ppt 강의에서 예시로 사용했던 titanic EDA 코드입니다.

In [3]:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
# notebook을 실행한 브라우저에서 바로 그림을 볼 수 있게 해주는 코드
```

In [4]:

```
df = pd.read_csv("titanic.csv")
```

변수 설명입니다.

변수명	변수명 설명
survival	Survival (0 = No; 1 = Yes)
pclass	Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)
name	Name
sex	Sex
age	Age
sibsp	Number of Siblings/Spouses Aboard
parch	Number of Parents/Children Aboard
ticket	Ticket Number
fare	Passenger Fare
cabin	Cabin
embarked	Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

1. 데이터의 모양 확인하기

head, tail 통해서 확인

In [7]:

```
display(df.head())
display(df.tail())
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Ca
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	C
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabi
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	Nal
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	B4
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	Nal
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C14
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	Nal

In [18]:

```
# 또는 그냥 df 출력시켜서 볼 수도 있음
df
```

Out[18]:

	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

2. 각 데이터의 타입 확인하기

- int64: 정수형 데이터
- float64: 실수형 데이터

- object: 문자열 데이터

In [15]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
PassengerId    891 non-null int64
Survived       891 non-null int64
Pclass         891 non-null int64
Name           891 non-null object
Sex            891 non-null object
Age            714 non-null float64
SibSp          891 non-null int64
Parch          891 non-null int64
Ticket         891 non-null object
Fare           891 non-null float64
Cabin          204 non-null object
Embarked       889 non-null object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

In [17]:

```
# 행과 열의 개수
df.shape
```

Out[17]:

```
(891, 12)
```

In [157]:

```
print(set(df["Sex"]))
print(set(df["Embarked"]))
```

```
{'female', 'male'}
{nan, 'C', 'Q', 'S'}
```

3. 결측치 확인하기

In [26]:

```
df.isnull().sum()
```

Out[26]:

```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

In [90]:

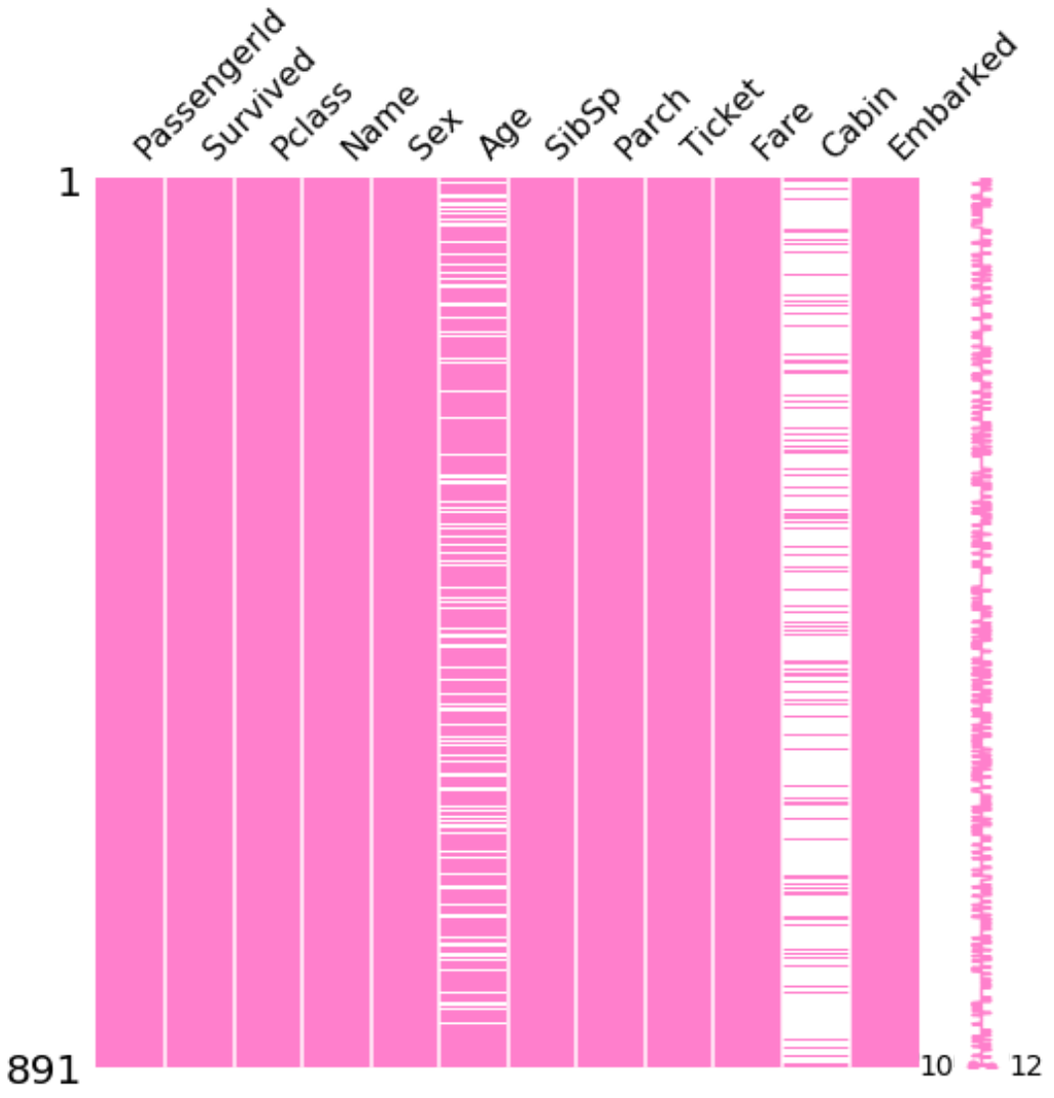
```
# msno 사용하면 좀 더 예쁘게 결측치 볼 수 있어요
import missingno as msno
```

In [51]:

```
msno.matrix(df=df, figsize=(8, 8), color=(1, 0.5, 0.8))
```

Out[51]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x267566d6b48>
```



4. 이상치 확인하기

In [129]:

```
df["Fare"].describe()
```

Out[129]:

```
count    891.000000
mean      32.204208
std       49.693429
min        0.000000
25%        7.910400
50%       14.454200
75%       31.000000
max      512.329200
Name: Fare, dtype: float64
```

In [162]:

```
df[df["Fare"] > 300]
```

Out[162]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cab
258	259	1	1	Ward, Miss. Anna	female	35.0	0	0	PC 17755	512.3292	N
679	680	1	1	Cardeza, Mr. Thomas Drake Martinez	male	36.0	0	1	PC 17755	512.3292	B B B
737	738	1	1	Lesurer, Mr. Gustave J	male	35.0	0	0	PC 17755	512.3292	B1

5. 종속변수 확인하기

kaggle에서는 타이타닉 데이터로 survived를 예측하는 문제를 냅니다.
즉 survived 값이 종속변수, 다른 값들이 독립변수

In [86]:

```
df["Survived"].value_counts()
```

Out[86]:

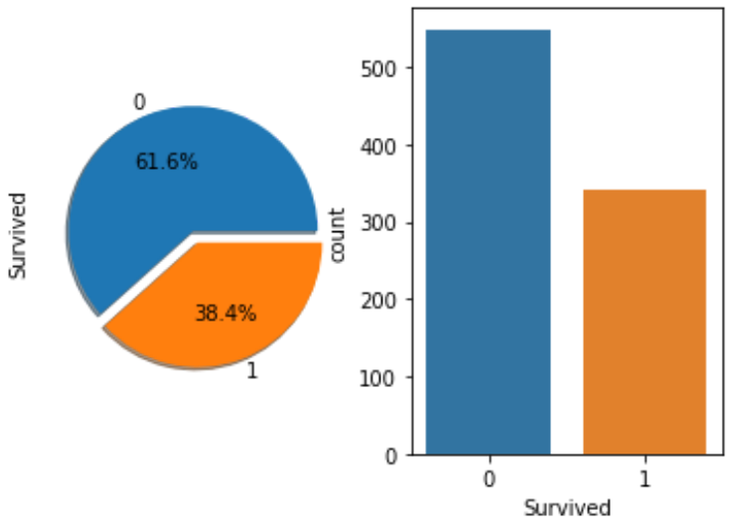
```
0    549
1    342
Name: Survived, dtype: int64
```

In [84]:

```
f, ax = plt.subplots(1,2)
df['Survived'].value_counts().plot.pie(explode=[0,0.1], autopct='%1.1f%%', ax=ax[0],
sns.countplot("Survived", data = df, ax = ax[1])
```

Out[84]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x267588dc608>
```



6. 각 변수의 분포 살펴보기

- Pclass: 자리 등급을 나타내는 데이터 (1,2,3등급)

Pclass에 따라 몇명이 생존했는지를 나타내봅시다

In [112]:

```
display(df["Pclass"].value_counts())
display(df[["Pclass", "Survived"]].groupby(["Pclass"]).sum())
```

```
3    491
1    216
2    184
Name: Pclass, dtype: int64
```

Pclass	Survived
1	136
2	87
3	119

In [131]:

```
df["Pclass"].value_counts()
```

Out[131]:

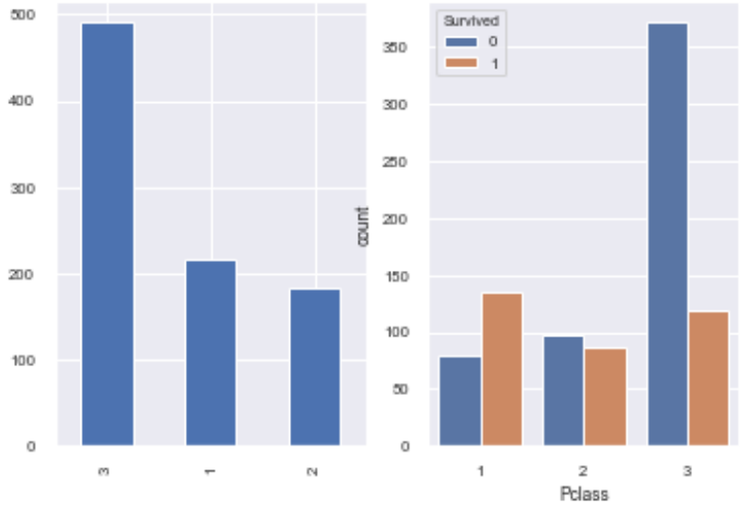
```
3    491
1    216
2    184
Name: Pclass, dtype: int64
```

In [163]:

