

In [9]:

```
import numpy as np
import pandas as pd
from sklearn import preprocessing
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style="white")
sns.set(style="whitegrid", color_codes=True)
import warnings
warnings.simplefilter(action='ignore')
```

In [13]:

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

Out[13]:

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

891 rows × 12 columns



In [14]:

```
train_df.head()
```

Out[14]:

	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

In [15]:

```
train_df.shape
```

Out[15]:

(891, 12)

In [16]:

```
train_df.describe()
```

Out[16]:

	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

In [17]:

```
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 [18]:

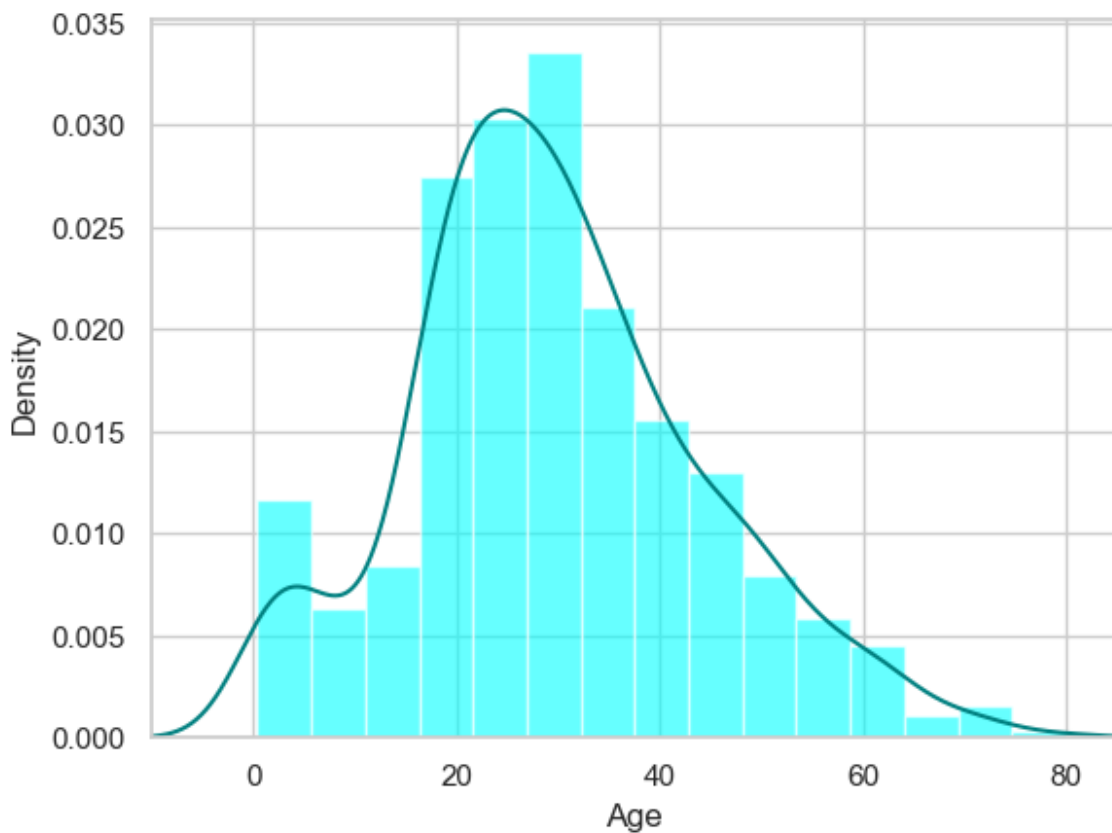
```
train_df.isnull().sum()
```

Out[18]:

```
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 [19]:

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



In [20]:

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

```
29.69911764705882
28.0
```

In [21]:

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

```
77.10437710437711
```

In [22]:

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

```
0.22446689113355783
```

In [23]:

```
print('Boarded passengers group by part of Embarkation(C=Cherburg,Q=Queenstown,S=Southam  
print(train_df['Embarked'].value_counts())  
sns.countplot(x='Embarked',data=train_df,palette='Set2')  
plt.show()
```

Boarded passengers group by part of Embarkation(C=Cherburg,Q=Queenstown,S=Southampton):

