```
#importing neccesary library
import pandas as pd

#reading the dataset
data = pd.read_csv("/content/drive/MyDrive/extreme_changes_post.csv")
data.head(10)
```

	bvdid	numberofemployees	date	title	pro	cons
0	GB00138691	284	11/20/2013	Learned a lot on the job, more than any other ...	Small company feel, good location, learn a lot...	hours get rough and expectations are high
1	GB00138691	284	5/27/2014	Human Resources	Company afforded many opportunities to learn a...	limited opportunities to move up unless someon...

Title column -most common words

```
#writing a code to run through the dataset TITLE column to rank most common words

#initializing empty list to store the split words from the dataframe
words_list = []
new_list=[]
final_list=[]
most_common_words1 = []
dictionary_value_count = {}

#iterating through the Title column and replacing all any words split by . and  / with '',
for word in data['title']:
    words_list.append(word.replace('.', ''))
    words_list.append(word.replace('/', ''))
    words_list.append(word.replace('-', ''))

#iterating through the word_list and converting all words to a string
for val in words_list:
  output1=str(val)
  new_list.append(output1)

#iterating through the new_list and splitting all the words and storing them in a new list
for n_val in new_list:
  output2=n_val.split()
  for res in output2:
    final_list.append(res)

#iterating through final_list and storing all split words into a dictionary
for word in final_list:
    if word not in dictionary_value_count:
        dictionary_value_count[word] = 1
    else:
        num = dictionary_value_count[word]
        dictionary_value_count[word] = (num + 1)

#function for obtaining the largest word count and aggreating them all
def largest_value(value):
    res = 0
    new_str = ''
```

```
    for key, val in value.items():
        if val > res:
            res = val
            new_str = key
    return [res, new_str]

#function for grouping all common words and ranking them together
def common_words( value,target):
    while len(most_common_words1) < target:
        most_common_words1.append(largest_value(value))
        del value[(largest_value(value)[1])]

common_words( dictionary_value_count,935)

print(most_common_words1)
```

```
[[207, 'Great'], [207, 'to'], [168, 'Good'], [165, 'work'], [144, 'company'], [126,
```

PRO column -most common words

```
#writing a code to run through the dataset pro column to rank most common words

#initializing empty list to store the split words from the dataframe
words_list = []
new_list=[]
final_list=[]
most_common_words2 = []
dictionary_value_count = {}

#iterating through the Title column and replacing all any words split by . and  / with '',
for word in data['pro']:
    words_list.append(word.replace('.', ''))
    words_list.append(word.replace('/', ''))
    words_list.append(word.replace('-', ''))

#iterating through the word_list and converting all words to a string
for val in words_list:
  output1=str(val)
  new_list.append(output1)

#iterating through the new_list and splitting all the words and storing them in a new list
for n_val in new_list:
  output2=n_val.split()
  for res in output2:
    final_list.append(res)

#iterating through final_list and storing all split words into a dictionary
for word in final_list:
    if word not in dictionary_value_count:
        dictionary_value_count[word] = 1
    else:
        num = dictionary_value_count[word]
```

```
        dictionary_value_count[word] = (num + 1)

#function for obtaining the largest word count and aggreating them all
def largest_value(value):
    res = 0
    new_str = ''
    for key, val in value.items():
        if val > res:
            res = val
            new_str = key
    return [res, new_str]

#function for grouping all common words and ranking them together
def common_words( value,target):
    while len(most_common_words2) < target:
        most_common_words2.append(largest_value(value))
        del value[(largest_value(value)[1])]

common_words( dictionary_value_count,935)

print(most_common_words2)
```

```
[[1113, 'and'], [814, 'to'], [660, 'the'], [462, 'a'], [384, 'of'], [381, 'work'], [
```

CONS column -most common words

```
#writing a code to run through the dataset cons column to rank most common words

#initializing empty list to store the split words from the dataframe
words_list = []
new_list=[]
final_list=[]
most_common_words3 = []
dictionary_value_count = {}

#iterating through the Title column and replacing all any words split by . and  / with '',
for word in data['cons']:
    words_list.append(word.replace('.', ''))
    words_list.append(word.replace('/', ''))
    words_list.append(word.replace('-', ''))


#iterating through the word_list and converting all words to a string
for val in words_list:
  output1=str(val)
  new_list.append(output1)

#iterating through the new_list and splitting all the words and storing them in a new list
for n_val in new_list:
  output2=n_val.split()
  for res in output2:
    final_list.append(res)
```

```
#iterating through final_list and storing all split words into a dictionary
for word in final_list:
    if word not in dictionary_value_count:
        dictionary_value_count[word] = 1
    else:
        num = dictionary_value_count[word]
        dictionary_value_count[word] = (num + 1)

#function for obtaining the largest word count and aggreating them all
def largest_value(value):
    res = 0
    new_str = ''
    for key, val in value.items():
        if val > res:
            res = val
            new_str = key
    return [res, new_str]

#function for grouping all common words and ranking them together
def common_words( value,target):
    while len(most_common_words3) < target:
        most_common_words3.append(largest_value(value))
        del value[(largest_value(value)[1])]

common_words( dictionary_value_count,935)

print(most_common_words3)
```

```
[[1413, 'the'], [1316, 'to'], [1167, 'and'], [744, 'a'], [726, 'of'], [720, 'is'], [
```

```
import pandas
df = pandas.DataFrame(data={"Title_col_most_common_words": most_common_words1, "PRO_col_mo
                            "CONS_col_most_common_words": most_common_words3})
df.to_csv("./most_common_words.csv", sep=',',index=False)
```

```
most_common_words_file=pd.read_csv('/content/most_common_words.csv')
most_common_words_file
```

	Title_col_most_common_words	PRO_col_most_common_words	CONS_col_most_common_w
0	[207, 'Great']	[1113, 'and']	[1413,
1	[207, 'to']	[814, 'to']	[1316
2	[168, 'Good']	[660, 'the']	[1167, '
3	[165, 'work']	[462, 'a']	[74
4	[144, 'company']	[384, 'of']	[726

THE END OF IMPLEMENTATION.THANK YOU!!!

932	[1, 'Nuclearopportunities']	[3, 'Corporate']	[6, 'appro
933	[1, 'Apprentice']	[3, 'Customers']	[6, 'exte
934	[1, 'Slipped']	[3, 'apart']	[6, 'igr

935 rows × 3 columns

✓ 0s completed at 2:06 AM