

# analysis on survival in disaster

In [88]:

```
1 import numpy as np
2 import pandas as pd
3 from sklearn import preprocessing
4 import matplotlib.pyplot as plt
5 import seaborn as sb
```

In [89]:

```
1 sb.set(style="white")
2 sb.set(style="whitegrid",color_codes=True)
3 import warnings
4 warnings.simplefilter(action='ignore')
```

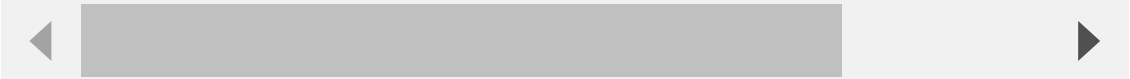
In [90]:

```
1 train_df=pd.read_csv(r"C:\Users\magam\Downloads\train.gender_submission.csv")
2 train_df
```

Out[90]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket		
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8
...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7

891 rows × 12 columns



In [91]:

```
1 train_df.head()
```

Out[91]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25
1	2	1	1Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.28
2	3	1	3Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.92
3	4	1	1Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.10
4	5	0	3Allen, Mr. William Henry	male	35.0	0	0	373450	8.05



In [92]:

```
1 train_df.shape
```

Out[92]:

(891, 12)

In [93]:

```
1 test_df=pd.read_csv(r"C:\Users\magam\Downloads\train.gender_submission.csv")
```

In [94]:

```
1 test_df.head()
```

Out[94]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25
1	2	1	1Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.28
2	3	1	3Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.92
3	4	1	1Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.10
4	5	0	3Allen, Mr. William Henry	male	35.0	0	0	373450	8.05



In [95]:

```
1 test_df.shape
```

Out[95]:

(891, 12)

In [96]:

```
1 train_df.describe()
```

Out[96]:

	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	32.204200
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693400
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	7.910000
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	512.329000



In [97]:

```
1 train_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  -
0   PassengerId  891 non-null    int64
1   Survived     891 non-null    int64
2   Pclass       891 non-null    int64
3   Name         891 non-null    object
4   Sex          891 non-null    object
5   Age          714 non-null    float64
6   SibSp        891 non-null    int64
7   Parch        891 non-null    int64
8   Ticket       891 non-null    object
9   Fare         891 non-null    float64
10  Cabin        204 non-null    object
11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

In [98]:

```
1 test_df.describe()
```

Out[98]:

	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	32.204200
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693400
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	7.910000
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	512.329000



In [99]:

```
1 test_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  -
0   PassengerId  891 non-null    int64
1   Survived     891 non-null    int64
2   Pclass       891 non-null    int64
3   Name         891 non-null    object
4   Sex          891 non-null    object
5   Age          714 non-null    float64
6   SibSp        891 non-null    int64
7   Parch        891 non-null    int64
8   Ticket       891 non-null    object
9   Fare         891 non-null    float64
10  Cabin        204 non-null    object
11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

