Data Science Course: Capstone Project 1

Capstone Mini-Project: Data Wrangling

1. What kind of cleaning steps did you perform?

Setting header and global index column when importing data.

```
	pd.read_csv('Data/Fantasy_Stats_2018-19.csv', header=1, index_col='Rk')
```

Sorting the data by year by creating a ‘Year’ column to the newly imported data file.

```
	fantasy_stats_2018['Year'] = '2018'
```

Cleaning column names by using the .rename method.

```
	fantasy_stats_2018 = fantasy_stats_2018.rename(columns={'VBD▼': 'VBD'})
```

Cleaning up player names by stripping additional characters attached to the end of the name by using .str.strip() .str.replace(,) and .str.partition() methods.

```
	qb['Player'] = qb['Player'].str.strip('*+')
	qb_week_leaders_2018['Player'].str.replace('\\', "..")
	qb_week_leaders_2018['Player'].str.partition("..")
```

I used these same cleaning steps for the running backs, wide receivers, and tight end data sources.

2. How did you deal with missing values, if any?

When running into null or missing values I used .fillna(0) method. This method was able to solve all missing value issues when dealing with numeric data.

```
	qb = qb.fillna(0)
```

Again, I used this to deal with missing values across all of my different data sources.

3. Were there outliers, and how did you handle them?

Yes, I used on data set from a database. This called for me to use SQL to read the data into my Jupyter notebook. To integrate this data I had to use SQLite, which meant creating a cursor to use .execute on where I was then able to write in SQL to sort and clean the data.

```
	conn = sqlite3.connect('Data/nfl.db')
cursor = conn.execute('SELECT * FROM stats_offense')

# Query Database
```

```
cursor.execute('''
SELECT * FROM stats_offense
WHERE season > 2013
''')
stats_df = cursor.fetchmany(10)
stats_df = pd.DataFrame(stats_df)

cols = [i[0] for i in cursor.description]
print(cols)

stats_df.columns = cols
stats_df
```