In [1]:
```
#Import libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import folium
import webbrowser
from IPython.display import set_matplotlib_formats
from matplotlib import rcParams
import warnings
```

In [2]:
```
warnings.filterwarnings('ignore')
```

In [3]:
```
df_costco=pd.read_csv("costcos-geocoded.csv")
df_ppg=pd.read_csv("ppg2008.csv",index_col=0)
```

In [4]:
```
df_costco.head()
```

Out[4]:

	Address	City	State	Zip Code	Latitude	Longitude
0	1205 N. Memorial Parkway	Huntsville	Alabama	35801-5930	34.743095	-86.600955
1	3650 Galleria Circle	Hoover	Alabama	35244-2346	33.377649	-86.812420
2	8251 Eastchase Parkway	Montgomery	Alabama	36117	32.363889	-86.150884
3	5225 Commercial Boulevard	Juneau	Alaska	99801-7210	58.359200	-134.483000
4	330 West Dimond Blvd	Anchorage	Alaska	99515-1950	61.143266	-149.884217

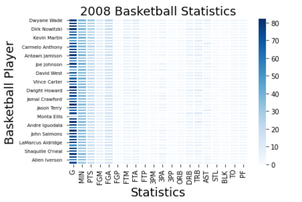
In [5]:
```
df_ppg.head()
```

Out[5]:

Name	G	MIN	PTS	FGM	FGA	FGP	FTM	FTA	FTP	3PM	3PA	3PP	ORB	DRB	TRB	AST	STL	BLK	TO	PF
Dwyane Wade	79	38.6	30.2	10.8	22.0	0.491	7.5	9.8	0.765	1.1	3.5	0.317	1.1	3.9	5.0	7.5	2.2	1.3	3.4	2.3
LeBron James	81	37.7	28.4	9.7	19.9	0.489	7.3	9.4	0.780	1.6	4.7	0.344	1.3	6.3	7.6	7.2	1.7	1.1	3.0	1.7
Kobe Bryant	82	36.2	26.8	9.8	20.9	0.467	5.9	6.9	0.856	1.4	4.1	0.351	1.1	4.1	5.2	4.9	1.5	0.5	2.6	2.3
Dirk Nowitzki	81	37.7	25.9	9.6	20.0	0.479	6.0	6.7	0.890	0.8	2.1	0.359	1.1	7.3	8.4	2.4	0.8	0.8	1.9	2.2
Danny Granger	67	36.2	25.8	8.5	19.1	0.447	6.0	6.9	0.878	2.7	6.7	0.404	0.7	4.4	5.1	2.7	1.0	1.4	2.5	3.1

Heat Map

In [6]:
```
sns.heatmap(df_ppg,cmap='Blues',linewidth=0.30)
plt.xticks(rotation=90)
plt.yticks(fontsize=7)
plt.xlabel("Statistics",size=18)
plt.ylabel("Basketball Player",size=18)
plt.title("2008 Basketball Statistics",size=18)
plt.show()
```



Spatial Plot

In [7]:
```
map=folium.Map(prefer_canvas=True)

def plot(point):
    folium.CircleMarker(location=[point.Latitude,point.Longitude],
                        radius=2, popup=point.Address+" "+point.City+" "+point.State+" "+point["Zip Code"],
                        fill_color=point.State,
                        fill=True,
                        fill_opacity=0.7,
                        weight=5).add_to(map)
```

In [8]:
```
df_costco.apply(plot,axis=1)
map.fit_bounds(map.get_bounds())
output_file="Map_Python.html"
map.save(output_file)
webbrowser.open(output_file,new=2)
```

Out[8]:
```
True
```

Contour Plot

In [9]:
```
Z=df_ppg.pivot_table(index='FGA',columns='PTS',values='FGP').T.values
x_unique=np.sort(df_ppg.FGA.unique())
y_unique=np.sort(df_ppg.PTS.unique())
```

In [10]:
```
X,Y=np.meshgrid(x_unique,y_unique)
Z[np.isnan(Z)]=0
pd.DataFrame(Z).round(3)
pd.DataFrame(Y).round(3)
pd.DataFrame(X).round(3)
```

Out[10]:

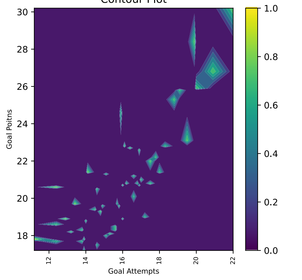
	0	1	2	3	4	5	6	7	8	9	...	29	30	31	32	33	34	35	36	37	38
0	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
1	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
2	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
3	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
4	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
5	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
6	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
7	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
8	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
9	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
10	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
11	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
12	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
13	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
14	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
15	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
16	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
17	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
18	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
19	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
20	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
21	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
22	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
23	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
24	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
25	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
26	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
27	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
28	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
29	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
30	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
31	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
32	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
33	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
34	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
35	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
36	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
37	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
38	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0

39 rows × 39 columns

In [11]:
```
set_matplotlib_formats('svg')
rcParams['figure.figsize']=5,5
fig=plt.figure()
ax=fig.add_subplot(111)

cp=ax.contourf(X,Y,Z)
ax=plt.gca()
PCM=ax.get_children()[2]
plt.colorbar(PCM,ax=ax)

plt.xticks(rotation=90)
plt.xticks(fontsize=7)
plt.xlabel('Goal Attempts',size=8)
plt.ylabel('Goal Poitns',size=8)
plt.title("Contour Plot")
plt.show()
```



In [ ]:
```
```

In [ ]:
```
```