**Name: Tausif Khan**

**Rollno: 12**

**Branch: SE/COMPS**

**Div: B**

**Course Name: Open Source Technology Lab (OSTL)**

**Experiment No: 3b.**

**AIM:** Write a menu driven program to demonstrate use of dictionary in python:

1. Concatenate an item in the existing dictionary.

2. Delete item of the existing dictionary.

3. Retrieve all keys from a dictionary

4. Retrieve all values from a dictionary

5. Retrieve all key-value pairs from a dictionary

6. Find a key and print its value.

**TOOLS USED**: Python 3.4.3, Terminal

**THEORY:**

**1. What is dictionary in Python?**

Python dictionary is an unordered collection of items. While other compound data types have only value as an element, a dictionary has a key: value pair.

Dictionaries are optimized to retrieve values when the key is known.

## How to create a dictionary?

Creating a dictionary is as simple as placing items inside curly braces {} separated by comma.

An item has a key and the corresponding value expressed as a pair, key: value.

While values can be of any data type and can repeat, keys must be of immutable type (string, number or tuple with immutable elements) and must be unique.

# empty dictionary

my\_dict = {}

# dictionary with integer keys

my\_dict = {1: 'apple', 2: 'ball'}

# dictionary with mixed keys

my\_dict = {'name': 'John', 1: [2, 4, 3]}

# using dict()

my\_dict = dict({1:'apple', 2:'ball'})

# from sequence having each item as a pair

my\_dict = dict([(1,'apple'), (2,'ball')])

**2. Explain and describe different methods in dictionary with example.**

Methods that are available with dictionary are tabulated below. Some of them have already been used in the above examples.

|  |  |  |
| --- | --- | --- |
| Method | Example | Description |
| clear() | d.clear() | Remove all items form the dictionary. |
| copy() | d1 = d.copy() | Return a shallow copy of the dictionary. |
| fromkeys() | d.fromkeys(seq [,v]) | Return a new dictionary with keys from seq and value equal to v (defaults to None). |
| get() | d.get(key[,d]) | Return the value of key. If key doesnot exit, return d (defaults to None). |
| items() | d.items() | Return a new view of the dictionary's items (key, value). |
| keys() | d.keys() | Return a new view of the dictionary's keys. |
| values() | d.values() | Return a new view of the dictionary's values. |
| update() | d.update(x) | Adds all elements from dictionary ‘x’ to ‘d’. |
| pop() | d.pop[(key[,d])](https://www.programiz.com/python-programming/methods/dictionary/pop) | Remove the item with key and return its value or d if key is not found. If d is not provided and key is not found, raises KeyError. |
| setdefault() | d. setdefault(key[,d]) | If key is in the dictionary, return its value. If not, insert key with a value of d and return d (defaults to None). |

# **3. How to display elements in dictionary using for loop.**

# You can loop through a dictionary by using a for loop.

When looping through a dictionary, the return value are the keys of the dictionary, but there are methods to return the values as well.

### **Example**

Print all key names in the dictionary, one by one:

for x in thisdict:  
  print(x)

### **Example**

Print all values in the dictionary, one by one:

for x in thisdict:  
  print(thisdict[x])

**Full example:-**

Code:-

def main():

stocks = {

'Apple': 146.48,

'Mango':44.11,

'Grapes':25.54

}

#print out all the keys

for c in stocks:

print(c)

#print key n values

for k, v in stocks.items():

print("Key : {0}, Value : {1}".format(k, v))

if \_\_name\_\_ == '\_\_main\_\_':

main()

**PROGRAM:**

dict={'Name':'Tausif','Rollno':12}

def mainMenu():

selection=int(input("Enter choice"))

if selection==1:

concatenate()

elif selection==2:

pop()

elif selection==3:

keys()

elif selection==4:

values()

elif selection==5:

items()

elif selection==6:

setdefault()

else:

print("Enter a valid Selection")

mainMenu()

def concatenate():

dict1={"Div":'B'}

print('Concatenate item to dictionary: ',dict.update(dict1))

print(dict)

def pop():

print('Delete item from dictionary: ',dict.pop('Name'))

print(dict)

def keys():

print('keys in dictionary: ',dict.keys())

def values():

print('Values in dictionary: ',dict.values())

def items():

print('Print dictionary items: ',dict.items())

def setdefault():

x = dict.setdefault("Name", "Mango")

print('Value returned: ',x)

mainMenu()

**Output-**

Enter choice 1

Concatenate item to dictionary: None

{'Name': 'Tausif', 'Rollno': 12, 'Div': 'B'}

Enter choice 2

Delete item from dictionary: Tausif

{'Rollno': 12}

Enter choice 3

keys in dictionary: dict\_keys(['Name', 'Rollno'])

Enter choice 4

Values in dictionary: dict\_values(['Tausif', 12])

Enter choice 5

Print dictionary items: dict\_items([('Name', 'Tausif'), ('Rollno', 12)])

Enter choice 6

Value returned: Tausif

**CONCLUSION:**

Thus we have studied and implement menu driven program using dictionary.