

PRACTICE

COMPETE

**JOBS** 

**LEADERBOARD** 

Q Search





Practice > Python > Sets > Set Mutations

# **Set Mutations ☆**



Rank: **3480** | Points: **1535** (1)



# Your Set Mutations submission got 10.00 points. Share Tweet Try the next challenge | Try a Random Challenge

Problem	Submissions	Leaderboard	Discussions	Editorial 🖰
We have seen the a	pplications of <i>union, int</i>	tersection, difference ar	nd <i>symmetric difference</i>	
but these operatior	ns do not make any char	nges or mutations to the	e set.	Di
We can use the fol	llowing operations to o	create mutations to a	set:	M. Su
.update() or  =				30
Update the set by a	dding elements from ar	n iterable/another set.		N
>>> H = set('	'Hacker")			Į.
>>> R = set('				Д
<pre>&gt;&gt;&gt; H.update( &gt;&gt;&gt; print H</pre>	(R)			Ç
· ·	, 'e', 'H', 'k', '	n', 'r', 'R'])		R.A
.intersection_upda	ate() or &=			ī
Update the set by keeping only the elements found in it and an iterable/another set.				М
>>> H = set('	'Hacker")			_↓
>>> R = set('				<u>J</u>
	ection_update(R)			
>>> print H set(['a', 'k'	])			<u>0</u>
.difference_update	<b>e()</b> or -=			'
Update the set by re	emoving elements foun	d in an iterable/anothe	r set.	
>>> H = set('	'Hacker")			
>>> R = set('	•			
<pre>&gt;&gt;&gt; H.differe &gt;&gt;&gt; print H</pre>	ence_update(R)			
set(['c', 'e'	', 'H', 'r'])			
.symmetric_differe	ence_update() or ^=			
Update the set by o	only keeping the elemen	ts found in either set, b	ut not in both.	
>>> H = set('	'Hacker")			
>>> R = set('	'Rank")			
>>> H.symmetr	ric_difference_upda	ate(R)		

Author	DOSH
Difficulty	Easy
Max Score	10
Submitted By	18354
NEED HELP?	

- View discussions
- View editorial
- View top submissions

RATE THIS CHALLENGE



# MORE DETAILS

- Download problem statement
- **↓** Download sample test cases
- Suggest Edits



```
>>> print H
set(['c', 'e', 'H', 'n', 'r', 'R'])
```

#### TASK

You are given a set  $m{A}$  and  $m{N}$  number of other sets. These  $m{N}$  number of sets have to perform some specific mutation operations on set  $m{A}$ .

Your task is to execute those operations and print the sum of elements from set  $m{A}$ .

#### **Input Format**

The first line contains the number of elements in set A.

The second line contains the space separated list of elements in set  $m{A}$ .

The third line contains integer N, the number of other sets.

The next 2 \* N lines are divided into N parts containing two lines each.

The first line of each part contains the space separated entries of the *operation name* and the *length of* the other set.

The second line of each part contains space separated list of elements in the other set.

0 < len(set(A)) < 1000

0 < len(otherSets) < 100

0 < N < 100

#### **Output Format**

Output the sum of elements in set A.

### Sample Input

```
16
1 2 3 4 5 6 7 8 9 10 11 12 13 14 24 52
4
intersection_update 10
2 3 5 6 8 9 1 4 7 11
update 2
55 66
symmetric_difference_update 5
22 7 35 62 58
difference_update 7
11 22 35 55 58 62 66
```

#### Sample Output

38

## Explanation

After the first operation, (intersection\_update operation), we get:

set 
$$A = set([1, 2, 3, 4, 5, 6, 7, 8, 9, 11])$$

After the second operation, (*update operation*), we get:

set 
$$A = set([1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 55, 66])$$

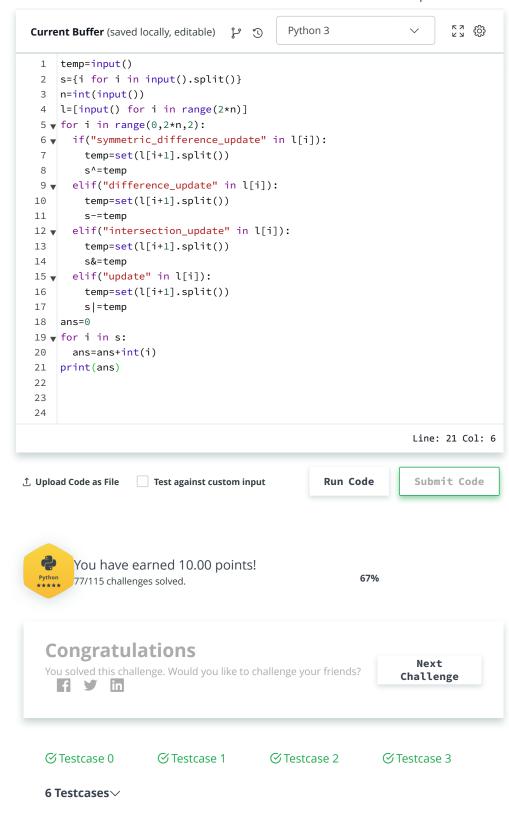
After the third operation, (symmetric\_difference\_update operation), we get:

set 
$$A = set([1, 2, 3, 4, 5, 6, 8, 9, 11, 22, 35, 55, 58, 62, 66])$$

After the fourth operation, ( difference\_update operation), we get:

set 
$$A = set([1, 2, 3, 4, 5, 6, 8, 9])$$

The sum of elements in set  $m{A}$  after these operations is  $m{38}$ .





Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature