Eg: d={“name”:”Kweet”,”phone”:[80,08],”color”:”White”}
print(d.update(“age”:28))
O/P: {“name”:”Kweet”,”phone”:[80,08],”color”:”White”,”age”:28}

Copy(): Return a shallow copy of the dictionary

Eg: d={“name”:”Kweet”,”phone”:[80,08],”color”:”White”,”age”:28}
v=d.copy()
print(v)
O/P: {“name”:”Kweet”,”phone”:[80,08],”color”:”White”,”age”:28}

Clear(): Removes all items from the Dictionary

Eg: d={“name”:”Kweet”,”phone”:[80,08],”color”:”White”,”age”:28}
v=d.copy()
print(v)
O/P: {“name”:”Kweet”,”phone”:[80,08],”color”:”White”,”age”:28}

<u>Nested Dictionary:</u>

A nested dictionary is a dictionary that contains other dictionary as it’s values. This allows for a hierarchical or nested structure where each key maps to another dictionary.

Eg: NesDict={

```
    ‘Person1’:{
            ‘name’:’Rob’,
            ‘Age’:30
            ‘city’:’BNG’ },
    ‘Person2’::{
            ‘name’:’Bob’,
            ‘Age’:35
            ‘city’:’HYD’ }
            }
```

To access the elements in nested dictionary using multiple key accesss.

Eg: print(NesDict[‘Person2’][‘Age’])
O/P: 35

Iterating using Dictionary:

We can iterate through each key in a dictionary by using for loop

```
Eg: d={“name”:”Kweet”,”phone”:[80,08],”color”:”White”,”age”:28}
    For i in d:
        print(i)
O/P: name
    Phone
    Color
    Age
```

```
Eg2: d={“name”:”Kweet”,”phone”:[80,08],”color”:”White”,”age”:28}
        For i in d.items():
                print(i)
        O/P: name Kweet
              Phone [80,08]
              Color White
              Age  28
```