```
# 숫자로도 파악할 수 있지만, 그래프로 보면 더 직관적이에요
f,ax = plt.subplots(1,2)
df["Pclass"].value_counts().plot(kind = "bar", ax = ax[0])
sns.countplot(x = "Pclass", hue = "Survived", data = df, ax = ax[1])
```

Out[163]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x267700d5348>
```

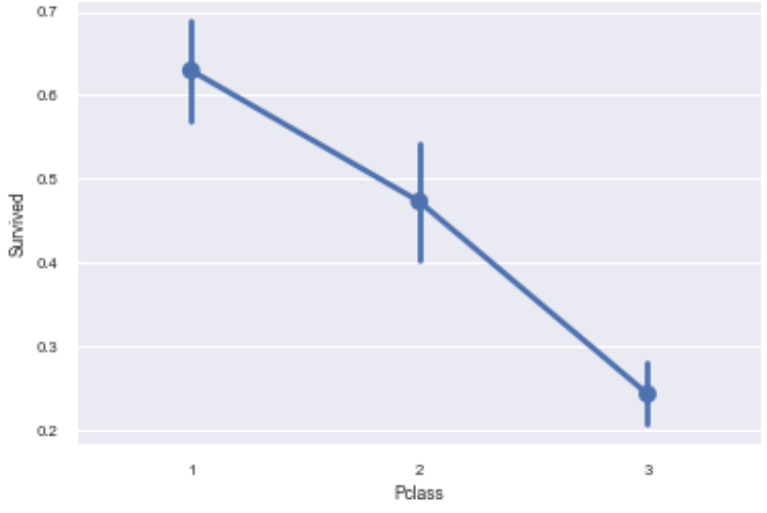


In [143]:

```
sns.pointplot(x = "Pclass", y = "Survived", data = df)
```

Out[143]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2676fc46808>
```

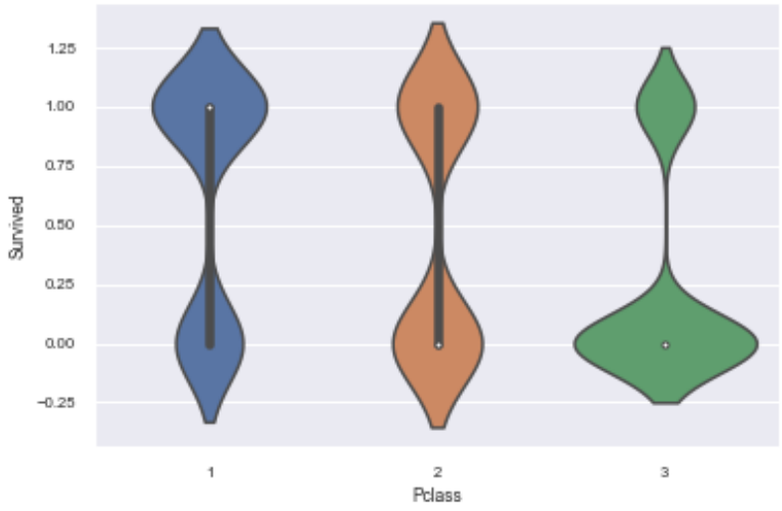


In [144]:

```
sns.violinplot(x = "Pclass", y = "Survived", data = df)
```

Out[144]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2676fc73388>
```



- Sex

성별에 따라선 어떻게 다른지 비교해봅시다

In [145]:

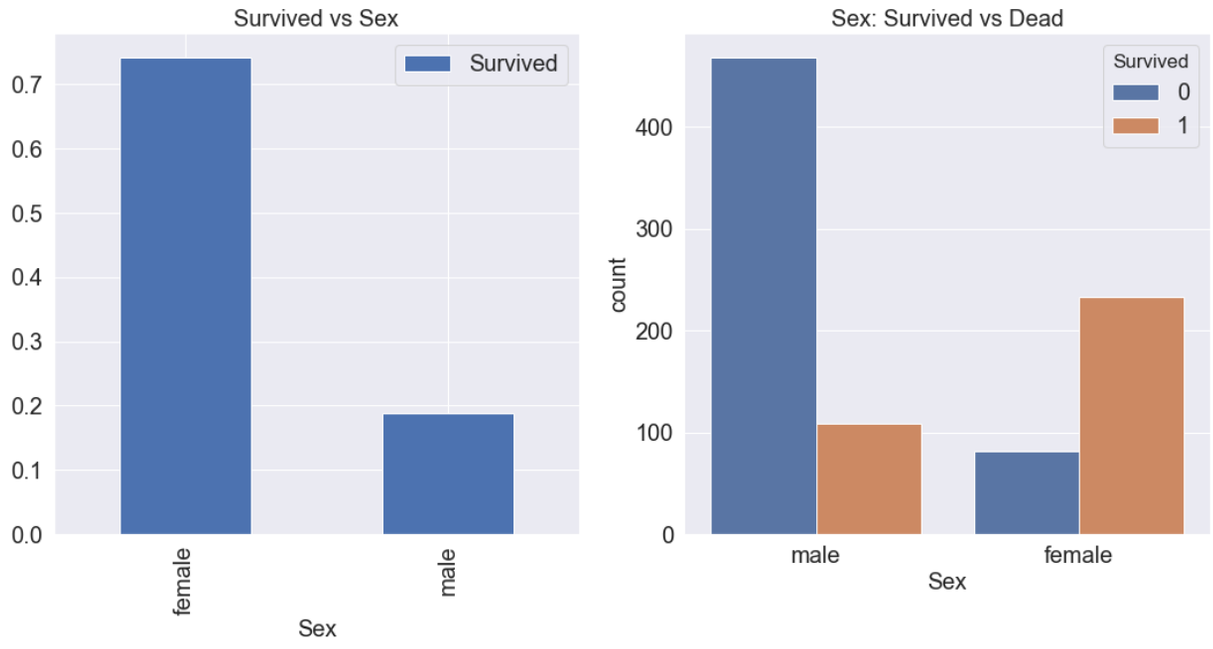
```
df.groupby(['Survived','Sex'])['Survived'].count()
```

Out[145]:

```
Survived  Sex
0         female     81
          male      468
1         female    233
          male      109
Name: Survived, dtype: int64
```

In [181]:

```
f, ax = plt.subplots(1, 2, figsize=(18, 8))
plt.rc("axes", titlesize = 20)
plt.rc("legend", fontsize = 20)
plt.rc("ytick", labelsize = 20)
plt.ylabel('y',size=20)
df[['Sex', 'Survived']].groupby(['Sex'], as_index=True).mean().plot.bar(ax=ax[0])
ax[0].set_title('Survived vs Sex')
sns.countplot('Sex', hue='Survived', data=df, ax=ax[1])
ax[1].set_title('Sex: Survived vs Dead')
plt.show()
```

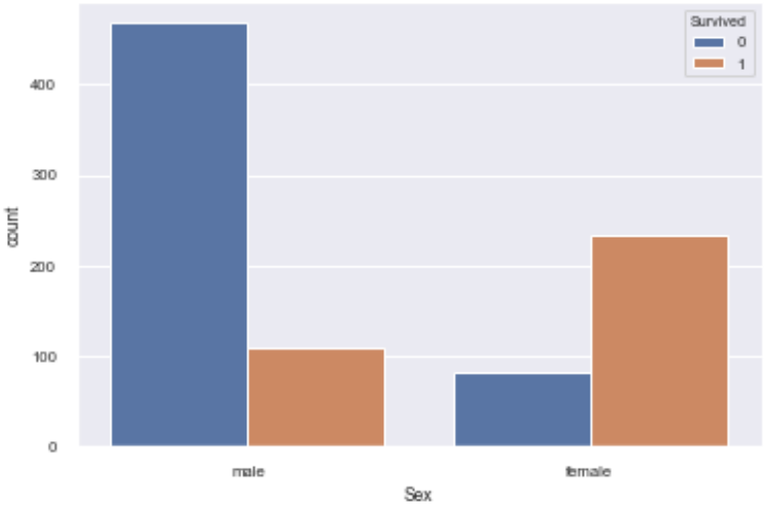


In [153]:

```
sns.countplot(x = "Sex",  hue = "Survived", data = df)
```

Out[153]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2676ff78a48>
```

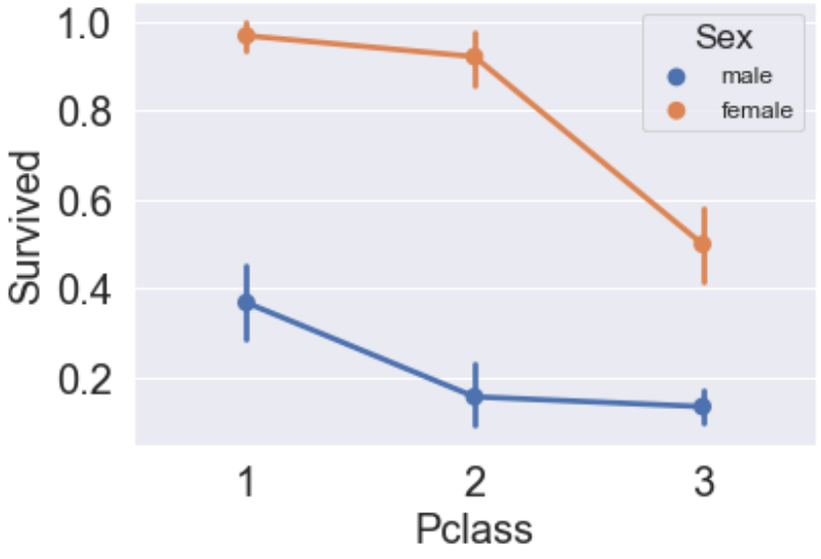


In [184]:

```
# 두개를 동시에 볼 수도 있어요
plt.rc("legend", fontsize = 12)
sns.pointplot(x = "Pclass", y = "Survived", hue = "Sex", data = df)
```

Out[184]:

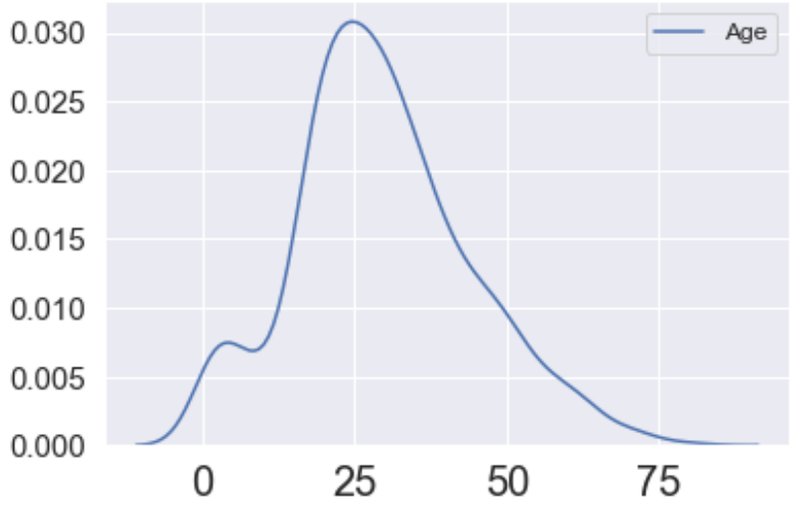
```
<matplotlib.axes._subplots.AxesSubplot at 0x26773a32e88>
```



- Age

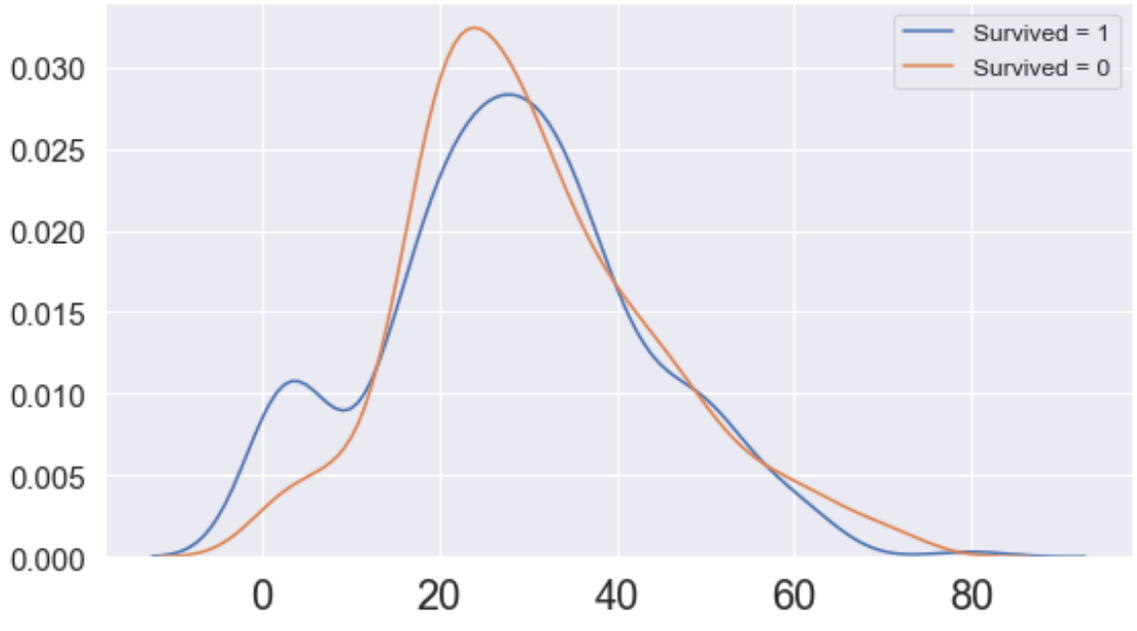
In [212]:

```
sns.kdeplot(df['Age'])
plt.show()
```



In [197]:

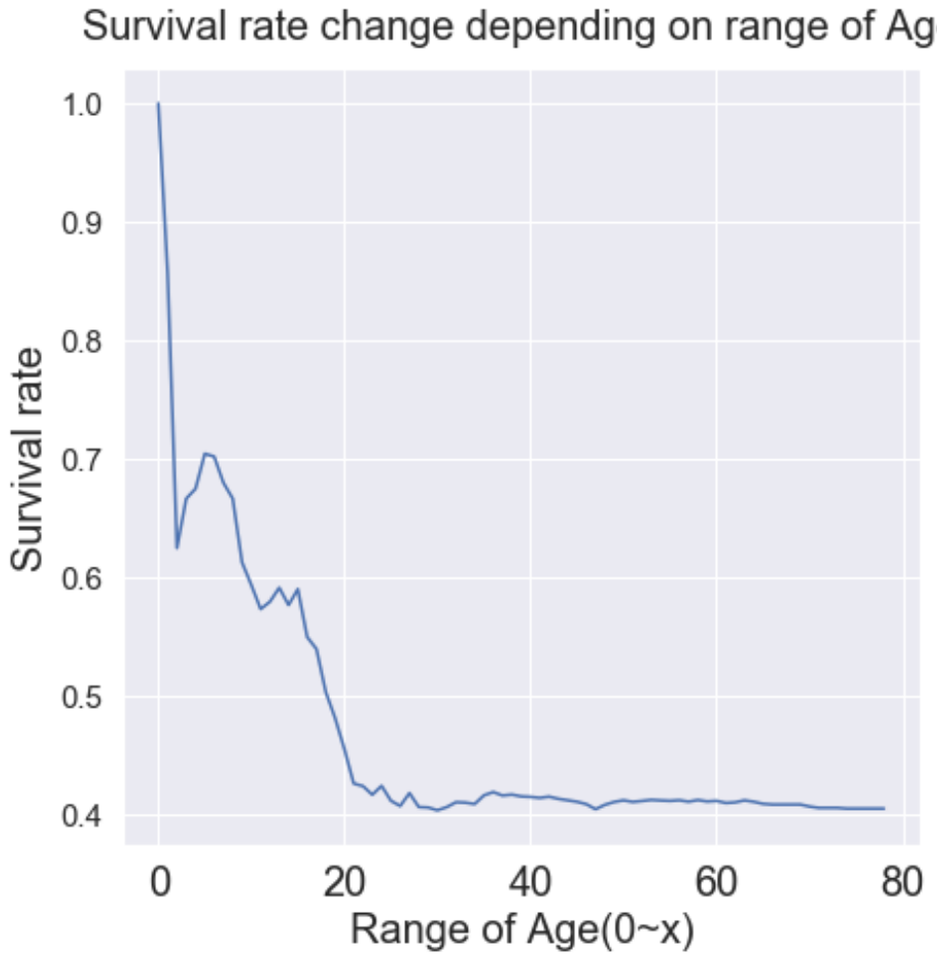
```
plt.rc("ytick", labelsize = 15)
fig, ax = plt.subplots(1, 1, figsize=(9, 5))
sns.kdeplot(df[df['Survived'] == 1]['Age'], ax=ax)
sns.kdeplot(df[df['Survived'] == 0]['Age'], ax=ax)
plt.legend(['Survived = 1', 'Survived = 0'])
plt.show()
```



In [214]:

```
cummulate_survival_ratio = []
for i in range(1, 80):
    cummulate_survival_ratio.append(df[df['Age'] < i]['Survived'].sum() / len(df[df[

plt.figure(figsize=(7, 7))
plt.plot(cummulate_survival_ratio)
plt.title('Survival rate change depending on range of Age', y=1.02)
plt.ylabel('Survival rate')
plt.xlabel('Range of Age(0~x)')
plt.show()
```

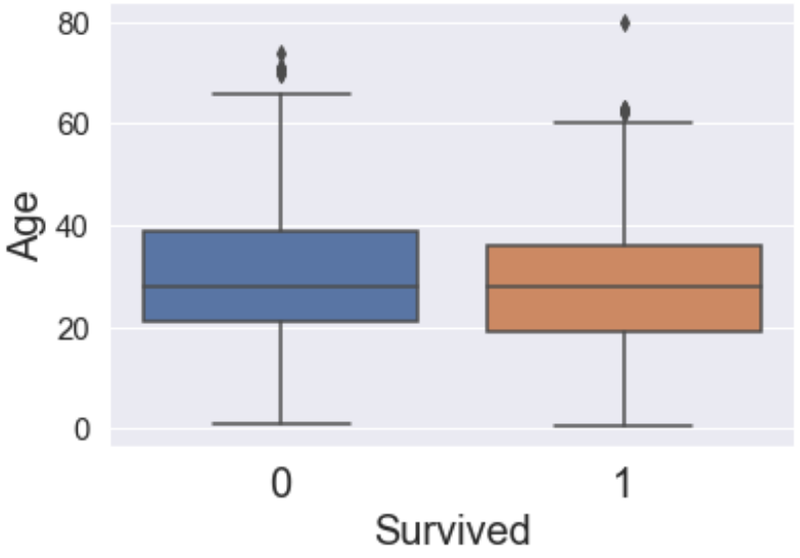


In [213]:

```
sns.boxplot( x = "Survived", y = "Age", data = df)
```

Out[213]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x26775d207c8>
```



