```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import nltk
nltk.download('stopwords')

from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
from nltk.stem.porter import PorterStemmer

import string
import re
import textblob
from textblob import TextBlob
import os

from wordcloud import WordCloud, STOPWORDS
from wordcloud import ImageColorGenerator
import warnings
%matplotlib inline
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
#Read the JSON generated from the CLI command above and create a pandas dataframe
df = pd.read_excel(r'/content/HR Employee Survey Responses.xlsx')
```

```
df.head(5)
```

	Response ID	Status	Department	Director	Manager	Supervisor	Staff	Question	Response	Resp
0	1	Complete	Human Resources	0	1	0	0	1. I know what is expected of me at work	4.0	Str A
1	2	Complete	Communications Office	0	0	0	0	1. I know what is expected of me at work	4.0	Str A
2	3	Complete	Parks and Recreation	0	1	0	0	1. I know what is expected of me at work	0.0	Appli
3	4	Complete	Human Resources	0	1	0	0	1. I know what is expected of me at work	3.0	A
4	5	Complete	Communications Office	0	0	0	0	1. I know what is expected of me at work	0.0	Appli

```
df.to_csv()
```

```
',Response ID,Status,Department,Director,Manager,Supervisor,Staff,Question,Response,Response Te
xt\n0,1,Complete,Human Resources,0,1,0,0,1. I know what is expected of me at work,4.0,Strongly
Agree\n1,2,Complete,Communications Office,0,0,0,0,1. I know what is expected of me at work,4.0,
Strongly Agree\n2,3,Complete,Parks and Recreation,0,1,0,0,1. I know what is expected of me at w
ork,0.0,Not Applicable\n3,4,Complete,Human Resources,0,1,0,0,1. I know what is expected of me a
t work,3.0,Agree\n4,5,Complete,Communications Office,0,0,0,0,1. I know what is expected of me a
t work,0.0,Not Applicable\n5,6,Complete,Prosecuting Attorney\'s Office,0,0,0,0,1. I know what i
s expected of me at work,4.0,Strongly Agree\n6,7,Complete,Prosecuting Attorney\'s Office,0,0,0,
0,1. I know what is expected of me at work,4.0,Strongly Agree\n7,8,Complete,Finance and Perform
ance Management,0,0,0,1,1. I know what is expected of me at work,4.0,Strongly Agree\n8,9,Comple
te Finance and Performance Management 0 0 0 0 1 I
```

```
df.shape
```

```
(14725, 10)
```

```
df.head()
```

	Response ID	Status	Department	Director	Manager	Supervisor	Staff	Question	Response	Respons Tex
	1	Complete	Human Resources	0	1	0	0	1. I know what is expected of me at work	4.0	Strong Agr
	2	Complete	Communications Office	0	0	0	0	1. I know what is expected of me at work	4.0	Strong Agr
	3	Complete	Parks and Recreation	0	1	0	0	1. I know what is expected of me at work	0.0	N Applicab
	4	Complete	Human Resources	0	1	0	0	1. I know what is expected of me at work	3.0	Agr
	5	Complete	Communications Office	0	0	0	0	1. I know what is expected of me at work	0.0	N Applicab

```
df.info()
```

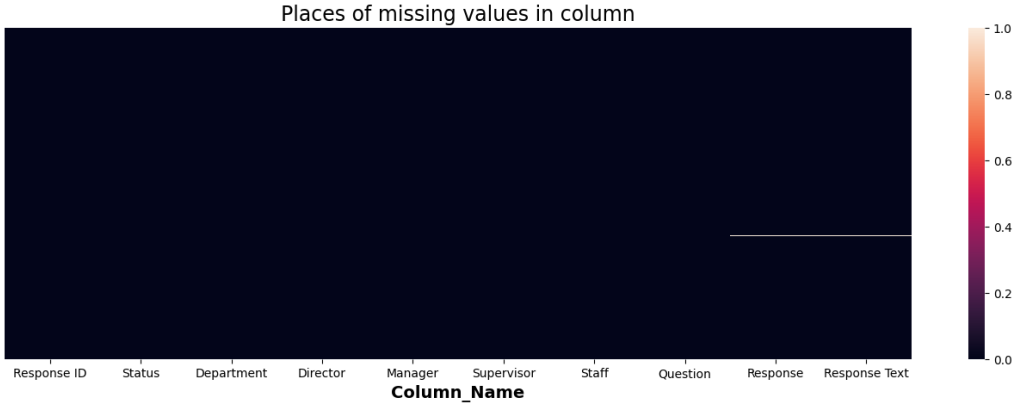
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14725 entries, 0 to 14724
Data columns (total 10 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Response ID    14725 non-null  int64
 1   Status         14725 non-null  object
 2   Department     14725 non-null  object
 3   Director       14725 non-null  int64
 4   Manager        14725 non-null  int64
 5   Supervisor     14725 non-null  int64
 6   Staff          14725 non-null  int64
 7   Question       14725 non-null  object
 8   Response       14590 non-null  float64
 9   Response Text  14590 non-null  object
```

