keyword union - use.

Day-10 Set

It is used to store multiple item in single variables.

* Set items are unchangeable, but you can add and remove the items.

* Set are in curly brackets.
* Set are unordered, the result will get appear.
* set don't allow the duplicate values.

Length and Datatype

thisset = {'table', 'car', 'laptop', 'pen'}

print (len(thisset)) → 4

print (type(thisset)) → set.

Constructor

thisset = set(('table', 'car', Lap, 'pen'))

print (thisset) → {'table', 'car', 'Lap', 'pen'}

we use tuple bracket () but it's set.

print (type(thisset))
↳ <class 'set'>,

duplicates.

this set = {'car', 'laptop', 'water', 'car'}

print (this set) → {'car', 'laptop', 'water'}.
It can't get the duplicate value.

print (len(thisset)) → 3
↳ Can't count the duplicate values.

Set are unordered and No duplicate Number.

using Loop Values :

```
Thisset = {'car', 'Laptop', 'water', 'pen'}
for x in thisset:
    if x == 'pen':
        Print ('The value is true')
    else:
        Print ('The Value is flase')
```

↳ output

The value is flase
The value is flase
The value is flase

⟹ The value is true

cheak the value

```
Thisset {'car', 'Laptop', 'water', 'pen'}

Print ('Laptop' in thisset)
```

↳ True.

* we cannot change the item but will add and remove

add the number

```
set = {'pen', 'paper', 'pencil'}
set.add ('note')
print (set) → {'pen', 'paper', 'Pencil', 'Note', 'pen'}.
```

⊛ set has not attribute "append".

※. In set all the result are ~~change~~ unorderable will be changed all times.

update the value

```
set1 = {'pen', 'paper', 'Notes'}
set2 = {'school', 'tution', 'examination'}
    set1.update(set2)
    print(set1) → {'pen', 'paper', 'Notes', 'school', 'tution',
                   'examination'}
```

add update List value

```
set1 = {'pen', 'paper', 'Notes'}
set  = {'today', 'yesterday'}
  set1.update(set)
   print(set1)
        ↳ {'~~pen~~', 'paper', 'Notes', 'today', '~~pen~~', 'pen',
                                                'yesterday'}
```

∴ here don't use 'add' key word.

remove values

```
set1.remove('yesterday')
    print(set1)
        ↳ {'Notes', 'today', 'pen', 'paper'}
```

discard means remove

using discard.

```
set1.discard('today')
print(set1)
        ↳ {'Notes', 'pen', 'paper'}
```

pop keyword.

↳ Can use values we only Index values.

```
set1.pop()
print(set1) → {'Notes', 'pen'}
```

∴ pop() → here ~~use~~ can't mention anything means last character will be removed.

for Loop.

```
set = {'blue', 'black', 'red'}
for x in set:
    print(x)
```

↳ blue
~~black~~.
red.
black.

using union

```
set = {'blue', 'black', 'red'}
set1 = {1, 2, 3}

set3 = set.union(set1).
print(set3)
```

↳ {'blue', 2, 'red', 3, 1, 'black'}

Intersection

In sets, the items are contains that are present in both sets.

```
x = {'car', 'bike', 'cycle', 'bus'}
y = {'bike', 'lorry', 'aeroplane'}.

z = x.intersection(y)
print(z).
```

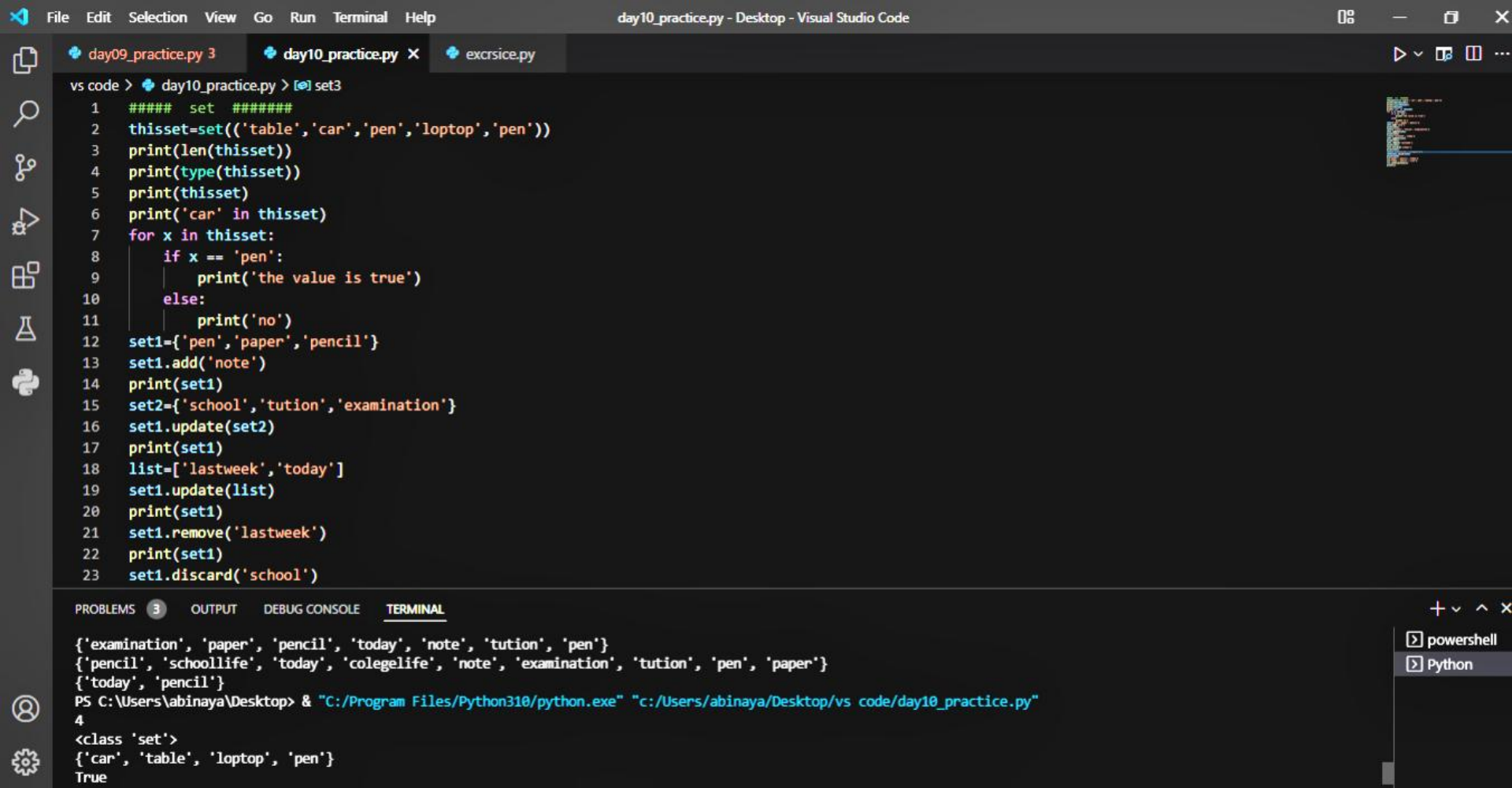
↳ {'bike'}

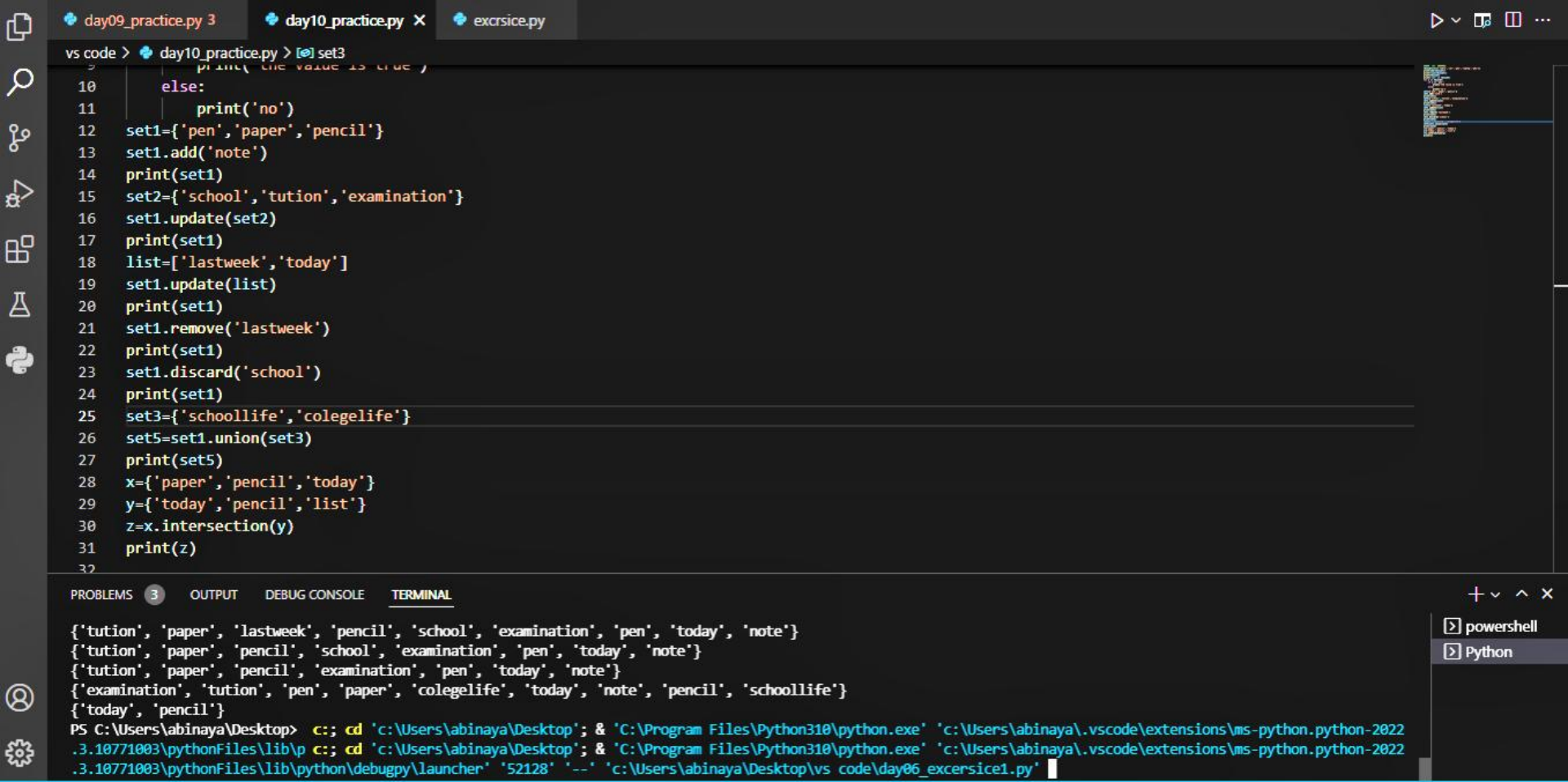
→ must there two value are same in sets.

Symmetric-difference()

```
z = x.symmetric_diffrence(y)
print(z).
```

↳ {'bike'} →


File Edit Selection View Go Run Terminal Help
day10_practice.py - Desktop - Visual Studio Code
day09_practice.py 3
day10_practice.py
excrsice.py
vs code > day10_practice.py > set3
set
thisset=set(('table','car','pen','loptop','pen'))
print(len(thisset))
print(type(thisset))
print(thisset)
print('car' in thisset)
for x in thisset:
if x == 'pen':
print('the value is true')
else:
print('no')
set1={'pen','paper','pencil'}
set1.add('note')
print(set1)
set2={'school','tution','examination'}
set1.update(set2)
print(set1)
list=['lastweek','today']
set1.update(list)
print(set1)
set1.remove('lastweek')
print(set1)
set1.discard('school')
PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
{'examination', 'paper', 'pencil', 'today', 'note', 'tution', 'pen'}
{'pencil', 'schoollife', 'today', 'colegelife', 'note', 'examination', 'tution', 'pen', 'paper'}
{'today', 'pencil'}
PS C:\Users\abinaya\Desktop> & "C:/Program Files/Python310/python.exe" "c:/Users/abinaya/Desktop/vs code/day10_practice.py"
4
<class 'set'>
{'car', 'table', 'loptop', 'pen'}
True
powershell
Python


day09_practice.py 3
day10_practice.py
excrsice.py
vs code > day10_practice.py > set3
print('the value is true')
else:
print('no')
set1={'pen','paper','pencil'}
set1.add('note')
print(set1)
set2={'school','tution','examination'}
set1.update(set2)
print(set1)
list=['lastweek','today']
set1.update(list)
print(set1)
set1.remove('lastweek')
print(set1)
set1.discard('school')
print(set1)
set3={'schoollife','colegelife'}
set5=set1.union(set3)
print(set5)
x={'paper','pencil','today'}
y={'today','pencil','list'}
z=x.intersection(y)
print(z)
PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
{'tution', 'paper', 'lastweek', 'pencil', 'school', 'examination', 'pen', 'today', 'note'}
{'tution', 'paper', 'pencil', 'school', 'examination', 'pen', 'today', 'note'}
{'tution', 'paper', 'pencil', 'examination', 'pen', 'today', 'note'}
{'examination', 'tution', 'pen', 'paper', 'colegelife', 'today', 'note', 'pencil', 'schoollife'}
{'today', 'pencil'}
PS C:\Users\abinaya\Desktop> c:; cd 'c:\Users\abinaya\Desktop'; & 'C:\Program Files\Python310\python.exe' 'c:\Users\abinaya\.vscode\extensions\ms-python.python-2022
.3.10771003\pythonFiles\lib\p c:; cd 'c:\Users\abinaya\Desktop'; & 'C:\Program Files\Python310\python.exe' 'c:\Users\abinaya\.vscode\extensions\ms-python.python-2022
.3.10771003\pythonFiles\lib\python\debugpy\launcher' '52128' '--' 'c:\Users\abinaya\Desktop\vs code\day06_excersice1.py'
powershell
Python