# ASSIGNMENT(DICTIONARY)

**1) what is dictionary and write all dictionary functions with example.**

**Ans)** If we want to represent a group of objects as key-value pairs then we should go for Dictionary.

**Functions of Dictionary:**

**dict():**

To create a dictionary

**d=dict()** ===>It creates empty dictionary

**d=dict({1:"python",2:"java"})** ==>It creates dictionary with specified elements **d=dict([(1,"python"),(2,"java"),(3,"c")])**==>It creates dictionary with the given list of tuple element

**len():** Returns the number of items in the dictionary

**clear():** To remove all elements from the dictionary

**get():** To get the value associated with the key

**d.get(key)** If the key is available then returns the corresponding value otherwise returns None.It wont raise any error.

**d.get(key,defaultvalue)** If the key is available then returns the corresponding value otherwise returns default value.

**Eg: d={1:"pavan",2:"sai",3:"krishna"}**

print(d[1]) ==>pavan

print(d[4]) ==>KeyError:4

print(d.get(1)) ==pavan

print(d.get(4)) ==>None

print(d.get(1,"Guest")) ==pavan

print(d.get(4,"Guest")) ==>Guest

**pop():**

**d.pop(key)** It removes the entry associated with the specified key and returns the corresponding value If the specified key is not available then we will get KeyError

**Eg:**

1) **d={1:"pk",2:"rc",3:"aa"}**

2) print(d.pop(100))

3) print(d)

4) print(d.pop(4))

5) Output

6) pk

7) {2: 'rc', 3: 'aa'}

8) KeyError: 4

4. popitem():

It removes an arbitrary item(key-value) from the dictionaty and returns it.

Eg: 1) **d={100:"durga",200:"ravi",300:"shiva"}**

2) print(d)

3) print(d.popitem())

4) print(d)

6) Output

7) {100: 'durga', 200: 'ravi', 300: 'shiva'}

8) (300, 'shiva')

9) {100: 'durga', 200: 'ravi'}

If the dictionary is empty then we will get KeyError

d={}

print(d.popitem()) ==>KeyError: 'popitem(): dictionary is empty'

**keys():**

It returns all keys associated eith dictionary

Eg: 1) **d={100:"durga",200:"ravi",300:"shiva"}**

2) print(d.keys())

3) for k in d.keys():

4) print(k)

6) Output

7) dict\_keys([100, 200, 300])

8) 100

9) 200

10) 300

**values():**

It returns all values associated with the dictionary

Eg: 1. **d={100:"durga",200:"ravi",300:"shiva"}**

2. print(d.values())

3. for v in d.values():

4. print(v)

6. Output

7. dict\_values(['durga', 'ravi', 'shiva'])

8. durga

9. ravi

10. shiva

**items():**

It returns list of tuples representing key-value pairs.

**[(k,v),(k,v),(k,v)]**

Eg: 1. **d={100:"durga",200:"ravi",300:"shiva"}**

2. for k,v in d.items():

3. print(k,"--",v)

4. Output

6. 100 – durga

7. 200 – ravi

8. 300 -- shiva

**copy():**

To create exactly duplicate dictionary(cloned copy)

d1=d.copy();

**setdefault():**

d.setdefault(k,v)

If the key is already available then this function returns the corresponding value. If the key is not available then the specified key-value will be added as new item to the dictionary

Eg: 1**. d={100:"durga",200:"ravi",300:"shiva"}**

2. print(d.setdefault(400,"pavan"))

3. print(d)

4. print(d.setdefault(100,"sachin"))

5. print(d)

7. Output

8. pavan

9. {100: 'durga', 200: 'ravi', 300: 'shiva', 400: 'pavan'} 10. durga 11. {100: 'durga', 200: 'ravi', 300: 'shiva', 400: 'pavan'}

**update():**

d.update(x)

All items present in the dictionary x will be added to dictionary d