In [100]:

```
1 #to find missing values
2 train_df.isnull().sum()
```

Out[100]:

```
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 [101]:

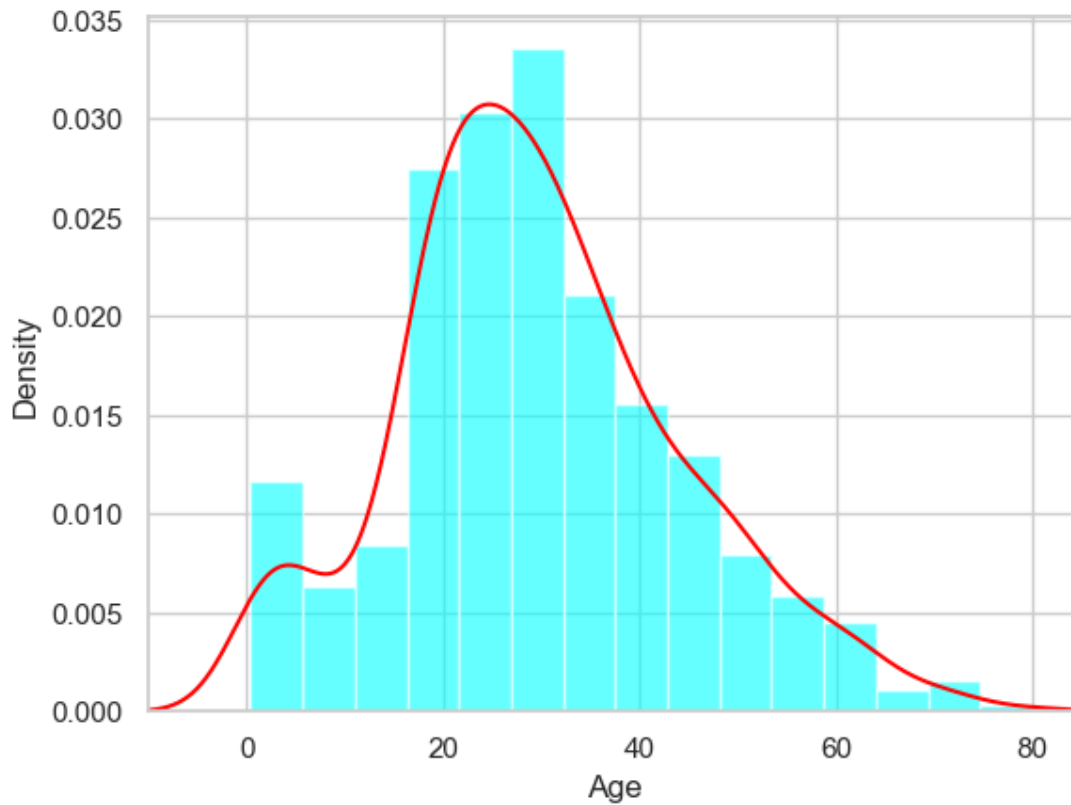
```
1 test_df.isnull().sum()
```

Out[101]:

```
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 [102]:

```
1 ax=train_df["Age"].hist(bins=15,density=True,stacked=True,color='cyan',alpha=0.6)
2 train_df["Age"].plot(kind='density',color='red')
3 ax.set(xlabel='Age')
4 plt.xlim(-10,85)
5 plt.show()
```



In [103]:

```
1 print(train_df["Age"].mean(skipna=True))
2 print(train_df["Age"].median(skipna=True))
```

29.69911764705882

28.0

In [104]:

```
1 print((train_df['Cabin'].isnull().sum()/train_df.shape[0]*100))
2 print((train_df['Embarked'].isnull().sum()/train_df.shape[0]*100))
```

77.10437710437711

0.22446689113355783



In [105]:

```
1 print('Boarded passangers grouped by port of embarkation(C=cherbourg,Q=Queensten  
2 print(train_df['Embarked'].value_counts())  
3 sb.countplot(x='Embarked',data=train_df,palette='Set2')  
4 plt.show()
```

Boarded passangers grouped by port of embarkation(C=cherbourg,Q=Queensten,S=Southampton):

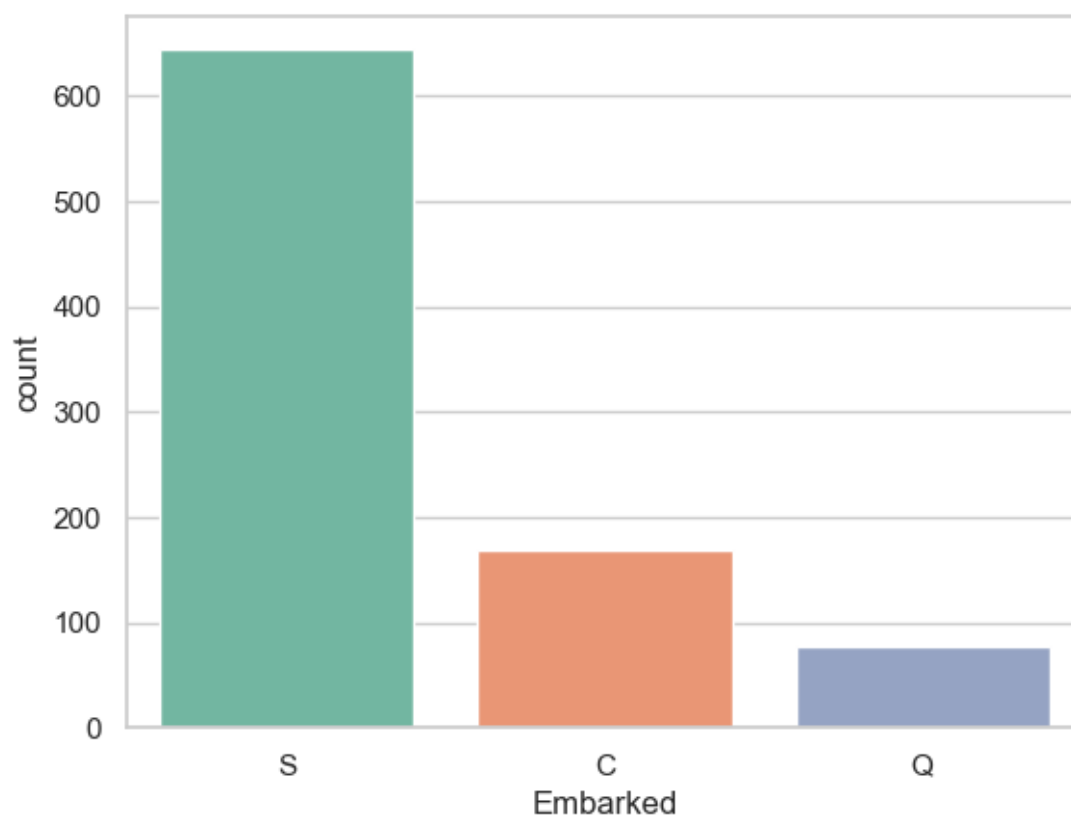
Embarked

S 644

C 168

Q 77

Name: count, dtype: int64



In [106]:

```
1 print(train_df['Embarked'].value_counts().idxmax())
```

S

In [107]:

```
1 train_data=train_df.copy()
2 train_data['Age'].fillna(train_df['Age'].median(skipna=True),inplace=True)
3 train_data["Embarked"].fillna(train_df['Embarked'].value_counts().idxmax(),inplace=True)
4 train_data.drop('Cabin',axis=1,inplace=True)
5 train_data.isnull().sum()
```

Out[107]:

PassengerId 0
Survived 0
Pclass 0
Name 0
Sex 0
Age 0
SibSp 0
Parch 0
Ticket 0
Fare 0
Embarked 0
dtype: int64

In [108]:

```
1 train_data.head()
```

Out[108]:

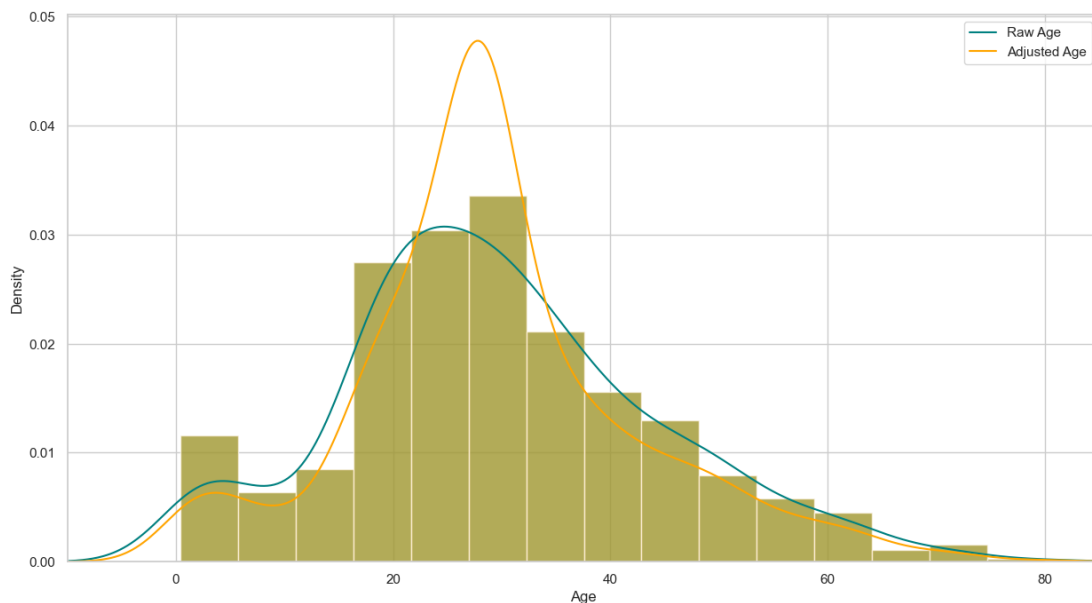
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...)	female	38.0	1	0	PC 17599	71.28
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.92
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.10
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.05

In [109]:

```

1 plt.figure(figsize=(15,8))
2 ax=train_df['Age'].hist(bins=15,density=True,stacked=True,color='teal',alpha=0.6)
3 train_df['Age'].plot(kind='density',color='teal')
4 train_df["Age"].hist(bins=15,density=True,stacked=True,color='orange',alpha=0.5)
5 train_data["Age"].plot(kind='density',color='orange')
6 ax.legend(['Raw Age','Adjusted Age'])
7 ax.set(xlabel='Age')
8 plt.xlim(-10,85)
9 plt.show()

```



In [110]:

```

1 #catagorical variables travelling alone
2 train_data['Travel Alone']=np.where((train_data['SibSp']+train_data['Parch'])>0,0
3 train_data.drop('SibSp',axis=1,inplace=True)
4 train_data.drop('Parch',axis=1,inplace=True)

```

In [111]:

```

1 #create categorical variables and drop some variables
2 training=pd.get_dummies(train_data,columns=["Pclass","Embarked","Sex"])
3 training.drop('Sex_female',axis=1,inplace=True)
4 training.drop('PassengerId',axis=1,inplace=True)
5 training.drop('Name',axis=1,inplace=True)
6 training.drop('Ticket',axis=1,inplace=True)

```

In [112]:

```
1 final_train=training
2 final_train.head()
```

Out[112]:

	Survived	Age	Fare	Travel Alone	Pclass_1	Pclass_2	Pclass_3	Embarked_C	Embarked
0	0	22.0	7.2500	0	False	False	True	False	Fa
1	1	38.0	71.2833	0	True	False	False	True	Fa
2	1	26.0	7.9250	1	False	False	True	False	Fa
3	1	35.0	53.1000	0	True	False	False	False	Fa
4	0	35.0	8.0500	1	False	False	True	False	Fa



In [113]:

```
1 test_df.isnull().sum()
```

Out[113]:

```
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 [114]:

```
1 test_data = test_df.copy()
2 test_data["Age"].fillna(train_df["Age"].median(skipna=True), inplace=True)
3 test_data["Fare"].fillna(train_df["Fare"].median(skipna=True), inplace=True)
4 test_data.drop('Cabin', axis=1, inplace=True)
5 test_data['TravelAlone']=np.where((test_data["SibSp"]+test_data["Parch"])>0, 0, 1)
6 test_data.drop('SibSp', axis=1, inplace=True)
7 test_data.drop('Parch', axis=1, inplace=True)
8 testing = pd.get_dummies(test_data, columns=["Pclass","Embarked","Sex"])
9 testing.drop('Sex_female', axis=1, inplace=True)
10 testing.drop('PassengerId', axis=1, inplace=True)
11 testing.drop('Name', axis=1, inplace=True)
12 testing.drop('Ticket', axis=1, inplace=True)
13 final_test=testing
14 final_test.head()
```

Out[114]:

	Survived	Age	Fare	TravelAlone	Pclass_1	Pclass_2	Pclass_3	Embarked_C	Emb
0	0	22.0	7.2500	0	False	False	True	False	
1	1	38.0	71.2833	0	True	False	False	True	
2	1	26.0	7.9250	1	False	False	True	False	
3	1	35.0	53.1000	0	True	False	False	False	
4	0	35.0	8.0500	1	False	False	True	False	

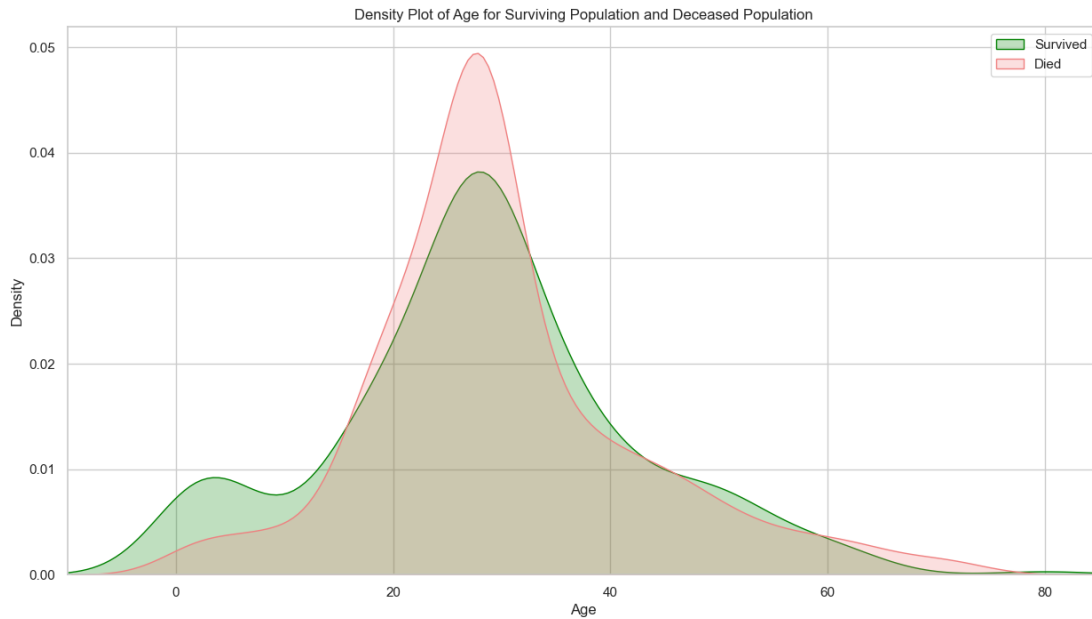


In [115]:

```

1 plt.figure(figsize=(15,8))
2 ax = sb.kdeplot(final_test["Age"][final_test.Survived == 1], color="green", shade=True)
3 sb.kdeplot(final_test["Age"][final_test.Survived == 0], color="lightcoral", shade=True)
4 plt.legend(['Survived', 'Died'])
5 plt.title('Density Plot of Age for Surviving Population and Deceased Population')
6 ax.set(xlabel='Age')
7 plt.xlim(-10,85)
8 plt.show()

```

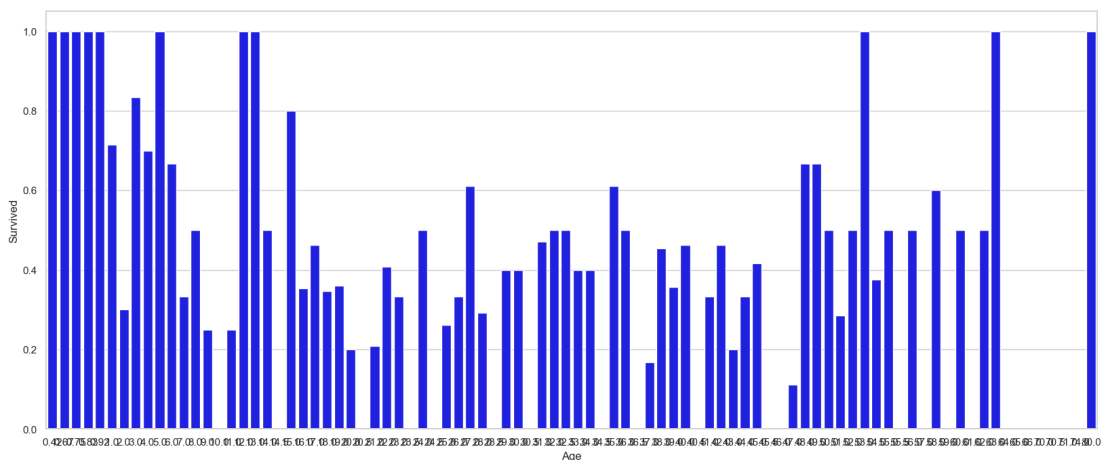


In [116]:

```

1 plt.figure(figsize=(20,8))
2 avg_survival_byage=final_train[["Age","Survived"]].groupby(['Age'], as_index=False).mean()
3 g = sb.barplot(x='Age',y='Survived', data=avg_survival_byage, color="blue")
4 plt.show()

```



In [117]:

```
1 final_train['IsMinor']=np.where(final_train['Age']<=16,1,0)
2 print(final_train['IsMinor'])
3
```

```
0      0
1      0
2      0
3      0
4      0
..
886    0
887    0
888    0
889    0
890    0
Name: IsMinor, Length: 891, dtype: int32
```

In [118]:

```
1 final_test['IsMinor']=np.where(final_test['Age']<=16,1,0)
2 print(final_test['IsMinor'])
```

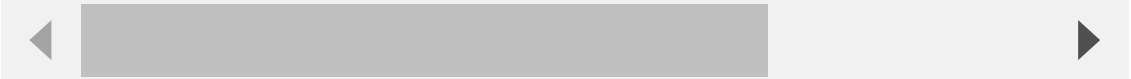
```
0      0
1      0
2      0
3      0
4      0
..
886    0
887    0
888    0
889    0
890    0
Name: IsMinor, Length: 891, dtype: int32
```

In [122]:

```
1 final_train.head()
```

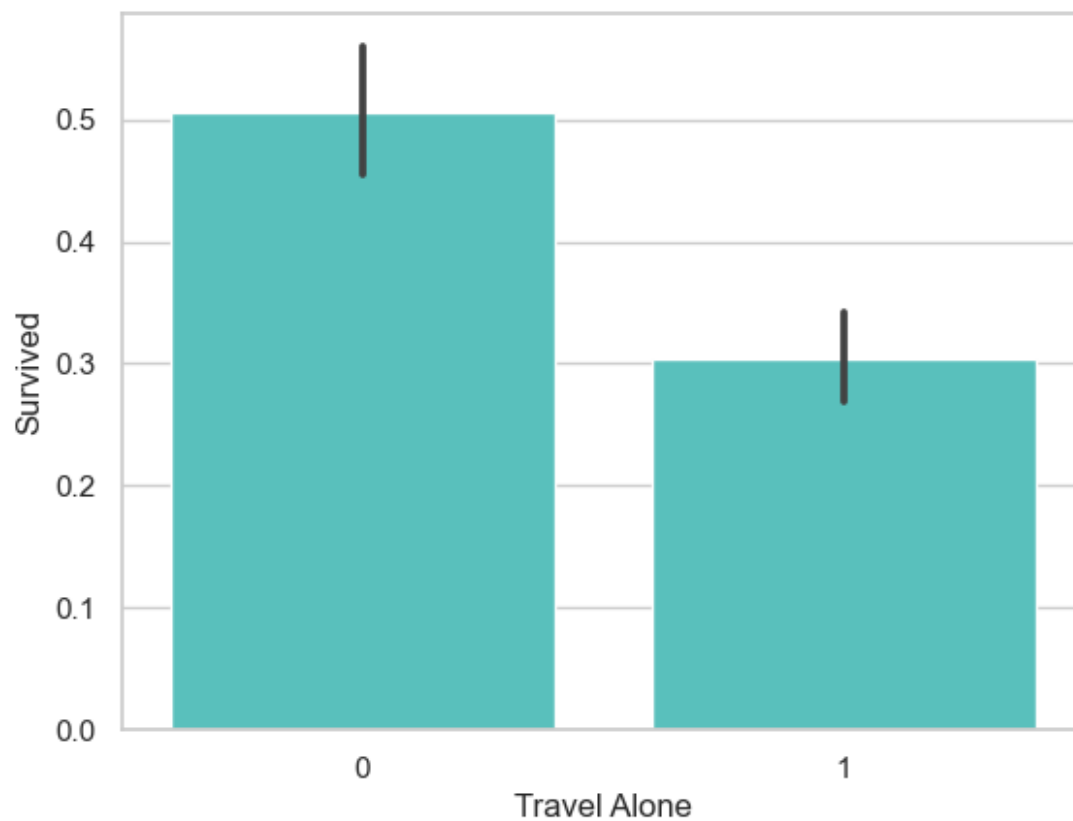
Out[122]:

	Survived	Age	Fare	Travel Alone	Pclass_1	Pclass_2	Pclass_3	Embarked_C	Embarked
0	0	22.0	7.2500	0	False	False	True	False	Fa
1	1	38.0	71.2833	0	True	False	False	True	Fa
2	1	26.0	7.9250	1	False	False	True	False	Fa
3	1	35.0	53.1000	0	True	False	False	False	Fa
4	0	35.0	8.0500	1	False	False	True	False	Fa



In [124]:

```
1 sb.barplot(x='Travel Alone', y='Survived', data=final_train, color="mediumturquoise")
2 plt.show()
```



In [ ]:

1