```
#EXPLORATORY DATA ANALYSIS - MACHINE LEARNING FROM DISASTER - TITANIC
```

```
#1.create dataframe
import pandas as pd
df = pd.read_csv('/content/train.csv')
df
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	C
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

```
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
```

```
df.shape #891 rows and 12 columns
```

```
(891, 12)
```

```
# total number of elements in dataframe
df.size
```

```
10692
```

```
# To dfficially check the null values
df.isnull().sum()
```

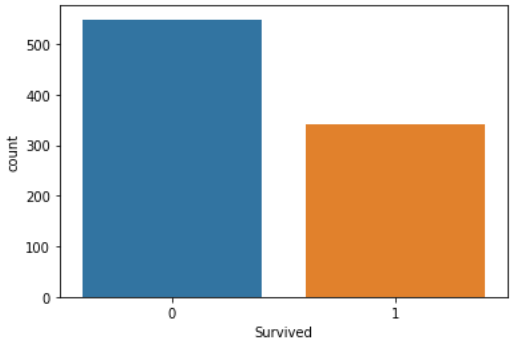
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

```
# I want to find out the exact count elements in each and every column
df.nunique()
```

```
PassengerId    891
Survived         2
Pclass           3
Name           891
Sex              2
Age             88
SibSp            7
Parch            7
Ticket         681
Fare           248
Cabin          147
Embarked         3
dtype: int64
```

```
# Visualisation tells us about the no of people Survived and Not Survived
#visualization - SEABORN
import seaborn as sns
sns.countplot(x = 'Survived',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b47d8db80>
```




```
#I want to know the exact count of the number of people Survived and Not Survived
df.groupby('Survived').size()
```

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

```
#an other way to find
df['Survived'].value_counts()
```

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

```
df.Survived.value_counts()
```

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

```
#I want to know how many males and how many females were on Titanic
df.Sex.value_counts()
```

```
male      577
female    314
Name: Sex, dtype: int64
```

```
df['Sex'].value_counts()
```

```
male      577
female    314
Name: Sex, dtype: int64
```

```
df.groupby('Sex').size()
```

```
Sex
female    314
male      577
dtype: int64
```

```
#I want the exact count
#No of males - Survived
#No of females - Survived
#No of males - Not Survived
#No of females - Not Survived

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

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

```
df[['Sex','Survived']].value_counts()
```

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

```
import numpy as np
survived_m = np.sum((df['Sex']=='male')&df['Survived']==1)
survived_m
```

```
109
```

```
#Let us divide the Age column into 4 categories
young = np.sum((df['Age']>=0)&(df['Age']<20))
adult = np.sum((df['Age']>=20)&(df['Age']<40))
midage = np.sum((df['Age']>=40)&(df['Age']<60))
old = np.sum((df['Age']>=60))
print(young)
print(adult)
print(midage)
print(old)
```

```
164
387
137
26
```

```
#The age column has got 714 values
164+387+137+26
```

```
714
```

```
#I want to find out the age of the youngest passenger
np.min(df['Age']) #here 0.42 years = 5 months
```

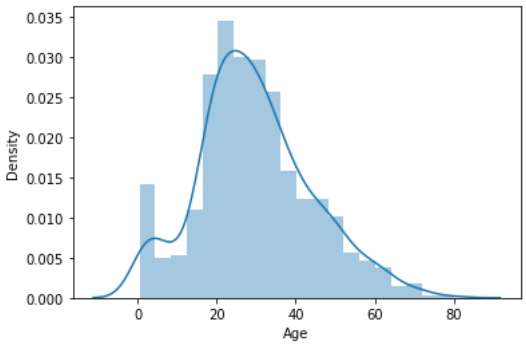
```
0.42
```

```
#I want to find out the age of the eldest passenger
np.max(df['Age']) #80 years
```

```
80.0
```

```
#DISTRIBUTION PLOT
sns.distplot(df['Age'])
```

```
/usr/local/lib/python3.8/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot`
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b4787e9a0>
```



✓ 1s completed at 7:24 PM