Plotly 를 활용한 Titanic 생존자 데이터 분석

In [2]:

```
import plotly
import cufflinks as cf
import pandas as pd
import numpy as np
```

In [3]:

```
print(plotly.__version__)
print(cf.__version__)
print(pd.__version__)
print(np.__version__)
```

```
4.13.0
0.17.3
1.0.5
1.18.5
```

In [4]:

```
# 오프라인 모드에서도 인터렉티브한 그래픽을 가능하도록 하기
# Enabling the offline mode for interactive plotting locally

from plotly.offline import download_plotlyjs,init_notebook_mode,plot,iplot
init_notebook_mode(connected=True)
cf.go_offline()
```

In [6]:

```
train = pd.read_csv("train.csv")
train
```

Out[6]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500

891 rows × 12 columns

In [19]:

```
1 train.describe()
```

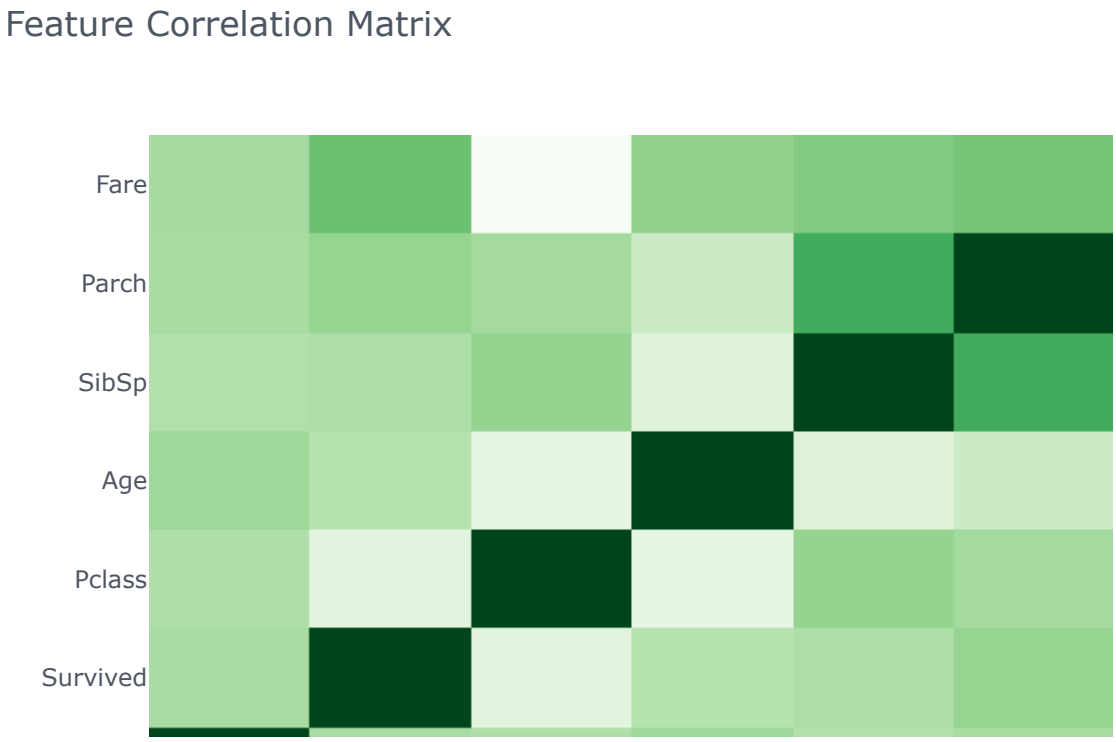
Out[19]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

- Heatmap

In [31]:

```
train.corr().iplot(kind='heatmap',colorscale="greens", title="Feature Correlation Matrix")
```



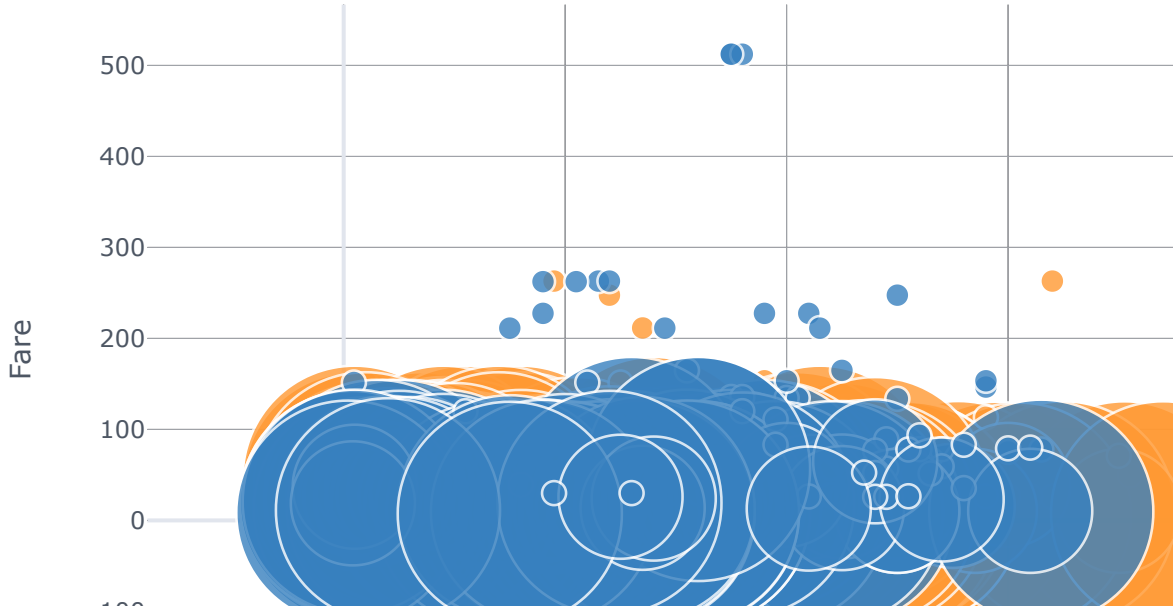
Heatmap => 각 칼럼간의 상관관계를 행렬로 나타내준다.

- Bubble

In [14]:

```
train[['Survived']] = train[['Survived']].astype('float64', copy=False)

train.iplot(kind="bubble", x="Age", y="Fare", categories="Survived",
            size='Pclass', text='Name', xTitle='Age', yTitle='Fare')
```



- Bubble => 나이와 운임요금간의 관계를 보여주고 Survived 와 Non_Survived 의 차이를 색으로 구분한다.

In [ ]:

```
```