In [1]:
```
import pandas as pd
from collections import OrderedDict
import os
import numpy as np
import math as mt
os.chdir('/Users/dummydata')
comb_df=pd.DataFrame()
for i in range(1,20):
    f_name='Data'+str(i)+'.xlsx'
    df=pd.read_excel(f_name,parse_date=['conversation_created_at'])
    new_df=pd.DataFrame({'BranchID':df.fac_id,'WorkerID':df.user_id,'Tim
e': df.conversation_created_at,
                         'Duration':df.duration})
    for j in range(1,14):
        new_df['Query'+str(j)]=df['page_name-'+str(j)]
        new_df['Response'+str(j)]=df['user_input-'+str(j)]
    comb_df=pd.concat([comb_df,new_df], ignore_index=True)
```

In [2]:
```
writer=pd.ExcelWriter('Combined0.xlsx')
comb_df.to_excel(writer,'Sheet1')
writer.save()
```

In [3]:
```
Qs=[];
for i in range(1,14):
    Qs.append(comb_df['Query'+str(i)].unique())
Q=pd.DataFrame({'Col1': np.concatenate(Qs)}).Col1.unique()
Q
```

Out[3]:
```
array(['In the last month, were all your wages, including overtime hour
s, paid on time?',
       'In the last month, have you experienced abuse from a manager, s
uch as swearing, physical abuse, or sexual harassment? ',
       'Are the fire exits in your factory always accessible at all tim
es? ',
       'In the last month, have you witnessed any child worker in your
factory? ',
       'On a scale of 0 to 4, how would you rate the cleanliness of the
toilet in the last month?',
       nan,
       'Are you forced to work overtime in your factory to avoid non-pa
yment and getting fired?',
       'On a scale of 0 to 4, how would you rate the cleanliness of the
canteen in the last month?',
       'Do you have access to clean drinking water at your factory floo
r?',
       'If you have any other feedback on your factory, press â€œ1â€\x9
d or else press â€œ2â€\x9d.',
       'What type of abuse did you experience? Swearing (S), Physical
(P) or Sexual Harassment (SH)',
       'Congratulations. You have completed the survey successfully. Yo
ur identity is safe with us. Stay connected. Press 1 if you want to hea
r the main menu or you can hang up the call. ',
       'In the last month, have you ever worked more than 10 hours in a
day? ',
       'Will you recommend this factory to a friend or family member?
',
       'Please leave a message on your feedback regarding your factory
after the tone and press the # key when finished.',
       'Are you free to join or form trade unions/worker welfare commit
tees in your factory?',
       'Child labor confirmation'], dtype=object)
```

In [4]:
```
arr_irrv=[5,11,16]#Remove non-question strings in array Q
Q_valid=np.delete(Q,arr_irrv)#Delete non-valid questions but unsorted
Q_Top=[]
for j in range(0,13):
    if len(comb_df['Query'+str(j+1)].unique())==1:
        if mt.isnan(comb_df['Query'+str(j+1)].unique()):
            print('Query '+str(j+1)+' is NaN in all column entries')
            Q_Top[j:]=['NaN']
            continue;
    Q_Top[j:]=comb_df['Query'+str(j+1)].describe().loc[['top']]
Q_list=np.hstack([Q_Top,Q_valid])#combine both lists
Q_list=Q_list[Q_list!='NaN']#np.delete(Q_list=='nan') #delete unnecessar
y rows
Q_uniq=pd.unique(pd.Series(Q_list))#Unique and resorted Questions
Q_uniq
```

Out[4]:
```
array(['In the last month, were all your wages, including overtime hour
s, paid on time?',
       'Are the fire exits in your factory always accessible at all tim
es? ',
       'In the last month, have you experienced abuse from a manager, s
uch as swearing, physical abuse, or sexual harassment? ',
       'On a scale of 0 to 4, how would you rate the cleanliness of the
toilet in the last month?',
       'In the last month, have you witnessed any child worker in your
factory? ',
       'On a scale of 0 to 4, how would you rate the cleanliness of the
canteen in the last month?',
       'If you have any other feedback on your factory, press â€œ1â€\x9
d or else press â€œ2â€\x9d.',
       'In the last month, have you ever worked more than 10 hours in a
day? ',
       'Will you recommend this factory to a friend or family member?
',
       'Are you forced to work overtime in your factory to avoid non-pa
yment and getting fired?',
       'Do you have access to clean drinking water at your factory floo
r?',
       'What type of abuse did you experience? Swearing (S), Physical
(P) or Sexual Harassment (SH)',
       'Please leave a message on your feedback regarding your factory
after the tone and press the # key when finished.',
       'Are you free to join or form trade unions/worker welfare commit
tees in your factory?'],
      dtype=object)
```

In [5]:
```
#If you want to remove specific values from the dataframes, replace them
 with empty string
for i in range(0,13):
    if len(comb_df['Query'+str(i+1)].unique())==1:
        if mt.isnan(comb_df['Query'+str(i+1)].unique()):
            print('Query '+str(i+1)+' is NaN in all column entries')
            Q_Top[j:]=['NaN']
            continue;
    Irrv_pos0=comb_df[comb_df['Query'+str(i+1)]==Q[arr_irrv[0]]].index
    Irrv_pos1=comb_df[comb_df['Query'+str(i+1)]==Q[arr_irrv[1]]].index
    Irrv_pos2=comb_df[comb_df['Query'+str(i+1)]==Q[arr_irrv[2]]].index
    print(i)
    if len(Irrv_pos0)>0:
        comb_df.iloc[Irrv_pos0,5+2*i]=''
    if len(Irrv_pos1)>0:
        comb_df.iloc[Irrv_pos1,5+2*i]=''
    if len(Irrv_pos2)>0:
        comb_df.iloc[Irrv_pos2,5+2*i]=''
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
```

In [6]:
```
comb_df=comb_df.replace(Q[arr_irrv[0]],'')
comb_df=comb_df.replace(Q[arr_irrv[1]],'')
comb_df=comb_df.replace(Q[arr_irrv[2]],'')
```

In [7]:
```
writer=pd.ExcelWriter('Combined1.xlsx')
comb_df.to_excel(writer,'Sheet1')
writer.save()
```

In [8]:
```
c=len(comb_df.iloc[0])#Total number of columns
r=len(comb_df)#Total number of rows
df_Qsort=pd.DataFrame()
df_Qsort=pd.concat([df_Qsort,comb_df])
X=[];
print(c,r)
for k in range(0,r):
    ii=0
#     print(k,end=",",flush=True)
    for i in range(4,c,2):
        for j in range(4,c,2):
            if comb_df.iloc[k,j]==Q_uniq[ii]:
                    df_Qsort.iloc[k,i]=comb_df.iloc[k,j]
                    df_Qsort.iloc[k,i+1]=comb_df.iloc[k,j+1]
                    if i!=j:
                        df_Qsort.iloc[k,j]=""
                        df_Qsort.iloc[k,j+1]=""
        ii=ii+1
    X.append(sum(df_Qsort.iloc[k,:]==''))
df_Qsort['Null_values']=pd.DataFrame({'Null_values':X}).values
Sorted_df=df_Qsort.sort_values('Null_values',ascending=True)#df_Qsort
```

```
30 3853
```

In [9]:
```
del df_Qsort['Null_values']
writer=pd.ExcelWriter('Combined2.xlsx')
df_Qsort.to_excel(writer,'Sheet1')
writer.save()
```

In [10]:
```
del Sorted_df['Null_values']
writer=pd.ExcelWriter('Combined3.xlsx')
Sorted_df.to_excel(writer,'Sheet1')
writer.save()
```

In [11]:
```
#Deleting unnecessary rows
```

In [12]:
```
Y=np.sort(X)
len(Sorted_df),len(Sorted_df.iloc[0]),len(Y),len(Y[Y<22])
```

Out[12]:
```
(3853, 30, 3853, 3624)
```

In [13]:
```
Sorted_df.head(2)
```

Out[13]:

	BranchID	Duration	Time	WorkerID	Query1	Response1	Query2	Response2
0	o1	108.692997	2016-07-15 07:08:00 UTC	17893	In the last month, were all your wages, includ...	1	Are the fire exits in your factory always acce...	1
1891	o8	108.354810	2016-07-20 12:04:52 UTC	10542	In the last month, were all your wages, includ...	1	Are the fire exits in your factory always acce...	1

2 rows × 30 columns

In [14]:
```
# Now we can see first four columns are reserved while next are Queries
 and their responses.
# We want to delete all those rows that have 1 or 2 responses. This mean
s Y<10.
# Since we have already arranged the data set, we can delete the require
d rows easily
comb_df2=Sorted_df.drop(Sorted_df.index[Y>21])
len(comb_df2)
```

Out[14]: 3624

In [15]:
```
writer=pd.ExcelWriter('Combined4.xlsx')
comb_df2.to_excel(writer,'Sheet1')
writer.save()
```