SEC Financial Statement Data II: Merge the Files

In [1]:
```
import pandas as pd
import os
```

Select a folder from the "data/sec/downloads" directory:

In [2]:
```
folder = '2020_12'
```

Read the file with the submissions::

In [3]:
```
filings = pd.read_table('data/sec/downloads/'+folder+'/sub.tsv')
filings[:3]
```

Out[3]:

	adsh	cik	name	sic	countryba	stprba	cityba	zipba	bas1
0	0000021510-20-000049	21510	COHERENT INC	3826.0	US	CA	SANTA CLARA	95054	5100 PATRICK HENRY DR
1	0000034088-20-000095	34088	EXXON MOBIL CORP	2911.0	US	TX	IRVING	75039-2298	5959 LAS COLINAS BLVD
2	0000045012-20-000111	45012	HALLIBURTON CO	1389.0	US	TX	HOUSTON	77032	3000 NORTH SAM HOUSTON PARKWAY EAST

3 rows × 40 columns

All columns:

In [4]:
```
filings.columns
```

Out[4]:
```
Index(['adsh', 'cik', 'name', 'sic', 'countryba', 'stprba', 'cityba', 'zipba',
       'bas1', 'bas2', 'baph', 'countryma', 'stprma', 'cityma', 'zipma',
       'mas1', 'mas2', 'countryinc', 'stprinc', 'ein', 'former', 'changed',
       'afs', 'wksi', 'fye', 'form', 'period', 'fy', 'fp', 'filed', 'accepted',
       'prevrpt', 'detail', 'instance', 'nciks', 'aciks', 'pubfloatusd',
       'floatdate', 'floataxis', 'floatmems'],
      dtype='object')
```

Look at these columns:

In [5]:
```
filings[['name','cik','sic','countryinc','filed','form']][:5]
```

Out[5]:

	name	cik	sic	countryinc	filed	form
0	COHERENT INC	21510	3826.0	US	20201201	10-K
1	EXXON MOBIL CORP	34088	2911.0	US	20201201	8-K

	name	cik	sic	countryinc	filed	form
2	HALLIBURTON CO	45012	1389.0	US	20201201	8-K
3	HUMANA INC	49071	6324.0	US	20201201	8-K
4	ALLETE INC	66756	4931.0	US	20201201	8-K

We want only 10-Qs and 10-Ks:

In [6]:
```
filings[(filings.form=='10-Q') | (filings.form=='10-K')]   [['name','cik','sic',
filings[filings.form.isin(['10-Q','10-K'])]                [['name','cik','sic',
```

Out[6]:

	name	cik	sic	countryinc	filed	form
0	COHERENT INC	21510	3826.0	US	20201201	10-K
8	TJX COMPANIES INC /DE/	109198	5651.0	US	20201201	10-Q
74	HENNESSY ADVISORS INC	1145255	6282.0	NaN	20201201	10-K
122	BORROWMONEY.COM, INC.	1656501	7389.0	US	20201201	10-K
150	GSG GROUP INC.	1668523	2750.0	NaN	20201201	10-K

Which rows have no country:

In [7]:
```
filings[filings.countryinc.isnull()]   [['name','cik','sic','countryinc','filed'
```

Out[7]:

	name	cik	sic	countryinc	filed	form
19	PPL CORP	922224	4911.0	NaN	20201201	8-K
27	J2 GLOBAL, INC.	1084048	4822.0	NaN	20201201	8-K
30	ELANCO ANIMAL HEALTH INC	1739104	2834.0	NaN	20201201	8-K

Which rows have a country:

In [8]:
```
filings[filings.countryinc.notnull()]   [['name','cik','sic','countryinc','filed
```

Out[8]:

	name	cik	sic	countryinc	filed	form
0	COHERENT INC	21510	3826.0	US	20201201	10-K
1	EXXON MOBIL CORP	34088	2911.0	US	20201201	8-K
2	HALLIBURTON CO	45012	1389.0	US	20201201	8-K

We also want to make sure that every row has a CIK.
Select all 10-Qs and 10-Ks and make sure every filing has a CIK:

In [9]:
```
filings = filings[filings.form.isin(['10-Q','10-K']) & (filings.cik.notnull())]
```

Read the numbers file:

