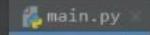


```
# collection = single "variable" used to store multiple values
#   List  = [] ordered and changeable. Duplicates OK
#   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
#   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
```

```
1 # collection = single "variable" used to store multiple values
2 #   List  = [] ordered and changeable. Duplicates OK
3 #   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6 fruits = ["apple", "orange", "banana", "coconut"]
7
8 print(fruits[1])
```

```
main.py x
1 # collection = single "variable" used to store multiple values
2 #     List  = [] ordered and changeable. Duplicates OK
3 #     Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #     Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6 fruits = ["apple", "orange", "banana", "coconut"]
7
8 print(fruits[::-1])
```

```
n: main x
C:\Users\HP\PycharmProjects\practice\venv\bin\python.exe C:/Users/HP/PycharmProjects/practice/main.py
['coconut', 'banana', 'orange', 'apple']
```



main.py

```
1 # collection = single "variable" used to store multiple values
2 #     List  = [] ordered and changeable. Duplicates OK
3 #     Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #     Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6 fruits = ["apple", "orange", "banana", "coconut"]
7
8 print(fruits[::2])
```

```
# collection = single "variable" used to store multiple values
#   List  = [] ordered and changeable. Duplicates OK
#   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
#   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
```

```
fruits = ["apple", "orange", "banana", "coconut"]
```

```
# print(fruits[0])
```



```
for fruit| in fruits:
    print(x)
```

```
Pr... main.py
practice C:\Users\HP\PycharmProjects\practice\venv\Scripts\python.exe C:/Users/HP/PycharmProjects/practice/main.py
> venv
main.py
External Libraries
Scratches and Consoles

1 # collection = single "variable" used to store multiple values
2 #   List  = [] ordered and changeable. Duplicates OK
3 #   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6 fruits = ["apple", "orange", "banana", "coconut"]
7 print(dir(fruits))
8
9 # print(fruits[0])
10 # for fruit in fruits:
11     # print(fruit)
```

```
run: main x
['__add__', '__class__', '__class_getitem__', '__contains__', '__delattr__', '__delitem__', '__dir__', '__doc__',
 __eq__', '__format__', '__ge__', '__getattribute__', '__gt__', '__hash__', '__init__', '__iter__', '__le__', '__lt__',
 '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__sizeof__', '__str__', '__subclasshook__']
```

```
Process finished with exit code 0
```

```
1 # collection = single "variable" used to store multiple values
2 #   List  = [] ordered and changeable. Duplicates OK
3 #   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6     fruits = ["apple", "orange", "banana", "coconut"]
7     # print(dir(fruits))
8     # print(help(fruits))
9     # print(len(fruits))
10    print("pineapple" in fruits)
11
12    # print(fruits[0])
```

main.py

main.py

C:\Users\HP\Pycharm

.py
Libraries
es and Consoles

```
10     # print("pineapple" in fruits)
11
12     # fruits[0] = "pineapple"
13     # fruits.append("pineapple")
14     # fruits.remove("apple")
15     # fruits.insert(0, "pineapple")
16     # fruits.sort()
17     # fruits.reverse()
18     # fruits.clear()
19     # print(fruits.index("apple"))
20     print(fruits.count("pine|"))
21
22     print(fruits)
```

```
main.py
1  # collection = single "variable" used to store multiple values
2  #   List  = [] ordered and changeable. Duplicates OK
3  #   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4  #   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6  fruits = ["apple", "orange", "banana", "coconut"]
7  # print(dir(fruits))
8  # print(help(fruits))
9  # print(len(fruits))
10 # print("pineapple" in fruits)
11
12 # fruits[0] = "pineapple"
13 # fruits.append("pineapple")
```

main.py

```
1 # collection = single "variable" used to store multiple values
2 #   List  = [] ordered and changeable. Duplicates OK
3 #   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6     fruits = {"apple", "orange", "banana", "coconut"}
7 # print(dir(fruits))
8 # print(help(fruits))
9 # print(len(fruits))
10 # print("pineapple" in fruits)
11
12 # fruits[0] = "pineapple"
13 # fruits.append("pineapple")
```

Notifications

practice / main.py



Project
practice C:\Users\HP\PycharmProjects\practice
venv
External Libraries
Scratches and Consoles

```
1 # collection = single "variable" used to store multiple values
2 #     List  = [] ordered and changeable. Duplicates OK
3 #     Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #     Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6     fruits = {"apple", "orange", "banana", "coconut"}
7     # print(dir(fruits))
8     # print(help(fruits))
9     # print(len(fruits))
10    # print("pineapple" in fruits)
11
12    print(fruits)
13    #for fruit in fruits:
```

Run: main



```
{'orange', 'apple', 'banana', 'coconut'}
```



The screenshot shows a PyCharm IDE interface with a dark theme. The top bar includes tabs for 'main.py' and 'main', along with various icons for file operations like save, run, and search. On the left, there's a 'Project' sidebar showing a 'practice' folder containing 'main.py', 'venv', and 'External Libraries'. A 'Scratches and Consoles' section is also present. The main code editor window displays the following Python script:

```
1 # collection = single "variable" used to store multiple values
2 #     List  = [] ordered and changeable. Duplicates OK
3 #     Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4 #     Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6     fruits = {"apple", "orange", "banana", "coconut"}
7     # print(dir(fruits))
8     # print(help(fruits))
9     # print(len(fruits))
10    # print("pineapple" in fruits)
11
12    print(fruits[0])
13
```

The line `print(fruits[0])` is highlighted in blue. In the bottom-left 'Run' panel, there's a 'main' tab and a 'Bookmarks' section. The run output shows the command `print(fruits[0])` followed by the error message `TypeError: 'set' object is not subscriptable`.

You cannot use indexing on a set because it is unordered.

main.py

main

```
5
6     fruits = {"apple", "orange", "banana", "coconut"}
7     # print(dir(fruits))
8     # print(help(fruits))
9     # print(len(fruits))
10    # print("pineapple" in fruits)
11
12    # fruits.add("pineapple")
13    # fruits.remove("apple")
14    # fruits.pop()
15    fruits.clear()
```

```
1  # collection = single "variable" used to store multiple values
2  #   List  = [] ordered and changeable. Duplicates OK
3  #   Set   = {} unordered and immutable, but Add/Remove OK. NO duplicates
4  #   Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6  fruits = ("apple", "orange", "banana", "coconut", "coconut")
7  # print(dir(fruits))
8  # print(help(fruits))
9  # print(len(fruits))
10 # print("pineapple" in fruits)
11
12 # fruits.add("pineapple")
13 # fruits.remove("apple")
```

```
4     # Tuple = () ordered and unchangeable. Duplicates OK. FASTER
5
6     fruits = ("apple", "orange", "banana", "coconut", "coconut")
7     # print(dir(fruits))
8     # print(help(fruits))
9     # print(len(fruits))
10    # print("pineapple" in fruits)
11
12    # print(fruits.index("apple"))
13    print(fruits.count("coconut"))
14
15    print(fruits)
16    #for fruit in fruits:
17        #print(fruit)
```