```
dtypes: float64(1), int64(5), object(4)
memory usage: 1.1+ MB
```

```
df.value_counts()
```

```
Response ID  Status    Department                          Director  Manager  Supervisor  Staff
Question                                                                      Response  Response
Text
11399        Complete  Communications Office               0         0        0           0
8. My supervisor holds employees accountable for performance                  4.0       Strongly
Agree    2
14076        Complete  Facilities Management               0         0        0           0
10. Overall I am satisfied with my job                                        3.0       Agree
2
14091        Complete  Finance and Performance Management  0         1        0           0
10. Overall I am satisfied with my job                                        4.0       Strongly
Agree    2
11629        Complete  Parks and Recreation                0         0        0           0
8. My supervisor holds employees accountable for performance                  2.0       Disagree
2
10963        Complete  Finance and Performance Management  0         0        1           0
8. My supervisor holds employees accountable for performance                  3.0       Agree
2

..
4908         Complete  Parks and Recreation                0         0        0           0
4. My supervisor, or someone at work, seems to care about me as a person      0.0       Not
Applicable    1
4909         Complete  Parks and Recreation                0         0        1           0
4. My supervisor, or someone at work, seems to care about me as a person      4.0       Strongly
Agree    1
4910         Complete  Planning and Public Works           0         0        0           0
4. My supervisor, or someone at work, seems to care about me as a person      4.0       Strongly
Agree    1
4911         Complete  District Court                      0         0        0           0
4. My supervisor, or someone at work, seems to care about me as a person      4.0       Strongly
Agree    1
14710        Complete  Human Services                      0         0        0           0
10. Overall I am satisfied with my job                                        4.0       Strongly
Agree    1
Length: 14575, dtype: int64
```

```
plt.figure(figsize=(17, 5))
sns.heatmap(df.isnull(), cbar=True, yticklabels=False)
plt.xlabel("Column_Name", size=14, weight="bold")
plt.title("Places of missing values in column",size=17)
plt.show()
```



```
import plotly.graph_objects as go
Top_Questions= df['Question'].value_counts().head(10)
Top_Questions
```

```
8. My supervisor holds employees accountable for performance                          1478
9. My department is inclusive and demonstrates support of a diverse workforce        1473
1. I know what is expected of me at work                                              1472
2. At work, I have the opportunity to do what I do best every day                     1471
3. In the last seven days, I have received recognition or praise for doing good work  1471
4. My supervisor, or someone at work, seems to care about me as a person              1471
5. The mission or purpose of our organization makes me feel my job is important      1471
6. I have a best friend at work                                                       1471
10. Overall I am satisfied with my job                                                1454
7. This last year, I have had opportunities at work to learn and grow                  962
Name: Question, dtype: int64
```

```
from nltk. corpus import stopwords
stop = stopwords.words('english')
df['Department'].apply(lambda x: [item for item in x if item not in stop])
df.shape
```

```
(14725, 10)
```

```
!pip install tweet-preprocessor
```

```
Collecting tweet-preprocessor
  Downloading tweet_preprocessor-0.6.0-py3-none-any.whl (27 kB)
Installing collected packages: tweet-preprocessor
Successfully installed tweet-preprocessor-0.6.0
```

```
punct  =  ['%','/',':','\\','&amp','&',';','?']

def remove_punctuations(text):
  for punctuation in punct:
    text = text.replace(punctuation,'')
  return text


df['Department'] = df['Department'].apply(lambda x: remove_punctuations(x))


#Drop tweets that has empty text fields
df['Department'].replace( '', np.nan, inplace=True)
df.dropna(subset=["Department"],inplace=True)
len(df)
```

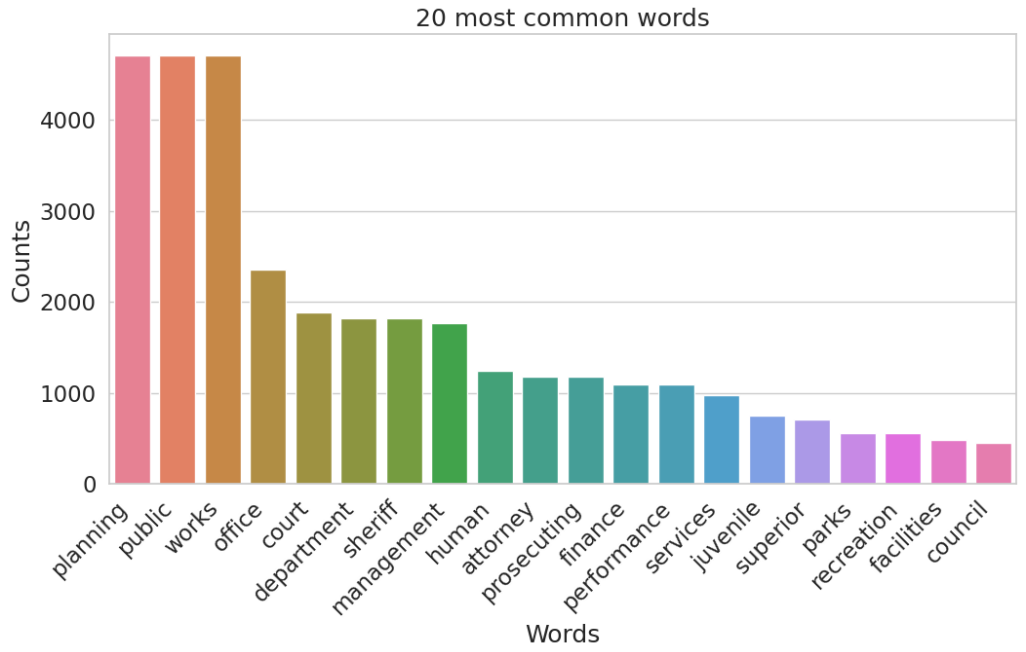
```
14725
```

```
df = df.reset_index(drop=True)
df.head(15)
```

5	6	Complete	Prosecuting Attorney's Office	0	0	0	0
6	7	Complete	Prosecuting Attorney's Office	0	0	0	0
7	8	Complete	Finance and Performance Management	0	0	0	1
8	9	Complete	Finance and Performance Management	0	0	0	0
9	10	Complete	Planning and Public Works	0	0	0	0
10	11	Complete	Planning and Public Works	0	0	0	0
11	12	Complete	Planning and Public Works	0	0	0	0
12	12	Complete	Planning and Public Works	0	0	0	0
13	13	Complete	Human Services	0	0	0	0
14	14	Complete	Human Services	0	0	0	1

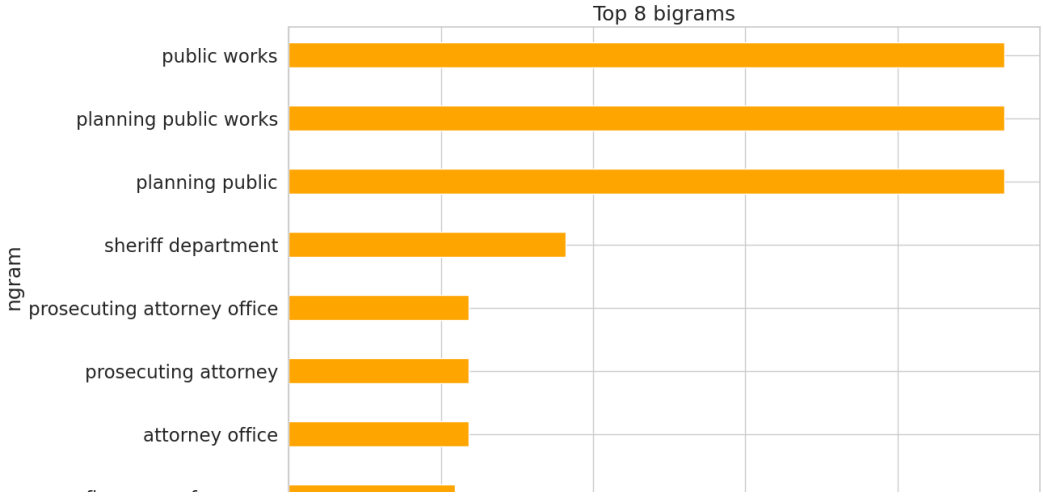
```
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer

sns.set_style('whitegrid')
%matplotlib inline

stop = stop + ['Human Resources', 'Prosecuting Attorneys Office', 'Planning and Public Works', 'Huma

def plot_20_most_common_words(count_data, count_vectorizer):
    words = count_vectorizer.get_feature_names_out()
    total_counts = np.zeros(len(words))

    for t in count_data:
        total_counts += t.toarray()[0]

    count_dict = dict(zip(words, total_counts))
    count_dict = sorted(count_dict.items(), key=lambda x: x[1], reverse=True)[:20]

    words = [w[0] for w in count_dict]
    counts = [w[1] for w in count_dict]

    x_pos = np.arange(len(words))

    plt.figure(figsize=(12, 6))
    sns.set_context('notebook', font_scale=1.5)
    sns.barplot(x=x_pos, y=counts, palette='husl')
    plt.title('20 most common words')
    plt.xticks(x_pos, words, rotation=45, ha='right')
    plt.xlabel('Words')
    plt.ylabel('Counts')
    plt.show()


count_vectorizer = CountVectorizer(stop_words=stop)
count_data = count_vectorizer.fit_transform(df['Department'])

# Visualize the 20 most common words
plot_20_most_common_words(count_data, count_vectorizer)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/feature_extraction/text.py:
  warnings.warn(
<ipython-input-39-a5dc7011acca>:28: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removec

  sns.barplot(x=x_pos, y=counts, palette='husl')
```




```
import cufflinks as cf
cf.go_offline()
cf.set_config_file(offline=False, world_readable=True)

def get_top_n_bigram(corpus, n=None) :
  vec = CountVectorizer(ngram_range=(2, 4), stop_words="english").fit(corpus)
  bag_of_words = vec.transform(corpus)
  sum_words = bag_of_words.sum(axis=0)
  words_freq =[(word, sum_words[0, idx]) for word, idx in vec.vocabulary_.items()]
  words_freq =sorted(words_freq, key = lambda x: x[1], reverse=True)
  return words_freq[:n]

common_words = get_top_n_bigram(df['Department'] , 8)
mydict={}
for word, freq in common_words:
  bigram_df = pd.DataFrame(common_words,columns = ['ngram', 'count'])

bigram_df.groupby( 'ngram' ).sum()['count'].sort_values(ascending=False).sort_values().plot.barh(titl
```

```
<Axes: title={'center': 'Top 8 bigrams'}, ylabel='ngram'>
```




```
def get_subjectivity(text):
  return TextBlob(text).sentiment.subjectivity
def get_polarity(text):
  return TextBlob(text).sentiment.polarity


df['subjectivity']=df[ 'Department'].apply(get_subjectivity)
df[ 'polarity' ]=df[ 'Department'].apply(get_polarity)
df.head()
```

	Response ID	Status	Department	Director	Manager	Supervisor	Staff	Qu
0	1	Complete	Human Resources	0	1	0	0	1 e
1	2	Complete	Communications Office	0	0	0	0	1 e
2	3	Complete	Parks and Recreation	0	1	0	0	1 e
3	4	Complete	Human Resources	0	1	0	0	1 e
4	5	Complete	Communications Office	0	0	0	0	1 e