Python Dictionary

Define

The values that the keys point to can be any Python data type.

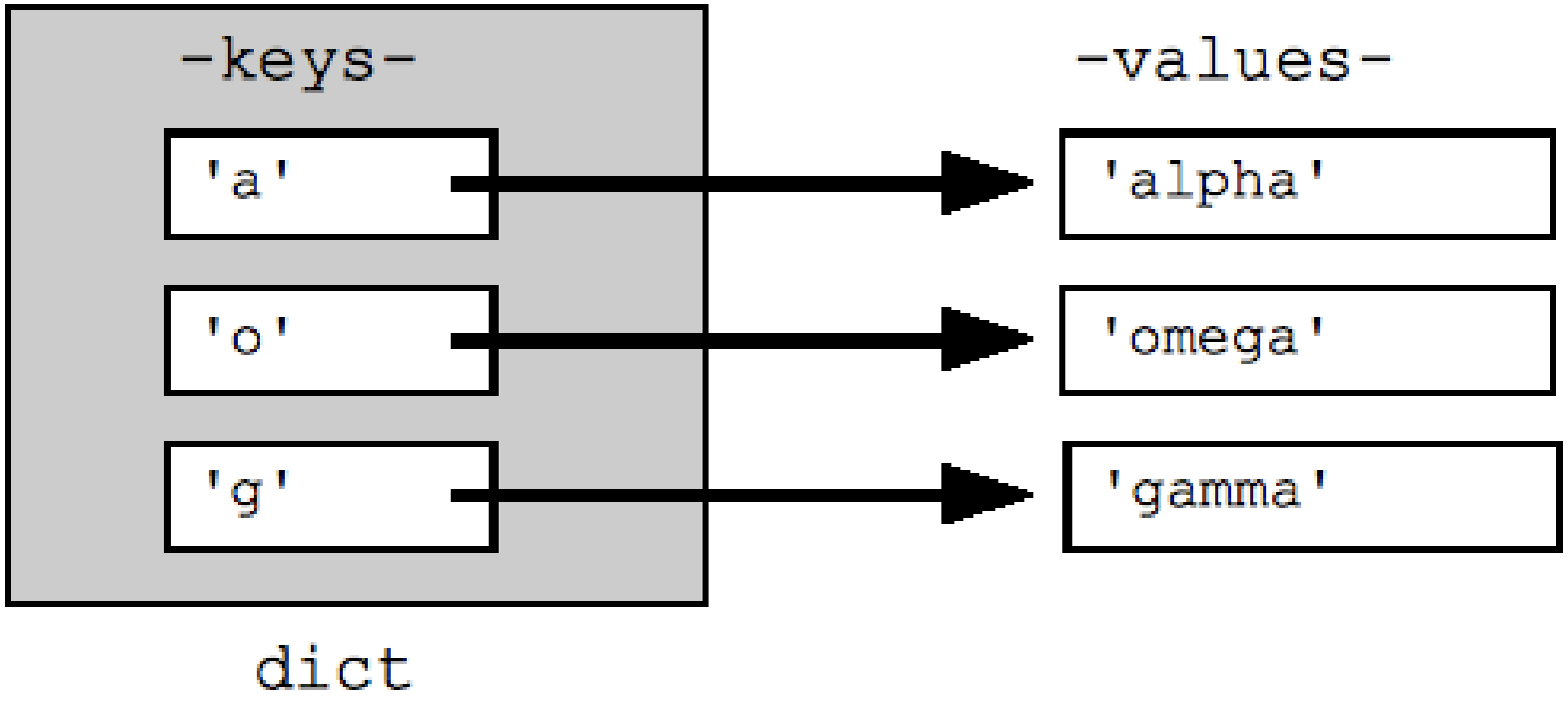
Dictionary is a data type.

Define

A Python dictionary is a mapping of unique keys to values.

Dictionaries are mutable, which means they can be changed.

Dictionary setup



Brackets used

{ } define

[ ] reference

( ) object /function

Dictionary

Consist of a KEY and Value

KEY -- MUST be unique

Value - can be any valid data type

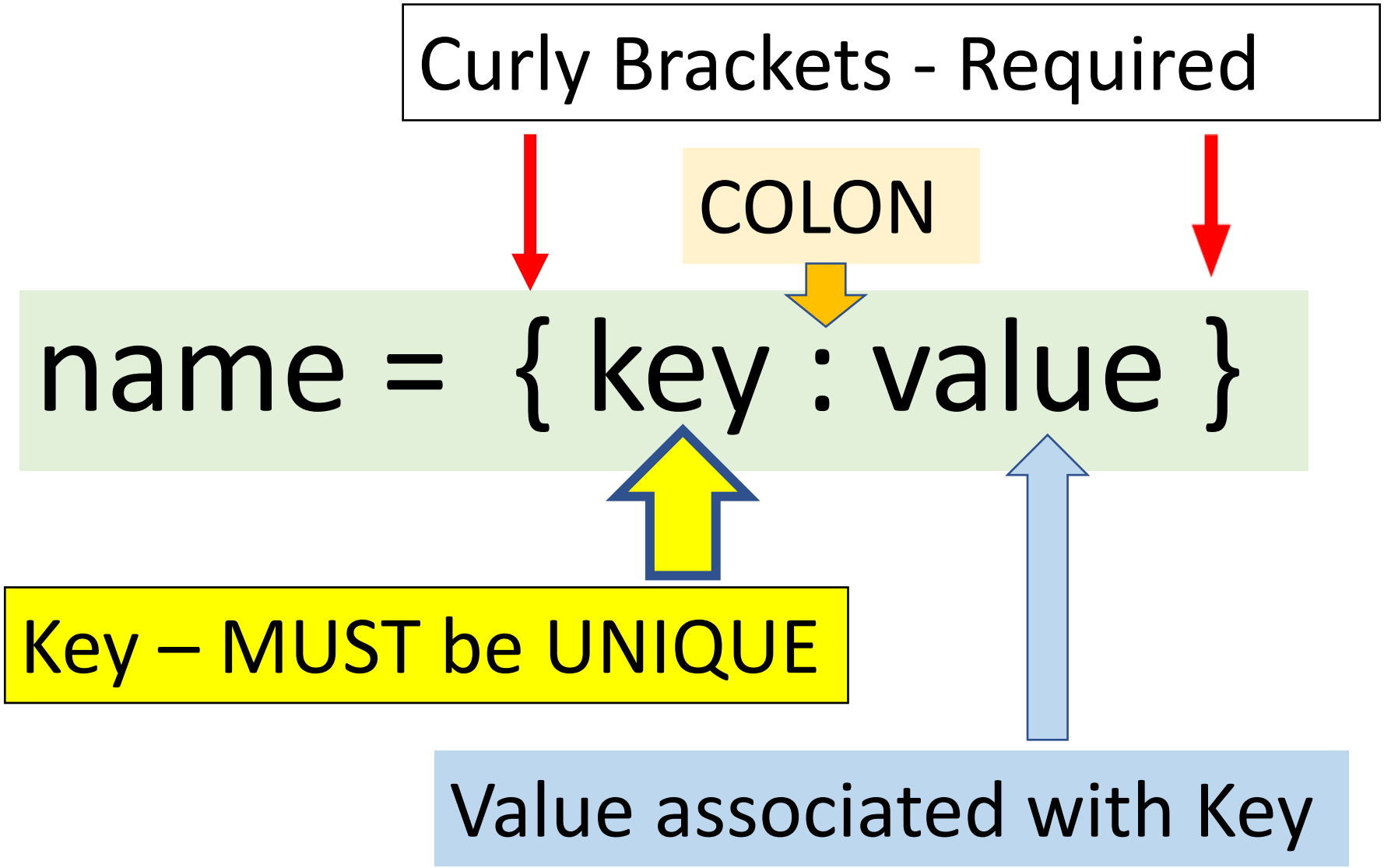
Key notes

The dictionary key does not have to be in sequence.