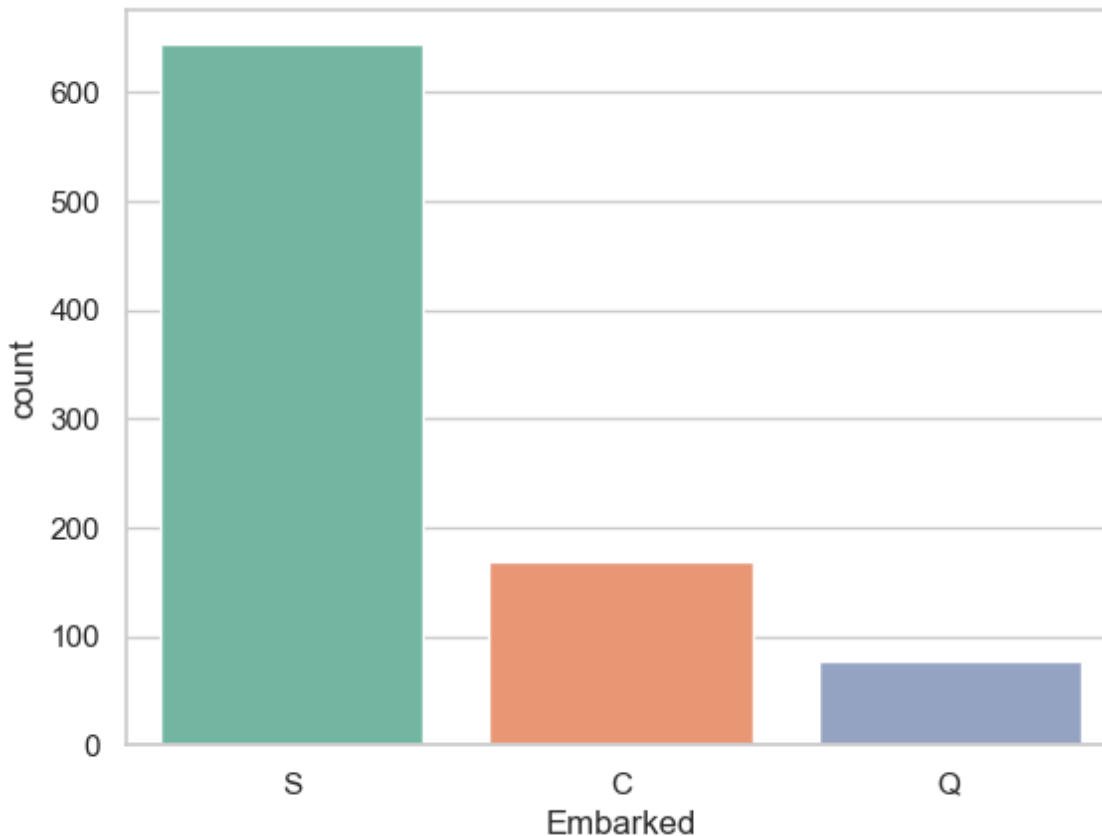
Embarked

S 644

C 168

Q 77

Name: count, dtype: int64



In [24]:

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

S

In [25]:

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

In [26]:

```
train_data.isnull().sum()
```

Out[26]:

```
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 [27]:

```
train_data.head()
```

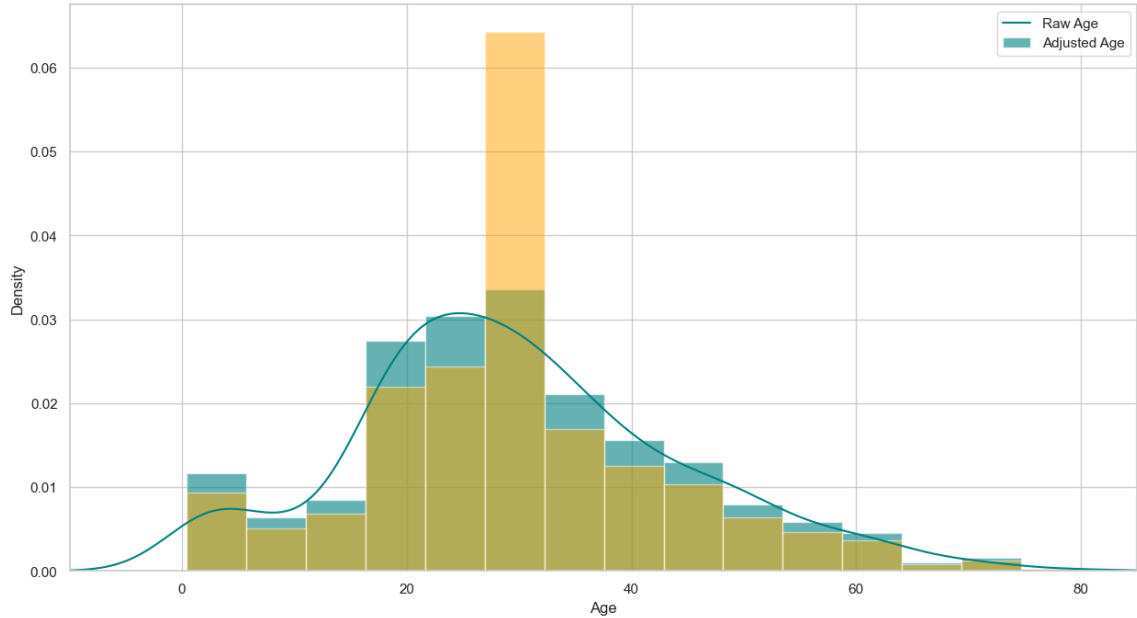
Out[27]:

	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



In [28]:

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



In [29]:

```
train_data['TravelAlone']=np.where((train_data["SibSp"]+train_data["Parch"])>0,0,1)
train_data.drop('SibSp',axis=1,inplace=True)
train_data.drop('Parch',axis=1,inplace=True)
```

In [30]:

```
training=pd.get_dummies(train_data,columns=["Pclass","Embarked","Sex"])
training.drop('Sex_female',axis=1,inplace=True)
training.drop('PassengerId',axis=1,inplace=True)
training.drop('Name',axis=1,inplace=True)
training.drop('Ticket',axis=1,inplace=True)
final_train=training
final_train.head()
```

Out[30]:

	Survived	Age	Fare	TravelAlone	Pclass_1	Pclass_2	Pclass_3	Embarked_C	Embark
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 [37]:

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

Out[37]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cat
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	N
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	N
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	N
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	N
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	N
...
413	1305	3	Spector, Mr. Woolf	male	NaN	0	0	A.5. 3236	8.0500	N
414	1306	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	108.9000	C1
415	1307	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	7.2500	N
416	1308	3	Ware, Mr. Frederick	male	NaN	0	0	359309	8.0500	N
417	1309	3	Peter, Master. Michael J	male	NaN	1	1	2668	22.3583	N

418 rows × 11 columns



In [38]:

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

Out[38]:

```

PassengerId      0
Pclass           0
Name             0
Sex              0
Age             86
SibSp            0
Parch            0
Ticket           0
Fare             1
Cabin           327
Embarked         0
dtype: int64

```

In [39]:

```

test_data=test_df.copy()
test_data["Age"].fillna(test_df["Age"].median(skipna=True),inplace=True)
test_data["Embarked"].fillna(test_df["Embarked"].value_counts().idxmax(),inplace=True)
test_data.drop('Cabin',axis=1,inplace=True)
test_data['TravelAlone']=np.where((test_data["SibSp"]+test_data["Parch"])>0,0,1)
test_data.drop('SibSp',axis=1,inplace=True)
test_data.drop('Parch',axis=1,inplace=True)
testing=pd.get_dummies(test_data,columns=["Pclass","Embarked","Sex"])
testing.drop('Sex_female',axis=1,inplace=True)
testing.drop('PassengerId',axis=1,inplace=True)
testing.drop('Name',axis=1,inplace=True)
testing.drop('Ticket',axis=1,inplace=True)
final_test=testing
final_test.head()

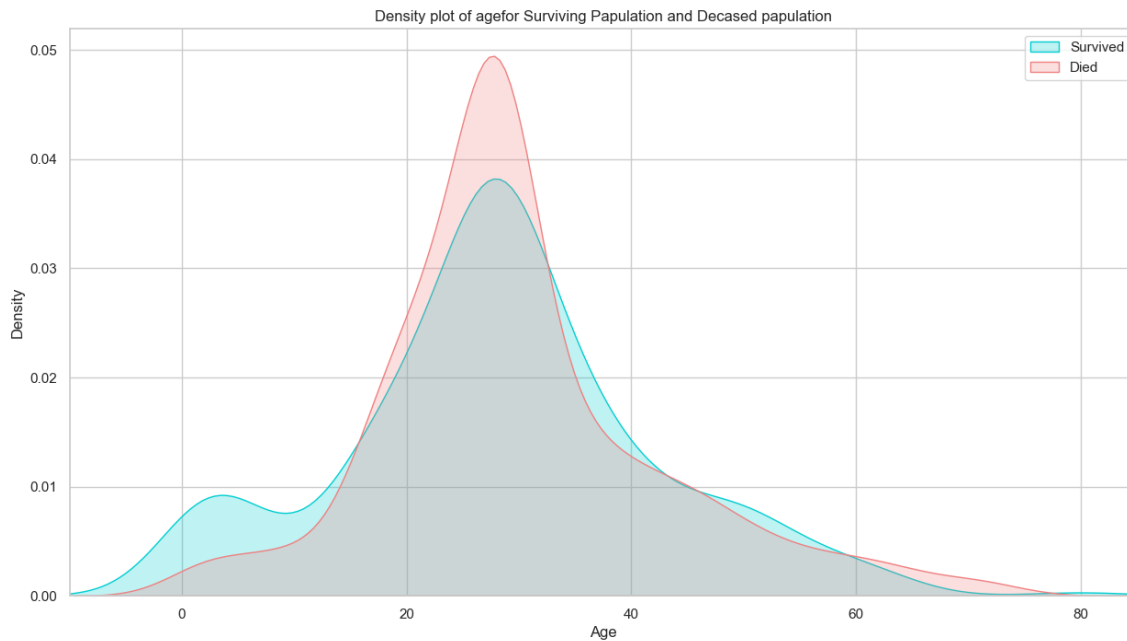
```

Out[39]:

	Age	Fare	TravelAlone	Pclass_1	Pclass_2	Pclass_3	Embarked_C	Embarked_Q	Em
0	34.5	7.8292	1	False	False	True	False	True	
1	47.0	7.0000	0	False	False	True	False	False	
2	62.0	9.6875	1	False	True	False	False	True	
3	27.0	8.6625	1	False	False	True	False	False	
4	22.0	12.2875	0	False	False	True	False	False	

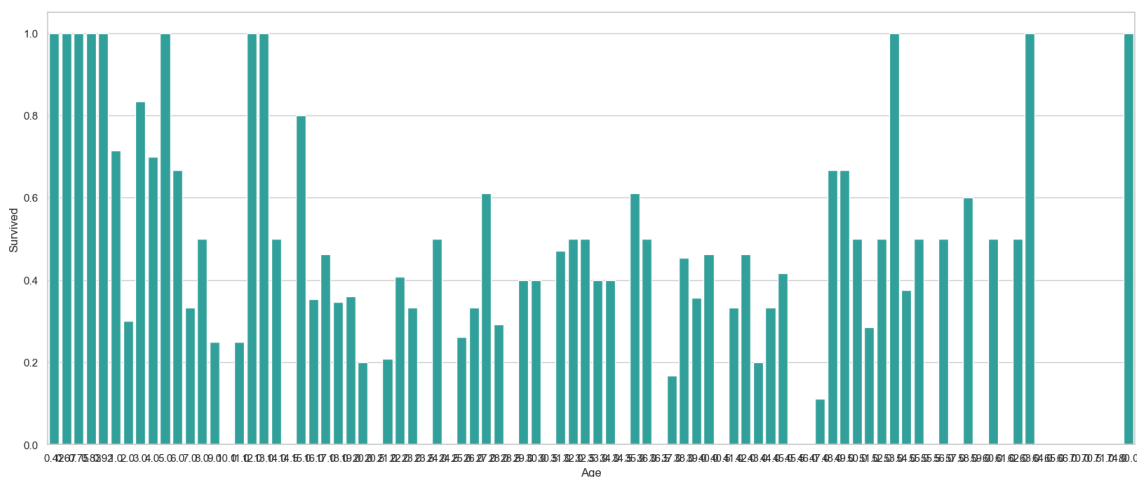
In [41]:

```
plt.figure(figsize=(15,8))
ax=sns.kdeplot(final_train["Age"][final_train.Survived==1],color="darkturquoise",shade=True)
sns.kdeplot(final_train["Age"][final_train.Survived==0],color="lightcoral",shade=True)
plt.legend(['Survived','Died'])
plt.title('Density plot of agefor Surviving Papulation and Decased papulation')
ax.set(xlabel='Age')
plt.xlim(-10,85)
plt.show()
```



In [42]:

```
plt.figure(figsize=(20,8))
avg_Survival_byage=final_train[["Age", "Survived"]].groupby(['Age'],as_index=False).mean()
g=sns.barplot(x='Age',y='Survived',data=avg_Survival_byage,color="lightSeaGreen")
plt.show()
```



In [45]:

```
final_train['IsMinor']=np.where(final_train['Age']<=16,1,0)
print(final_train['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 [46]:

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

```
0      0
1      0
2      0
3      0
4      0
```

..

```
413    0
414    0
415    0
416    0
417    0
```

Name: IsMinor, Length: 418, dtype: int32

In []: