```
In [2]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
        %matplotlib inline
```

```
In [3]: xls = pd.ExcelFile('Technical 1.xlsx')
        df1 = pd.read_excel(xls, 'INVOICE DETAILS')
        df2 = pd.read_excel(xls, 'PAYMENT DETAILS')
```

```
In [4]: df3=df1.groupby(['Company','INV.NUMBER']).sum()
        df3.head(10)
```

Out[4]:

		INV.AMT
Company	INV.NUMBER	
Company A	1070172	465.15
	1071948	1715.80
	1074640	3045.54
	1075568	444.34
	1076452	100.84
	1079268	3887.60
	1081130	1026.00
	1082049	88.00
	1082050	354.00
	1082936	1070.20

```
In [5]: df4=df2.groupby(['Company','INV.NUMBER']).sum()
        df4.head(10)
```

Out[5]:

Company	INV.NUMBER	AMT.PD
Company A	1068733	-2800.90
	1070172	-465.15
	1071948	-1715.80
	1074640	-3045.54
	1075568	-444.34
	1076452	-100.84
	1079268	-3887.60
	1081130	-1026.00
	1082049	-88.00
	1082050	-354.00

```
In [6]: df5=pd.merge(df3, df4, how='outer', on=['Company','INV.NUMBER'])
```

```
In [7]: df5.head()
```

Out[7]:

Company	INV.NUMBER	INV.AMT	AMT.PD
Company A	1070172	465.15	-465.15
	1071948	1715.80	-1715.80
	1074640	3045.54	-3045.54
	1075568	444.34	-444.34
	1076452	100.84	-100.84

```
In [62]: df1['In df2?']=df1['INV.NUMBER'].isin(df2['INV.NUMBER'])
         df=pd.merge(df1, df2, how='outer', on=['INV.NUMBER','INV.NUMBER'])
         print(df[df['In df2?'].isnull()])
         #We can see the AMT.PD exists but INV.AMT does not exist. This is sensless and
         hence we will subtracted from the
         #sum of AMT.PD later on from total payments
```

```
      Company_x Type Of Transaction_x  INV.NUMBER INV.DATE_x Part  INV.AMT  \
1204        NaN                   NaN     1068733        NaT  NaN      NaN
1205        NaN                   NaN     1198514        NaT  NaN      NaN

      In df2?  Company_y Type Of Transaction_y INV.DATE_y    DUE.DATE  \
1204      NaN  Company A               Payment 2019-06-13  2019-03-01
1205      NaN  Company A               Payment 2019-07-26  2019-09-24

     PAYMENT.DATE  AMT.PD Ontime or late
1204   2019-06-27 -2800.9           Late
1205   2019-08-23 -7443.4         Ontime
```

```
In [61]: print(2800.9+7443.4) # Will be subtracted later on
```

```
10244.3
```

```
In [8]: df5['NullOrNot']=df5['AMT.PD'].isna()
        df5.reset_index(inplace=True)
        df5.head()
```

Out[8]:

	Company	INV.NUMBER	INV.AMT	AMT.PD	NullOrNot
0	Company A	1070172	465.15	-465.15	False
1	Company A	1071948	1715.80	-1715.80	False
2	Company A	1074640	3045.54	-3045.54	False
3	Company A	1075568	444.34	-444.34	False
4	Company A	1076452	100.84	-100.84	False

```
In [9]: l=[]
        for i in range(len(df5)):
            if df5['NullOrNot'].iloc[i]==True:
                l.append(df5['INV.NUMBER'].iloc[i])
        len(l)
```

Out[9]: 35

```
In [10]:  uA=0 #Unpaid amount by company A
          uB=0
          uC=0
          p=[]
          for i in range(len(df5)):
              if df5['NullOrNot'].iloc[i]==True and df5['Company'].iloc[i]=="Company A":
                  uA=df5['INV.AMT'].iloc[i]+uA
              elif df5['NullOrNot'].iloc[i]==True and df5['Company'].iloc[i]=="Company
           B":
                  uB=df5['INV.AMT'].iloc[i]+uB
              elif df5['NullOrNot'].iloc[i]==True and df5['Company'].iloc[i]=="Company
           C":
                  uC=df5['INV.AMT'].iloc[i]+uC
```

```
In [11]:  print(uA) #unpaid amt by A
          print(uB) #unpaid amt by B
          print(uC) #unpaid amt by C
          print(uA+uB+uC) #Total unpaid amount
```

```
23876.66
9124.36
12221.9
45222.920000000006
```

```
In [12]:  overpaidA=0
          underpaidA=0
          underpaidB=0
          underpaidC=0
          for i in range(len(df5)):
              if df5['NullOrNot'].iloc[i]==False:
                  if df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i]<0 and df5['Company'].il
          oc[i]=='Company A':
                      overpaidA=overpaidA-(df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i])
                      #p.append(df5['INV.NUMBER'].iloc[i])
                      #print(df5['INV.NUMBER'].iloc[i])
                  elif df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i]>0 and df5['Company'].
          iloc[i]=='Company A':
                      underpaidA=(df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i])+underpaid
          A
                  elif df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i]>0 and df5['Company'].
          iloc[i]=='Company B':
                      underpaidB=(df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i])+underpaid
          B
                  elif df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i]>0 and df5['Company'].
          iloc[i]=='Company C':
                      underpaidC=(df5['INV.AMT'].iloc[i]+df5['AMT.PD'].loc[i])+underpaid
          C
```

```
In [18]:  print(underpaidA) # Underpaid amount paid by A
          print(underpaidB) # Underpaid amount paid by B
          print(underpaidC) # Underpaid amount paid by C
          print(underpaidA+underpaidB+underpaidC) #Total Underpaid amount
```

```
156.8900000000087
212.0000000000001
6942.810000000001
7311.70000000001
```

```
In [14]:  df2['Ontime or late'] = np.where(df2['PAYMENT.DATE'] <= df2['DUE.DATE'], 'Onti
          me', 'Late')
```

```
In [15]:  df2.head()
```

Out[15]:

	Company	Type Of Transaction	INV.NUMBER	INV.DATE	DUE.DATE	PAYMENT.DATE	AMT.PD	Ontime or late
0	Company A	Payment	1070172	2019-01-03	2019-03-04	2019-04-02	-465.15	Late
1	Company B	Payment	1070173	2019-01-03	2019-03-04	2019-03-14	-63.00	Late
2	Company A	Payment	1071948	2019-01-07	2019-03-08	2019-03-21	-1715.80	Late
3	Company A	Payment	1074640	2019-01-10	2019-03-11	2019-03-21	-3045.54	Late
4	Company A	Payment	1075568	2019-01-11	2019-03-12	2019-03-21	-444.34	Late

```python
In [16]: LA=0
         EA=0
         LB=0
         EB=0
         LC=0
         EC=0
         for i in range (len(df2)):
             if df2['Ontime or late'].iloc[i] =='Late' and df2['Company'].iloc[i]=='Com
         pany A':
                 LA=LA -df2['AMT.PD'].iloc[i]
             elif df2['Ontime or late'].iloc[i]=='Ontime'and df2['Company'].iloc[i]=='C
         ompany A':
                 EA=EA-df2['AMT.PD'].iloc[i]

             elif df2['Ontime or late'].iloc[i] =='Late' and df2['Company'].iloc[i]=='C
         ompany B':
                 LB=LB -df2['AMT.PD'].iloc[i]
             elif df2['Ontime or late'].iloc[i]=='Ontime'and df2['Company'].iloc[i]=='C
         ompany B':
                 EB=EB-df2['AMT.PD'].iloc[i]

             elif df2['Ontime or late'].iloc[i] =='Late' and df2['Company'].iloc[i]=='C
         ompany C':
                 LC=LC -df2['AMT.PD'].iloc[i]
             elif df2['Ontime or late'].iloc[i]=='Ontime'and df2['Company'].iloc[i]=='C
         ompany C':
                 EC=EC-df2['AMT.PD'].iloc[i]
```

```python
In [19]: print(LA) #Late payment by A
         print(EA) #Ontime payment by A
         print(LB)
         print(EB)
         print(LC)
         print(EC)
         print(LA+EA)
         print(LB+EB)
         print(LC+EC)
         print(LA+LB+LC+EA+EB+EC) #Total (late or ontime) amount paid
```

```
254529.24000000008
7443.4
148989.44
156.86
197276.39000000004
43578.380000000005
261972.64000000007
149146.3
240854.77000000005
651973.7100000001
```

```
In [35]: Company=['A', 'B', 'C']
         data = {
                 'Ontime':[EA,EB,EC],
                 'Late':[LA,LB,LC],
                 'Unpaid':[uA,uB,uC]
                }
```

```
In [38]: dta= pd.DataFrame(data=data,index=Company)
         dta['Ontime+Late+Unpaid']= dta['Ontime']+dta['Late']+dta['Unpaid']
```

```
In [39]: dta # Condensed result
```

Out[39]:

	Ontime	Late	Unpaid	Ontime+Late+Unpaid
A	7443.40	254529.24	23876.66	285849.30
B	156.86	148989.44	9124.36	158270.66
C	43578.38	197276.39	12221.90	253076.67

```
In [43]: dta['Ontime+Late+Unpaid'].sum()-10244.3 # As mentioned above we will subtract
          this amount
```

Out[43]: 686952.3300000001

```
In [41]: df1['INV.AMT'].sum()
```

Out[41]: 694264.03

```
In [47]: print(694264.03-686952.3300000001) #This difference is created because of unde
         rpayment(when inv amt != payment amt) of
         #(ontime or late) payments
         print(underpaidA+underpaidB+underpaidC) #Total Underpaid amount || Calculated
          above
```

```
7311.699999999953
7311.70000000001
```