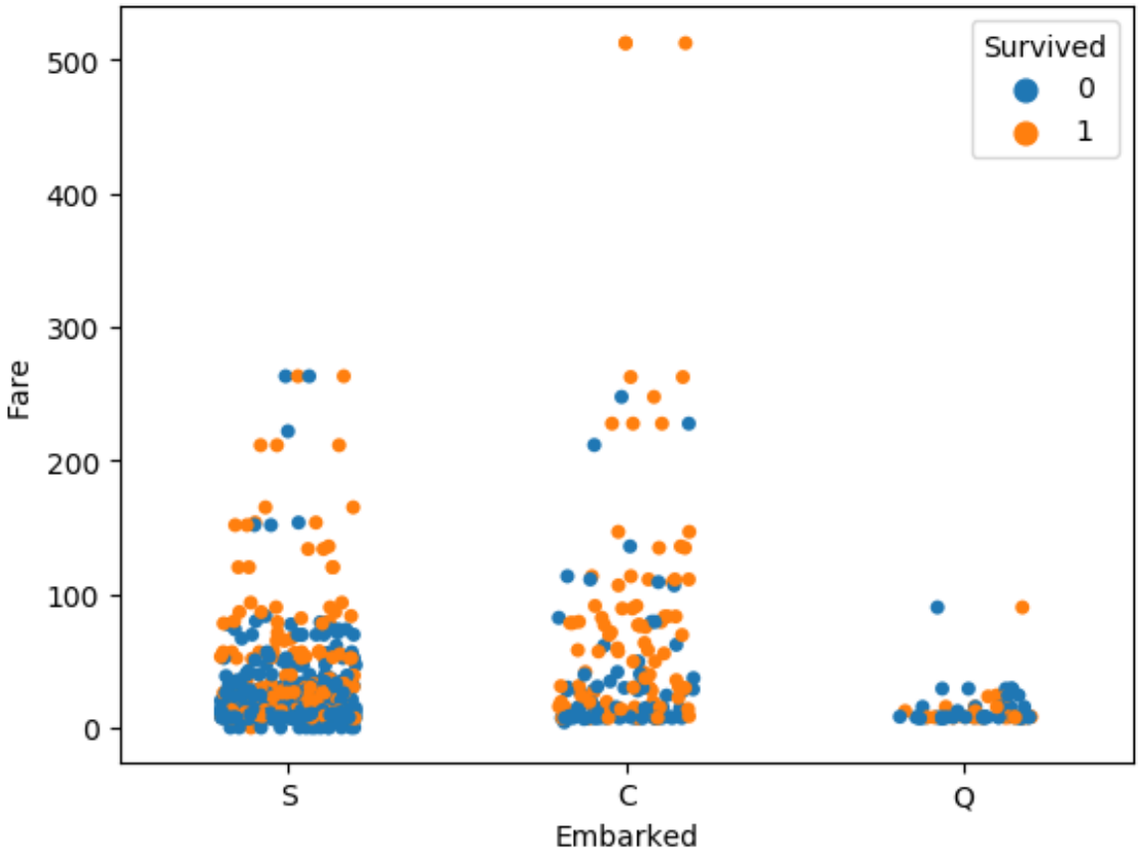
- Embarked

In [244]:

```
sns.stripplot( x = "Embarked", y = "Fare", hue = "Survived", data = df, jitter = 0.2
```

Out[244]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2677ae357c8>
```



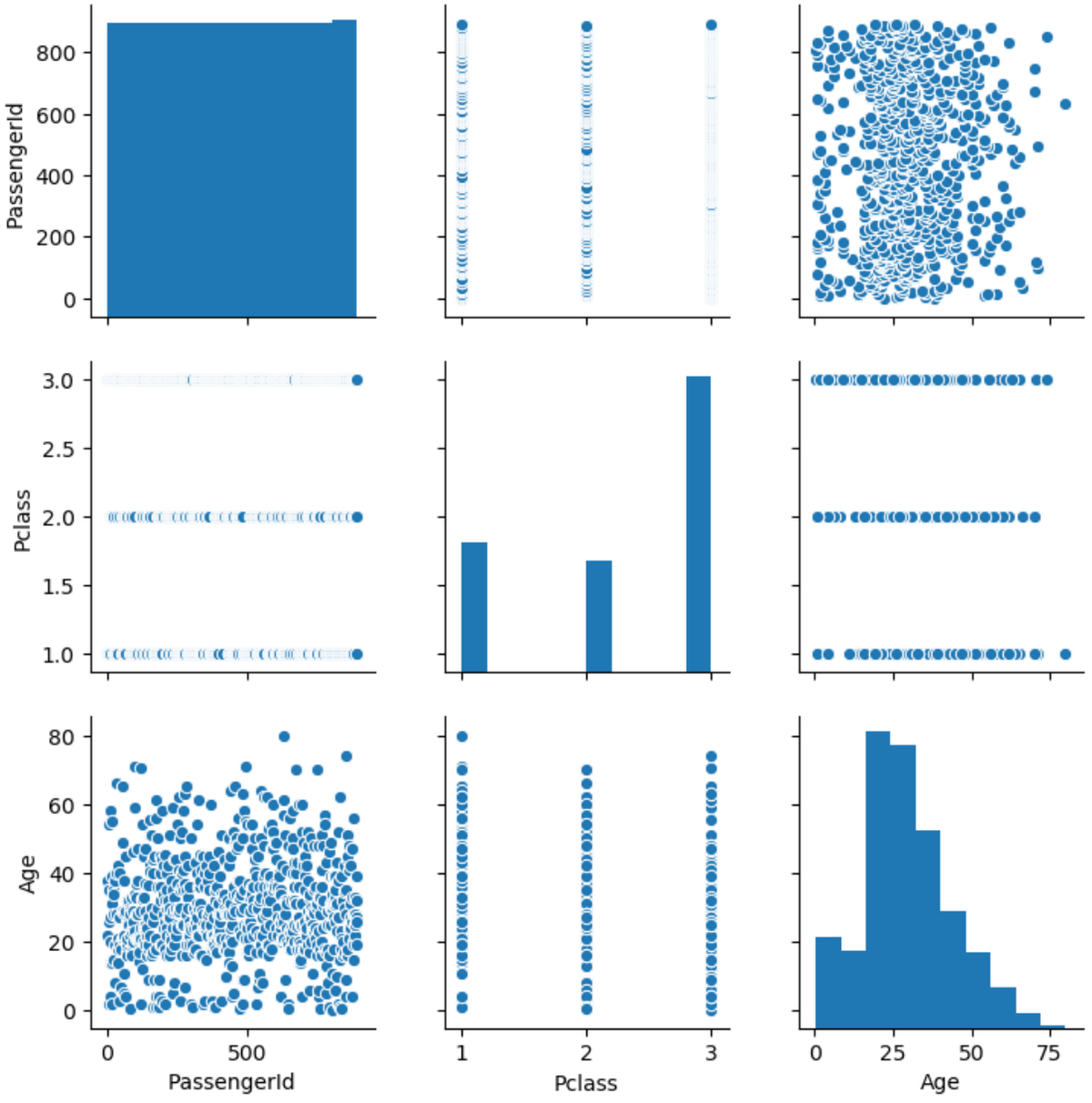
In [243]:

```
# 이렇게 한번에 그려볼 수도 있어요
# 대각성분은 histogram, 나머지는 scatterplot입니다
sns.reset_defaults()
sns.pairplot(df, vars = ["PassengerId","Pclass","Age"])
```

```
C:\Users\wkdgu\Anaconda3\lib\site-packages\numpy\lib\histograms.py:82
4: RuntimeWarning: invalid value encountered in greater_equal
  keep = (tmp_a >= first_edge)
C:\Users\wkdgu\Anaconda3\lib\site-packages\numpy\lib\histograms.py:82
5: RuntimeWarning: invalid value encountered in less_equal
  keep &= (tmp_a <= last_edge)
```

Out[243]:

```
<seaborn.axisgrid.PairGrid at 0x2677a818ac8>
```



In [246]:

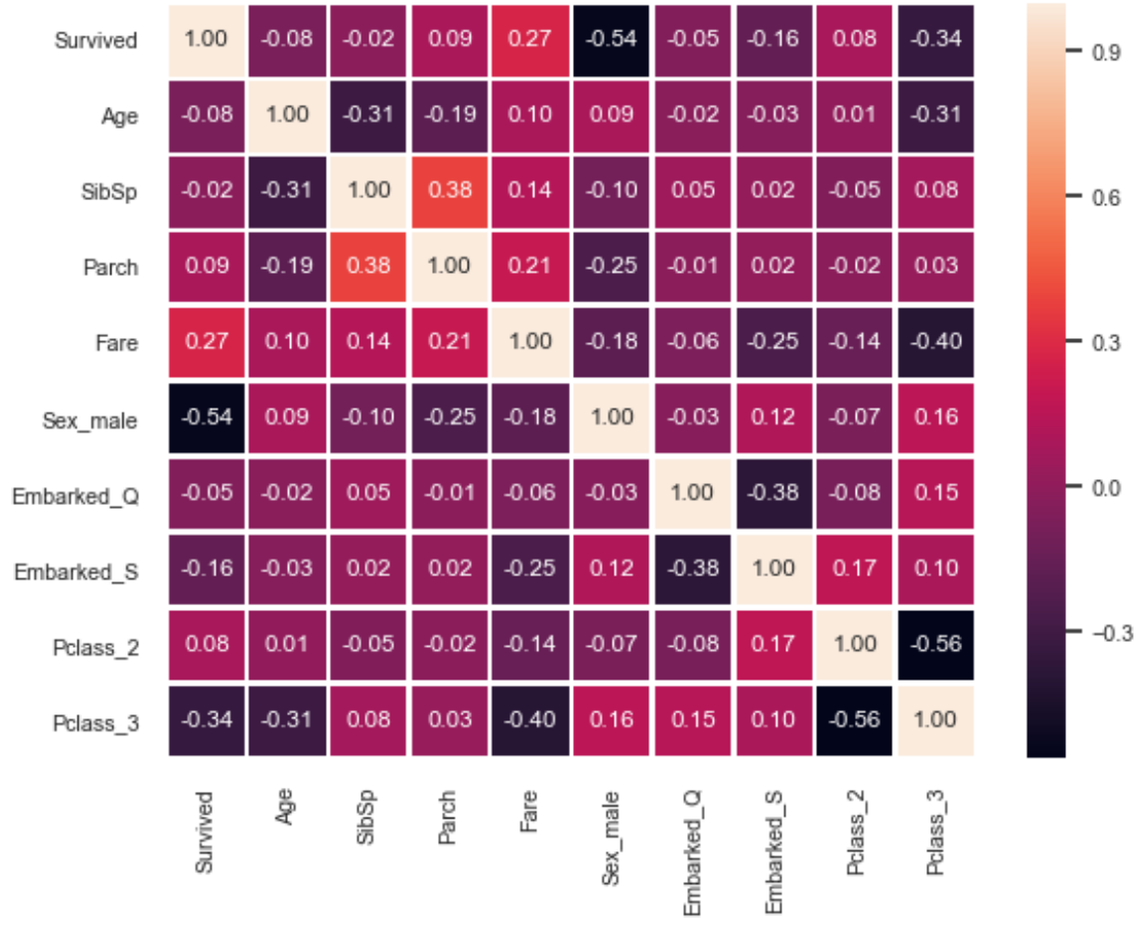
```
df_ml = df.copy()

df_ml = pd.get_dummies(df_ml, columns=['Sex', 'Embarked', 'Pclass'], drop_first=True
# 필요없는 column과 na값 drop
df_ml.drop(['PassengerId','Name','Ticket', 'Cabin'],axis=1,inplace=True)
df_ml.dropna(inplace=True)

sns.set(font_scale = 0.7)
ax = sns.heatmap(df_ml.corr(), annot = True, fmt = ".2f", linewidths = 1)
buttom, top = ax.get_ylim()
ax.set_ylim(buttom + 0.5, top - 0.5)
```

Out[246]:

```
(10.0, 0.0)
```



In [ ]: