File Validator Script

In [230]:

```
import pandas as pd
import numpy as np
import re

pd.set_option('display.max_columns', None)
pd.options.mode.chained_assignment = None  # default='warn'
```

In [209]:

```
valid = pd.read_excel('Validation.xlsx')
```

In [210]:

```
valid
```

Out[210]:

	HEADER NAME	MISSING ERROR	FORMAT ERROR
0	DATE	Error Message if header is missing: DATE is a ...	Acceptable formats:\nM/D/YYYY\nYYYYMMDD\n\nErr...
1	ID	Error Message if header is missing: ID is a re...	Acceptable formats: 60 characters alpha numeri...
2	SE	Error Message if header is missing: SE is a re...	Acceptable formats: 7 characters alpha numeric...
3	CU	Error Message if header is missing: CU is a re...	Acceptable formats: 9 characters alpha numeric...
4	IS	Error Message if header is missing: IS is a re...	Acceptable formats: 12 characters alpha numeri...
5	DE	Error Message if header is missing: DE is a re...	NaN

In [211]:

```
df_test = pd.read_csv('testing_file.csv')
```

Header Checker Function

In [228]:

```
def header_checker(df_test):
    header_list = ['DATE','ID','SE','CU','IS','DE']
    missing_headers = list(set(header_list) - set(list(df_test)))
    if len(missing_headers):
        print('Missing headers')
        for header in missing_headers:
            print('{} is a required field'.format(header))
        print('Some headers are missing, please address the issues before proceeding')
    else:
        print('All headers found, proceeding to check individual headers')

        alnum_headers = {'ID':[60,'60 characters alpha numeric with no consecutive spaces'],
                         'SE':[7,'Invalid SE format'],
                         'CU':[9,'Invalid CU format'],
                         'IS':[12,'Invalid CU format']}

        df = df_test.loc[df_test['CD']=='EQ']

        for header,message in alnum_headers.items():
            df['{} Checker'.format(header)] = np.where(df['{}'.format(header)].str.isalnum() & (df['
'{}'.format(header)].str.len()<=message[0]),
                                       'Correct Format','{}'.format(message[1]))
            df['{} Checker'.format(header)] = np.where(df['{}'.format(header)].str.isspace(),'{}'.f
ormat(message[1]),df['{} Checker'.format(header)])
```

```
            df['{} Checker'.format(header)] = np.where(df['{}'.format(header)].isna(),'Missing {}'.
format(header),df['{} Checker'.format(header)])

        regpat = '^((0|1)\d{1})\/((0|1|2)\d{1})\/((19|20)\d{2})|\d{4}(0[1-9]|1[012])(0[1-9]|[12][0-
9]|3[01])'

        df['DATE Checker'] = np.where(df['DATE'].str.match(regpat),'Correct Format','DATE format sh
ould be MM/DD/YYYY or YYYYMMDD')
        df['DATE Checker'] = np.where(df['DATE'].isna(),'Missing Date',df['DATE Checker'])

        df['DE Checker'] = np.where(df['DE'].isna(),'Missing DE','Correct Format')

        display(df)
```

Test Validator

In [231]:

```
temp1 = df_test.drop(columns='DATE',axis=1)
temp2 = df_test.drop(columns=['ID','SE'])

print('\n')
header_checker(temp1)
print('\n')
header_checker(temp2)
print('\n')
header_checker(df_test)
```

```
Missing headers
DATE is a required field
Some headers are missing, please address the issues before proceeding


Missing headers
SE is a required field
ID is a required field
Some headers are missing, please address the issues before proceeding


All headers found, proceeding to check individual headers
```

	CD	DATE	ID	SE	CU	IS	DE	ID Checker
0	EQ	20190907	Test123	AED1134	ER1234567	SMT@12345678	Test1	Correct Format
1	EQ	9/7/2019	NaN	EWQ2345	ZR123456712	NaN	Test2	Missing ID
2	EQ	NaN	Test456	NaN	TEST12123	SMT@1234567890	Test3	Correct Format
3	EQ	##...	Test123	1234QW1	NaN	NaN	NaN	Correct Format
4	EQ	##...	try987	NaN	NaN	TRS1234543212	NaN	Correct Format
5	EQ	##...	Excel	567@123890	TR1234523	NaN	Test4	Correct Format
6	EQ	##	try987	NaN	NaN	TRS1234543212	NaN	Correct

	CD	DATE	ID	SE	CU	IS	DE	Format ID Checker
7	EQ	##...	Excel!	567@123890	TR1234523	NaN	Test4	60 characters alpha numeric with no consecutiv...

In [227]:

```
df = df_test.loc[df_test['CD']=='EQ']
```