1. Python | Ways to remove a key from dictionary
2. Ways to sort list of dictionaries by values in Python – Using itemgetter
3. Ways to sort list of dictionaries by values in Python – Using lambda function
4. Python | Merging two Dictionaries
5. Python – Convert key-values list to flat dictionary
6. Python – Insertion at the beginning in OrderedDict
7. Python | Check order of character in string using OrderedDict( )
8. Dictionary and counter in Python to find winner of election
9. Python – Append Dictionary Keys and Values ( In order ) in dictionary
10. Python | Sort Python Dictionaries by Key or Value

Solve

Here are the solutions to the Python tasks you've listed:

1. \*\*Python | Ways to remove a key from dictionary\*\*:

- \*\*Using `del` keyword\*\*:

```python

my\_dict = {'a': 1, 'b': 2, 'c': 3}

del my\_dict['b']

print(my\_dict) # Output: {'a': 1, 'c': 3}

```

- \*\*Using `pop()` method\*\*:

```python

my\_dict = {'a': 1, 'b': 2, 'c': 3}

my\_dict.pop('b')

print(my\_dict) # Output: {'a': 1, 'c': 3}

```

- \*\*Using `dict comprehension`\*\* (to create a new dictionary):

```python

my\_dict = {'a': 1, 'b': 2, 'c': 3}

new\_dict = {k: v for k, v in my\_dict.items() if k != 'b'}

print(new\_dict) # Output: {'a': 1, 'c': 3}

```

2. \*\*Ways to sort list of dictionaries by values in Python – Using `itemgetter`\*\*:

```python

from operator import itemgetter

my\_list = [{'name': 'John', 'age': 29}, {'name': 'Jane', 'age': 25}, {'name': 'Dave', 'age': 30}]

sorted\_list = sorted(my\_list, key=itemgetter('age'))

print(sorted\_list)

# Output: [{'name': 'Jane', 'age': 25}, {'name': 'John', 'age': 29}, {'name': 'Dave', 'age': 30}]

```

3. \*\*Ways to sort list of dictionaries by values in Python – Using `lambda` function\*\*:

```python

my\_list = [{'name': 'John', 'age': 29}, {'name': 'Jane', 'age': 25}, {'name': 'Dave', 'age': 30}]

sorted\_list = sorted(my\_list, key=lambda x: x['age'])

print(sorted\_list)

# Output: [{'name': 'Jane', 'age': 25}, {'name': 'John', 'age': 29}, {'name': 'Dave', 'age': 30}]

```

4. \*\*Python | Merging two Dictionaries\*\*:

- \*\*Using `update()` method\*\*:

```python

dict1 = {'a': 1, 'b': 2}

dict2 = {'b': 3, 'c': 4}

dict1.update(dict2)

print(dict1) # Output: {'a': 1, 'b': 3, 'c': 4}

```

- \*\*Using dictionary unpacking\*\*:

```python

dict1 = {'a': 1, 'b': 2}

dict2 = {'b': 3, 'c': 4}

merged\_dict = {\*\*dict1, \*\*dict2}

print(merged\_dict) # Output: {'a': 1, 'b': 3, 'c': 4}

```

5. \*\*Python – Convert key-values list to flat dictionary\*\*:

```python

from itertools import zip\_longest

test\_list = [('a', 1), ('b', 2), ('c', 3)]

flat\_dict = dict(zip\_longest(\*test\_list))

print(flat\_dict)

# Output: {'a': 1, 'b': 2, 'c': 3}

```

6. \*\*Python – Insertion at the beginning in `OrderedDict`\*\*:

```python

from collections import OrderedDict

my\_dict = OrderedDict([('b', 2), ('c', 3)])

my\_dict.update({'a': 1})

my\_dict.move\_to\_end('a', last=False)

print(my\_dict)

# Output: OrderedDict([('a', 1), ('b', 2), ('c', 3)])

```

7. \*\*Python | Check order of character in string using `OrderedDict()`\*\*:

```python

from collections import OrderedDict

def check\_order(input\_string, pattern):

order\_dict = OrderedDict.fromkeys(input\_string)

pattern\_index = 0

for key in order\_dict:

if key == pattern[pattern\_index]:

pattern\_index += 1

if pattern\_index == len(pattern):

return True

return False

input\_string = "hello world"

pattern = "hlo"

print(check\_order(input\_string, pattern)) # Output: True

```

8. \*\*Dictionary and counter in Python to find winner of election\*\*:

```python

from collections import Counter

votes = ['Alice', 'Bob', 'Alice', 'Charlie', 'Bob', 'Alice']

vote\_count = Counter(votes)

winner = max(vote\_count, key=vote\_count.get)

print(winner) # Output: 'Alice'

```

9. \*\*Python – Append Dictionary Keys and Values (In order) in dictionary\*\*:

```python

test\_dict = {'gfg': 1, 'is': 2, 'best': 3}

res\_keys = []

res\_vals = []

for key, value in test\_dict.items():

res\_keys.append(key)

res\_vals.append(value)

print("The ordered keys list is : " + str(res\_keys))

print("The ordered values list is : " + str(res\_vals))

# Output: The ordered keys list is : ['gfg', 'is', 'best']

# Output: The ordered values list is : [1, 2, 3]

```

10. \*\*Python | Sort Python Dictionaries by Key or Value\*\*:

- \*\*Sort by Key\*\*:

```python

my\_dict = {'b': 3, 'a': 1, 'c': 2}

sorted\_by\_key = dict(sorted(my\_dict.items()))

print(sorted\_by\_key) # Output: {'a': 1, 'b': 3, 'c': 2}

```

- \*\*Sort by Value\*\*:

```python

my\_dict = {'b': 3, 'a': 1, 'c': 2}

sorted\_by\_value = dict(sorted(my\_dict.items(), key=lambda item: item[1]))

print(sorted\_by\_value) # Output: {'a': 1, 'c': 2, 'b': 3}

```

Let me know if you need any further explanations or examples!