Just has to be unique

(no duplicates)

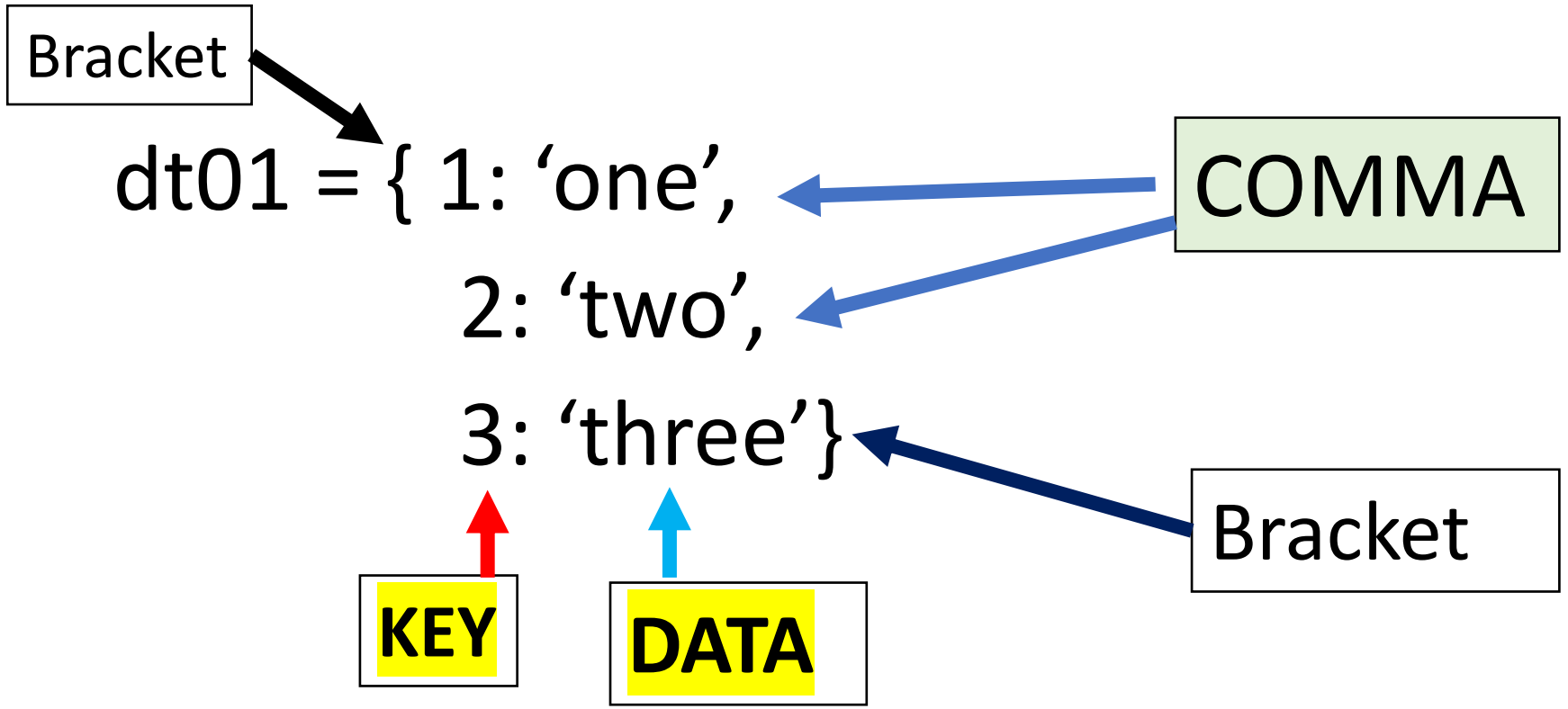
Dictionary Structure



Create Dictionary

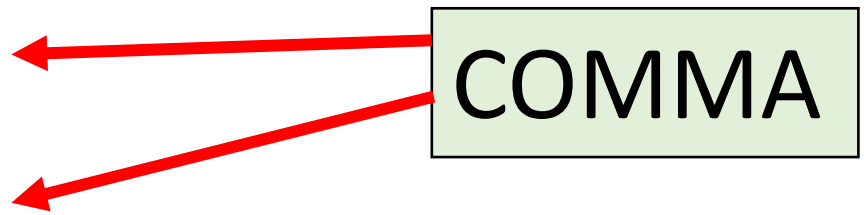
Dictionary Structure
Multiple Entries

Separate with commas



Dictionary Structure
Multiple Entries

```
dt01 = { 1: 'one',
         2: 'two',
         3: 'three'}
```



Note:

Only one pair of curly brackets,

entries come in pairs (key:value)

Try This

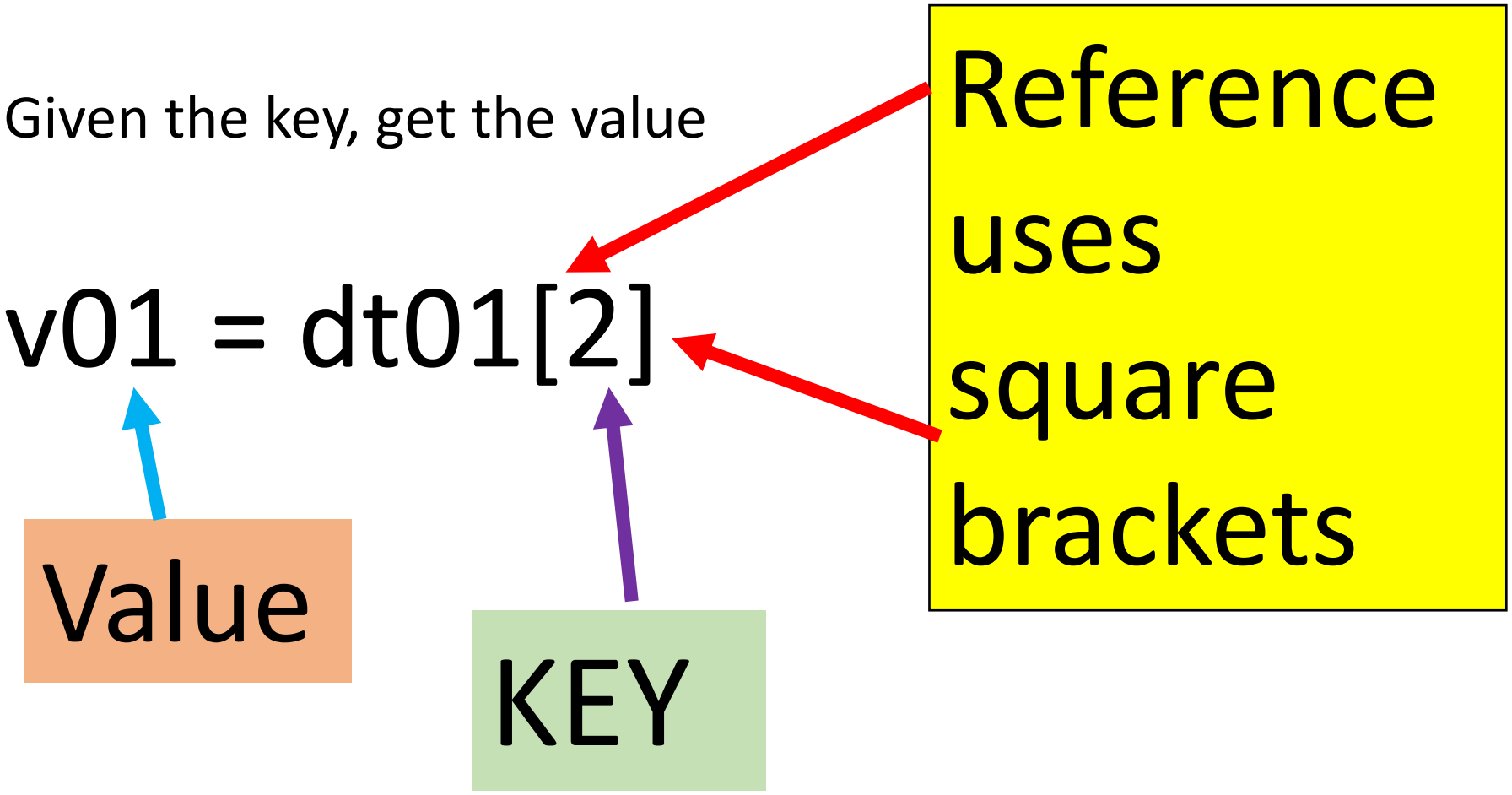
```
def main():
    # set up dictionary
    dt01 = { 1: 'one',
             2: 'two',
             3: 'three’,
             4: ‘four’}
    print('Dictionary example')
    print(dt01)
main()
```

dt01

Referencing a dictionary

Referencing a dictionary

Given the key, get the value



TRY THIS

```
def main():
    # set up dictionary
    dt01 = { 1: 'one',
             7:'seven',
              2: 'two',
              3: 'three',
              4: 'four',
              5: 'five'}
    print('Dictionary example')
    print(dt01)
    v01 = dt01[7]
    print('Value for 7 is: ',v01)
    a=4
    v02 = dt01[a]
    print('Value for ', a, ' is: ',v02)
main()
```

dt02

Print Dictionary

Straight print

print(dt01)

Prints the whole dictionary
In key, value format

Use iteration (for loop)

Use: for variable in dictionary

Add this to program:

```
for v99 in dt01:
    print(v99, ' ',dt01[v99])
```

values only

Add to program:

```
print('----------')
for v88 in dt01:
    print(dt01[v88])
```

Or

```
for v77 in dt01.values():
    print(v77)
```

Add to PGM

```
def main():
    # set up dictionary
    dt01 = { 1: 'one',
             2: 'two',
             3: 'three',
             4: 'four'}
    print('Dictionary example')
    print(dt01)
    v01 = dt01[2]
    print('Value for 2 is: ',v01)
    a=4
    v02 = dt01[a]
    print('Value for ', a, ' is: ',v02)
    print('----------')
    for v88 in dt01:
        print(v88, ' - ',dt01[v88])
main()
```

Add to Dictionary

Add to dictionary

Need: Key and value

Format: dt01[k] = v

Caution: If key already exist, the value will replace the value

Add to PGM

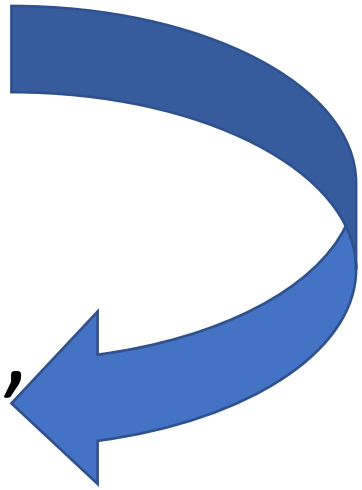
```
def main():
    # set up dictionary
    dt01 = { 1: 'one',
             2: 'two',
             3: 'three',
             4: 'four'}
    print('Dictionary example')
    print(dt01)
    a = 10
    b = ‘ten'
    dt01[a] = b
    print(dt01)
main()
```

Change value in dictionary

Change value in dictionary

Add to Program:

```
dt01[8] = 'ate'
print(dt01)
print('-------')
dt01[8] = 'eight'
print(dt01)
```



Change is destructive
Cannot go back to previous value

Add to PGM

```
def main():
    # set up dictionary
    dt01 = { 1: 'one',
             2: 'two',
             4: 'four'}
    print('Dictionary example')
    print(dt01)
    a = 5
    b = 'five'
    dt01[a] = b
    print(dt01)
    input('enter key to continue')
    dt01[8] = 'ate'
    print(dt01)
    input('enter key to continue')
    dt01[8] = 'eight'
    print(dt01)
main()
```

dt05

Delete dictionary entry

Form: del name(KEY)

del dt01[8]

Try this

```
def main():
    # set up dictionary
    dt01 = { 1: 'one', 2: 'two', 4: 'four'}
    print('Dictionary example')
    print(dt01)
    a = 5
    b = 'five'
    dt01[a] = b
    print(dt01)
    input('enter key to continue')
    dt01[8] = 'eight'
    print(dt01)
    input('enter key to continue')
    del dt01[8]
    print (dt01)
main()
```

Error Handling: KeyError trap with try & except

The KeyError is raised when the KEY in the dictionary is NOT FOUND

Try this

```
def test1():
    dt01={1:'one',2:'two',4:'four'}
    print('Dictionary example')
    print(dt01)
    del dt01[8]
    print (dt01)
def test2():
    a= 8
    dt01={1:'one',2:'two',4:'four'}
    print('Dictionary example')
    print(dt01)
    try:
       del dt01[a]
    except:
        print('Key: ',a,' not found in dictionary')
    print (dt01)
def main():
    test1()
    test2()
main()
```

dt07

TIP

Instead of print('-----')

Use:

agn = input('Hit enter key to continue')

or

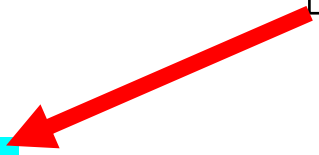
input('hit enter to continue')

Operations

Find number of elements (sets) in dictionary

Add to program:

Use len function



```
zz = len(dt01)
print('Number of sets is: ', zz)
```

TRY THIS

```
def main():
    dt01={1:'one',2:'two',3:'three',4: 'four'}
    dt02={'A':1,'B':2,'C':3,'D':4,'E':5,'F':6}
    print('Dictionary example')
    print('----dt01 ----')
    print(dt01)
    input('Enter key to continue')
    zz = len(dt01)
    print('Number of sets in dt01 is: ', zz)
    input('Enter key to continue')
    print('----dt02 ----')
    print(dt02)
    input('Enter key to continue')
    zz = len(dt02)
    print('Number of sets in dt02 is: ', zz)
main()
```

Check if key exists

Form:

if key in dictionary:

else:

TRY THIS

```
def main():
    dt01={1:'one',2:'two',3:'three',
          4:'four',7:'seven',10: 'ten'}
    print('Dictionary example')
    print(dt01)
    a=1
    while a > 0:
        a = int(input('enter a number: '))
        if a in dt01:
            print(a, ' in dictionary')
        else:
            print(a, ' not in dictionary')
main()
```

Check if key does not exist

Form:

if key not in dictionary:

else:

TRY THIS

```
def main():
    dt01={1:'one',2:'two',3:'three',
          4:'four',7:'seven',10: 'ten'}
    print('Dictionary example')
    print(dt01)
    a=1
    while a > 0:
        a = int(input('enter a number: '))
        if a not in dt01:
            print(a, ' not in dictionary’)
        else:
            print(a, ' in dictionary')
main()
```

Define empty dictionary

dt01 ={}

You can build dictionary using a loop

Load / build dictionary

```
def main():
    dt01={}
    akey = 'x'
    print('hit enter key to quit')
    while akey !='':
        akey = input('Enter key value: ')
        if len(akey) == 0:
            done = 0
        else:
            avalue =input('Enter value for key: ')
            dt01[akey]=avalue
    print(' ------------------------')
    print('number of entries is: ',len(dt01))
    print('\n',dt01)
main()
```

more

More functions

1 dict.clear()

Removes all elements of dictionary dict

2 dict.copy()

Returns a shallow copy of dictionary dict

3 dict.keys()

Returns list of dictionary dict's keys

4 dict.values()

Returns list of dictionary dict's values

TRY THIS

```
def main():
    dt02={'A':1,'B':2,'C':3,'D':4,'E':5,'F':6}
    print('Dictionary example')
    print(dt02,'\n')
    print(dt02.keys(),'\n')
    print(dt02.values(),'\n')
    input('enter key to continue')
    dt99=dt02.copy()
    print('dt99 copy of dt02')
    print(dt99)
    input('enter key to continue')
    dt02.clear()
    print('dt02 is cleared')
    print(dt02)
main()
```

done