[17]:
```
sample_fielders = modified_fielders_df[["Name", "Team", "Position"]]
```

[18]:
```
Sample_Pitcher_Data = sample_pitchers.sort_values('Name', ascending = True)
```

[19]:
```
Sample_Fielder_Data = sample_fielders.sort_values('Name', ascending = True)
```

[ ]:
```

```

Save our new dataframes

[20]:
```
Sample_Pitcher_Data.to_excel('Project1_Sampled_Pitchers_Data.xlsx', index =␣
↪False, header = True)
```

[21]:
```
Sample_Fielder_Data.to_excel('Project1_Sample_Fielders_Data.xlsx', index =␣
↪False, header = True)
```

Switching over to excel I entered the total games that each player in both data frames played. For pitchers no matter if they batted or not, if they entered the game at any point it is considered a game. For batters any game they entered no matter if it was in a pinch situation counts.We then saw a pattern that seemed that for batters that had higher wOBA and wRC+ seemed to correlate to how many MLB Games they've appeared in since 2012 to the present day. With pitchers we saw that xFIP and K/9 correlated with higher number of MLB Games. I'm going to start by reloading in my new data set as a csv with our new relevant stats.

[22]:
```
Game_Data_Fielders = pd.read_csv('/Users/thomasoakley/Project_1_2012_Datasets/
↪Project1_Sampled_Fielders_Data.csv')
```

[23]:
```
Game_Data_Fielders.head()
```

[23]:

	Name	Team	Position	MLB Appearances	wOBA	wRC+	Score
0	Khris Davis	Brewers	OF	938	0.507	218	1.894762
1	Tommy La Stella	Braves	2B	531	0.483	207	1.801667
2	Travis d'Arnaud	Mets	C	544	0.483	207	1.801667
3	Dan Vogelbach	Cubs	1B	244	0.466	180	1.639683
4	Adam Eaton	D-Backs	OF	831	0.435	161	1.495476

[24]:
```
Game_Data_Fielders.describe()
```

[24]:

	MLB Appearances	wOBA	wRC+	Score
count	75.000000	75.000000	75.000000	75.000000
mean	192.160000	0.342813	105.066667	1.069481
std	275.857486	0.061567	41.249188	0.298347
min	0.000000	0.109000	-43.000000	-0.041984
25%	0.000000	0.312500	85.500000	0.934881
50%	40.000000	0.340000	105.000000	1.082143
75%	294.500000	0.370500	127.000000	1.225040
max	938.000000	0.507000	218.000000	1.894762