Python Programming

EDUCATION FOR PEACE

(ESTD-1995)

RGM College of Engineering & Technology (Autonomous)

Department of Computer Science & Engineering

Academic Year : 2020-2021

SET DATA TYPE - 2

Guido Van Rossum

Learning Mantra

If you really strong in the basics, then remaining things will become so easy.

Agenda:

1. Important functions / methods of set datatype

   i. copy()

   ii. pop()

   iii. remove()

   iv. discard()

   v. clear()

3. copy():

- Returns copy of the set. It is cloned object (Backup copy).

Eg:

```
s={10,20,30}

s1=s.copy()

print(s1)

print(s)
```

Output:

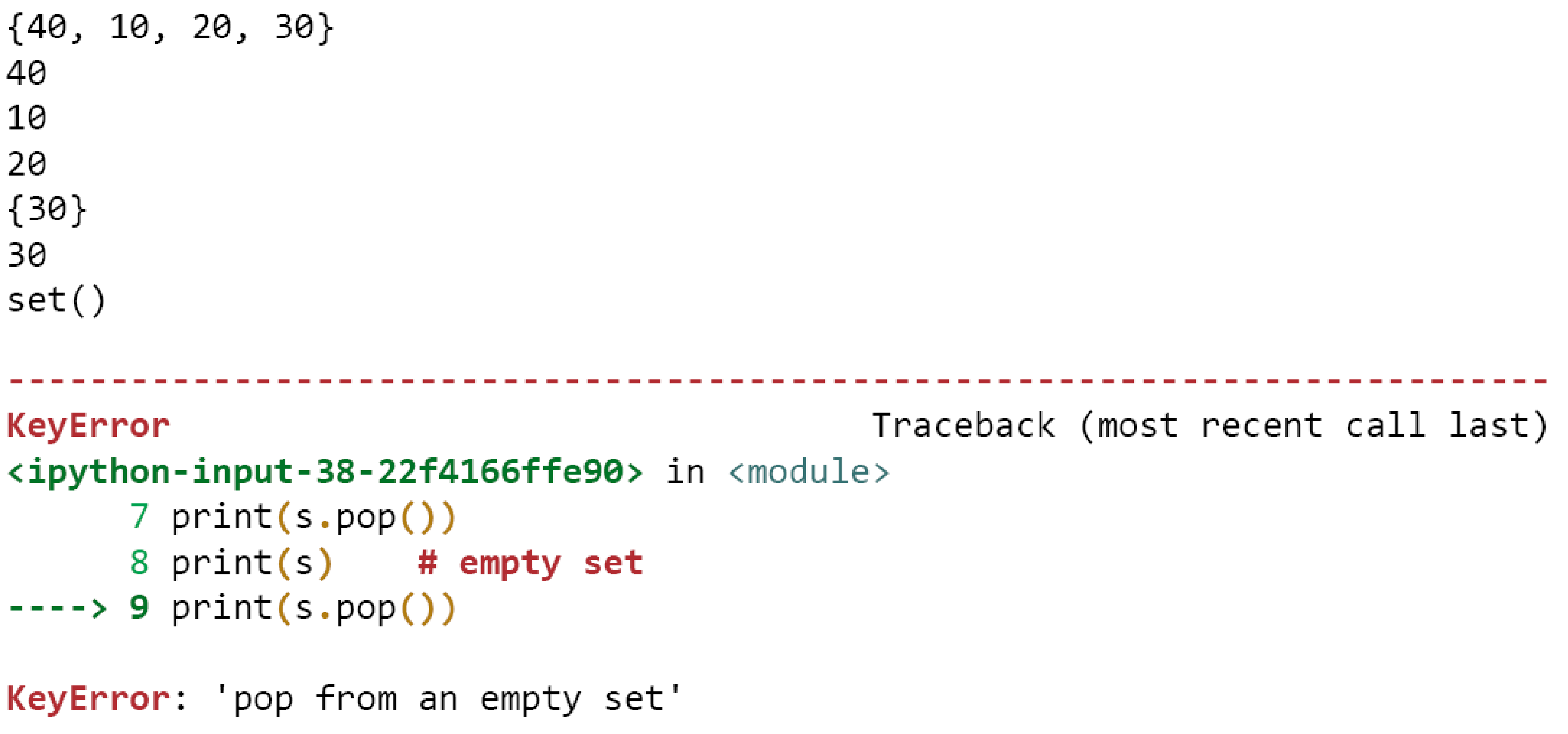
```
{10, 20, 30}

{10, 20, 30}
```

4. pop():

- It removes and returns some random element from the set.

Eg:

```
s={40,10,30,20}

print(s)

print(s.pop())

print(s.pop())

print(s.pop())

print(s)

print(s.pop())

print(s)          # empty set

print(s.pop())
```

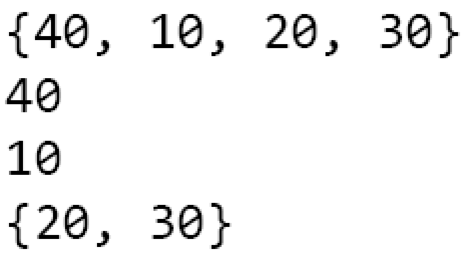


```
{40, 10, 20, 30}
40
10
20
{30}
30
set()

---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-38-22f4166ffe90> in <module>
      7 print(s.pop())
      8 print(s)    # empty set
----> 9 print(s.pop())

KeyError: 'pop from an empty set'
```

Consider the following case,

I.

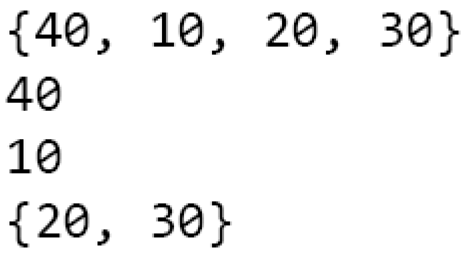
```
s={40,10,30,20}
print(s)
print(s.pop())
print(s.pop())
print(s)
```



```
{40, 10, 20, 30}
40
10
{20, 30}
```

II.

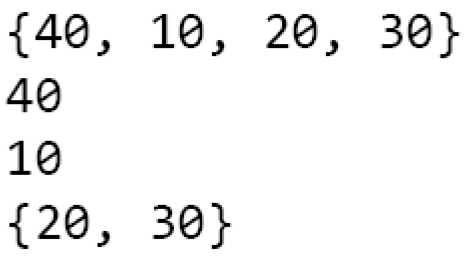
```
s={40,10,30,20}
print(s)
print(s.pop())
print(s.pop())
print(s)
```



```
{40, 10, 20, 30}
40
10
{20, 30}
```

III.

```
s={40,10,30,20}
print(s)
print(s.pop())
print(s.pop())
print(s)
```



```
{40, 10, 20, 30}
40
10
{20, 30}
```

Key point to observe:

How many times you may execute the code, the elements which are popped from the set in same order. The reason is ---

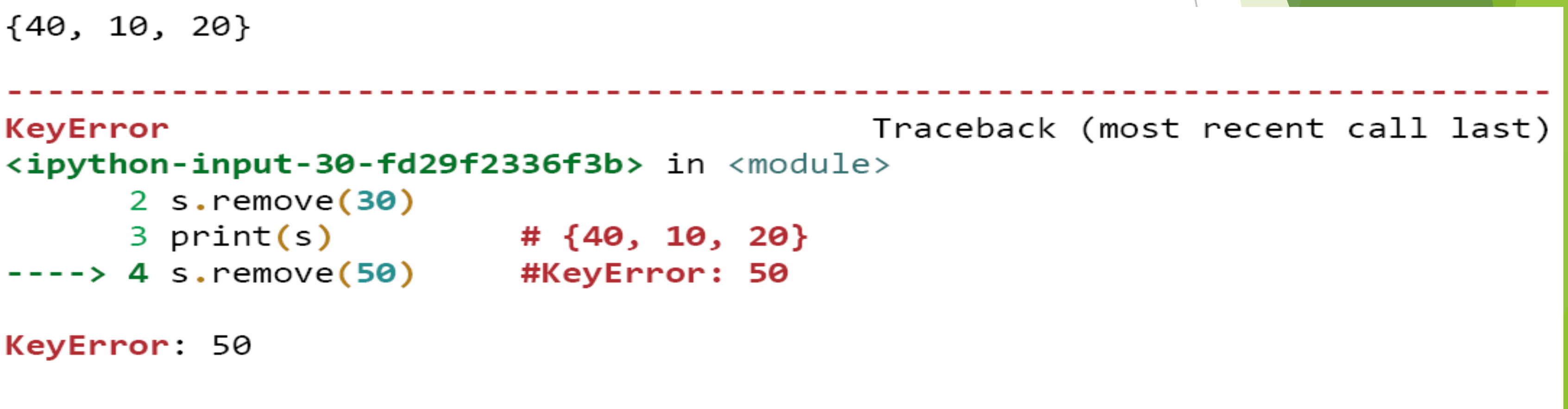
- All the elements of set are inserted based on some hash code. If that order is fixed then it is always going to return one by one. But in which order these elements are inserted we don't know.

5. remove(x):

- It removes specified element from the set.
- If the specified element not present in the set, then we will get **KeyError**.

Eg:

```
s={40,10,30,20}

s.remove(30)

print(s) # {40, 10, 20}

s.remove(50) # KeyError: 50
```



```
{40, 10, 20}

---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-30-fd29f2336f3b> in <module>
      2 s.remove(30)
      3 print(s)          # {40, 10, 20}
----> 4 s.remove(50)      #KeyError: 50

KeyError: 50
```

6. discard(x):

- It removes the specified element from the set.
- If the specified element not present in the set then we won't get any error.

Eg:

```
s={10,20,30}
s.discard(10)
print(s)            #{20, 30}
s.discard(50)
print(s)            #{20, 30}
```

Output:

```
{20, 30}
{20, 30}
```

Answer the following questions :

Q. What is the difference between remove() and discard() functions in Set?

Q. Explain differences between pop(),remove() and discard() functions in Set?

7.clear():

- To remove all elements from the Set.

Eg:

```
s={10,20,30}
print(s)
s.clear()
print(s)
```

Output:

```
{10, 20, 30}
set( )
```

Any question?

If you try to practice programs yourself, then you will learn many things automatically

Spend few minutes and then enjoy the study

Thank You