In [10]:
```
numbers = pd.read_table('data/sec/downloads/'+folder+'/num.tsv', encoding='ISO-8
numbers
```

```
/Users/janschneider/opt/anaconda3/lib/python3.7/site-packages/IPython/core/inter
activeshell.py:3147: DtypeWarning: Columns (12) have mixed types.Specify dtype o
ption on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

Out[10]:

	adsh	tag	version	ddate	qtrs	
0	0001640334-20-002990	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20180131	0	
1	0001640334-20-002990	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20170131	0	
2	0001640334-20-002995	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20190831	0	
3	0001640334-20-002995	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20200831	0	
4	0001640334-20-003002	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20191231	0	
...	...	...	...	...	...	
567315	0001477932-20-007602	WarrantsExercisedValue	0001477932-20-007602	20190630	0	
567316	0001477932-20-007602	WarrantsExercisedValue	0001477932-20-007602	20200331	4	
567317	0001477932-20-007602	WarrantsGranted	0001477932-20-007602	20190630	2	s
567318	0001477932-20-007602	WarrantsGranted	0001477932-20-007602	20190331	4	s
567319	0001477932-20-007602	WarrantsIssuedShares	0001477932-20-007602	20190331	4	s

567320 rows × 16 columns

Exclude all segments:

In [11]:
```
numbers = numbers[(numbers.dimh=='0x00000000')]  # Business segment (for example
numbers
```

Out[11]:

	adsh	tag	version	ddate	qtrs
0	0001640334-20-002990	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20180131	0
1	0001640334-20-002990	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20170131	0
2	0001640334-20-002995	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20190831	0
3	0001640334-20-002995	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20200831	0
4	0001640334-20-003002	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20191231	0

	adsh	tag	version	ddate	qtrs	
...	...	...	...	...	...	
311173	0001477932-20-007602	TotalAccruedMaintenanceFees	0001477932-20-007602	20200331	4	
311180	0001477932-20-007602	TotalSharesEarned	0001477932-20-007602	20200930	2	sł
311181	0001477932-20-007602	TotalSharesEarned	0001477932-20-007602	20200331	4	sł
311182	0001477932-20-007602	VestPercentageAtGrant	0001477932-20-007602	20200331	4	
311185	0001477932-20-007602	WorkingCapitalDeficit	0001477932-20-007602	20200331	0	

252766 rows × 16 columns

Merge the two tables on adsh, keep only rows that appear in both tables:

In [12]:
```
numbers.merge(filings, on='adsh', how='inner')
```

Out[12]:

	adsh	tag	version	ddate	qtrs
0	0001640334-20-002990	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20180131	0
1	0001640334-20-002990	AccountsPayableAndAccruedLiabilitiesCurrent	us-gaap/2019	20170131	0
2	0001640334-20-002990	AdditionalPaidInCapital	us-gaap/2019	20180131	0
3	0001640334-20-002990	AdditionalPaidInCapital	us-gaap/2019	20170131	0
4	0001640334-20-002990	Assets	us-gaap/2019	20180131	0
...	...	...	...	...	...
182725	0001096906-20-000515	SharesIssuedForOilAndGasProperty	0001096906-20-000515	20201031	1
182726	0001096906-20-000515	StockIssuedDuringPeriodValueSharesStockSplits	0001096906-20-000515	20201031	2
182727	0001096906-20-000515	SubsequentEvents	0001096906-20-000515	20200430	0
182728	0001096906-20-000515	SubsequentEvents	0001096906-20-000515	20201031	0

	adsh	tag	version	ddate	qtrs
182729	0001096906-20-000515	WorkingCapitalDeficit	0001096906-20-000515	20201031	0

182730 rows × 55 columns

Keep only these columns:

In [13]:
```
keep_these_columns = ['cik','sic','countryinc','tag','filed','ddate','qtrs','val
```

An now all together:

In [14]:
```
merged = numbers.merge(filings, on='adsh', how='inner')[keep_these_columns]
merged[:3]
```

Out[14]:

	cik	sic	countryinc	tag	filed	ddate	q
0	1699126	7372.0	US	AccountsPayableAndAccruedLiabilitiesCurrent	20201202	20180131	
1	1699126	7372.0	US	AccountsPayableAndAccruedLiabilitiesCurrent	20201202	20170131	
2	1699126	7372.0	US	AdditionalPaidInCapital	20201202	20180131	

How many days between 'filed' and 'ddate'?

In [15]:
```
date1 = merged.filed.iloc[0]
date1
```

Out[15]:
```
20201202
```

In [16]:
```
date2 = merged.ddate.iloc[0]

date1 - date2
```

Out[16]:
```
21071
```

→ this doesn't work. To calculate the difference between two calendar dates we need to first transform these integers into date objects:

In [17]:
```
pd.to_datetime(date1, format='%Y%m%d')
```

Out[17]:
```
Timestamp('2020-12-02 00:00:00')
```

In [18]:
```
pd.to_datetime(date1, format='%Y%m%d') - pd.to_datetime(date2, format='%Y%m%d')
```

Out[18]:
```
Timedelta('1036 days 00:00:00')
```

Interpret the 'filed' and 'ddate' as dates:

In [19]:
```
merged['filed'] = pd.to_datetime(merged.filed, format='%Y%m%d', errors='coerce')
merged['ddate'] = pd.to_datetime(merged.ddate, format='%Y%m%d', errors='coerce')
```

Remove any missing dates:

In [20]:
```
merged = merged[merged.filed.notnull() & merged.ddate.notnull()]
```

Remove duplicated rows:

In [21]:
```
merged = merged.drop_duplicates()
```

Save the merged file (you need to create the directory 'data/sec/merged/'):

In [22]:
```
merged.to_csv('data/sec/merged/'+folder+'.csv', index=False)
```

Put all of this into a function:

In [23]:
```
def merge_sec_files(folder):

    keep_these_columns = ['cik','sic','countryinc','tag','filed','ddate','qtrs',

    filings = pd.read_table('data/sec/downloads/'+folder+'/sub.tsv')
    numbers = pd.read_table('data/sec/downloads/'+folder+'/num.tsv', encoding='I

    filings = filings[filings.form.isin(['10-Q','10-K']) & filings.cik.notnull()
    numbers = numbers[(numbers.dimh=='0x00000000')]

    merged = numbers.merge(filings, on='adsh', how='inner')[keep_these_columns]

    merged['filed'] = pd.to_datetime(merged.filed, format='%Y%m%d', errors='coer
    merged['ddate'] = pd.to_datetime(merged.ddate, format='%Y%m%d', errors='coer

    merged = merged[merged.filed.notnull() & merged.ddate.notnull()].drop_duplic

    merged.to_csv('data/sec/merged/'+folder+'.csv', index=False)

    return merged
```

Use function like this:

In [24]:
```
merge_sec_files('2020_11')
```

```
/Users/janschneider/opt/anaconda3/lib/python3.7/site-packages/IPython/core/inter
activeshell.py:3338: DtypeWarning: Columns (12) have mixed types.Specify dtype o
ption on import or set low_memory=False.
  if (await self.run_code(code, result,  async_=asy)):
```

Out[24]:

	cik	sic	countryinc	tag	filed
0	1360901	6282.0	US	AccountsPayableAndAccruedLiabilitiesCurrent	2020-11-02
1	1360901	6282.0	US	AccountsPayableAndAccruedLiabilitiesCurrent	2020-11-02

	cik	sic	countryinc	tag	filed
2	1360901	6282.0	US	AccumulatedOtherComprehensiveIncomeLossNetOfTax	2020-11-02
3	1360901	6282.0	US	AccumulatedOtherComprehensiveIncomeLossNetOfTax	2020-11-02
4	1360901	6282.0	US	AdditionalPaidInCapital	2020-11-02
...	...	...	...	...	...
1690320	1003815	6513.0	US	EntityCommonStockSharesOutstanding	2020-11-12
1690321	1003815	6513.0	US	SubscriptionPayable	2020-11-12
1690322	1003815	6513.0	US	SubscriptionPayable	2020-11-12
1690323	1776909	7812.0	US	EntityCommonStockSharesOutstanding	2020-11-16
1690324	1261379	6221.0	US	EntityCommonStockSharesOutstanding	2020-11-13

1504170 rows × 8 columns

And now loop over all files:

In [ ]:
```
for folder in os.listdir('data/sec/downloads/'):
    print(folder)
    merge_sec_files(folder)
```