

## ✓ TASK-1 CODSOFT(DATA SCIENCE)

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df=pd.read_csv("/content/drive/MyDrive/codsodt1/Titanic-Dataset.csv")
df.head(20)
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46	S
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750	NaN	S
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333	NaN	S
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708	NaN	C
10	11	1	3	Sandstrom, Miss. Marguerite Rut	female	4.0	1	1	PP 9549	16.7000	G6	S
11	12	1	1	Bonnell, Miss. Elizabeth	female	58.0	0	0	113783	26.5500	C103	S
12	13	0	3	Saunders, Mr. William Henry	male	20.0	0	0	A/5. 2151	8.0500	NaN	S
13	14	0	3	Andersson, Mr. Anders Johan	male	39.0	1	5	347082	31.2750	NaN	S
14	15	0	3	Vestrom, Miss. Hulda Amanda Adolfina	female	14.0	0	0	350406	7.8542	NaN	S
15	16	1	2	Hewlett, Mrs. (Mary D Kingcome)	female	55.0	0	0	248706	16.0000	NaN	S

```
df.shape
```

```
(891, 12)
```

```
df.describe()
```

	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

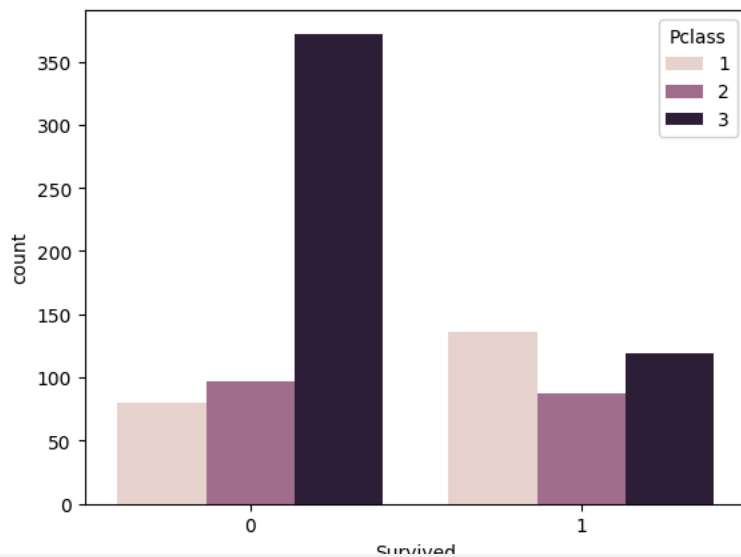
✓ from above cell it is clear there are few missing values in age column

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

count	
Survived	
0	549
1	342

```
#let's visualize the count of survivals wrt pclass  
sns.countplot(x=df['Survived'],hue=df['Pclass'])
```

<Axes: xlabel='Survived', ylabel='count'>



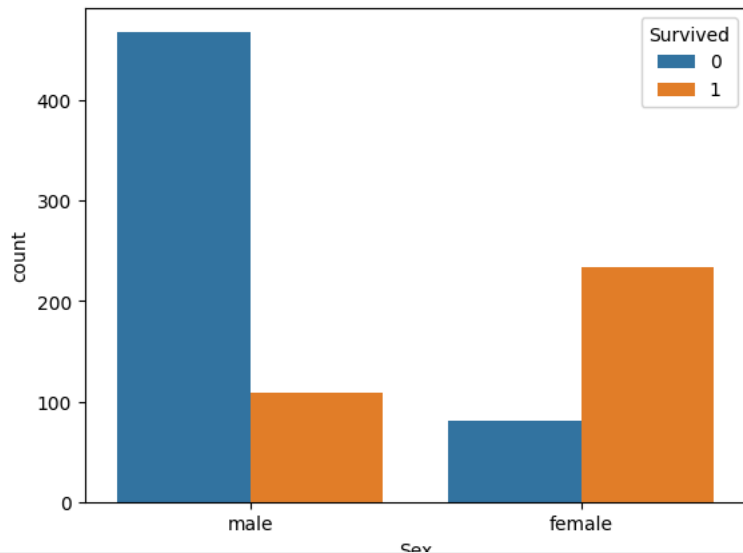
```
df['Sex']
```

Sex	
0	male
1	female
2	female
3	female
4	male
...	...
886	male
887	female
888	female
889	male
890	male

891 rows × 1 columns

```
#lets visulaize the count of survivals wrt gender  
sns.countplot(x=df['Sex'],hue=df['Survived'])
```

<Axes: xlabel='Sex', ylabel='count'>



```
#look at survival rate by sex
df.groupby(['Sex'])['Survived'].mean()
```

```
Survived
Sex
female  0.742038
male    0.188908
```

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

```
array(['male', 'female'], dtype=object)
```

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
df['Sex']=le.fit_transform(df['Sex'])
df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	1	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...)	0	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	0	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S

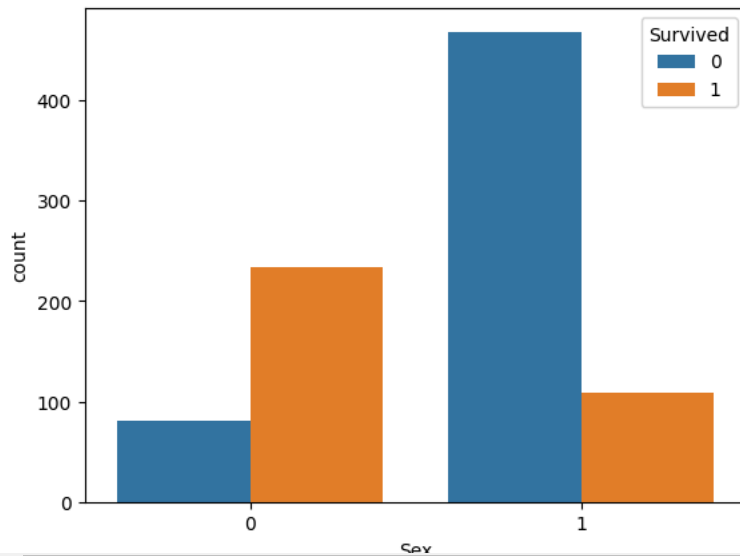
```
df['Sex'],df['Survived']
```

```
(0    1
 1    0
 2    0
 3    0
 4    1
 ..
886   1
887   0
888   0
889   1
890   1
Name: Sex, Length: 891, dtype: int64,
0    0
1    1
2    1
3    1
4    0
..
886   0
887   1
888   0
889   1)
```

```
890 0
      Name: Survived, Length: 891, dtype: int64)
```

```
#graph is also similar after transform data
sns.countplot(x=df['Sex'],hue=df['Survived'])
```

```
<Axes: xlabel='Sex', ylabel='count'>
```




```
df.isna().sum()
```

```
0
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
```

```
#after dropping non required column
df=df.drop(['Age'],axis=1)
```

```
df_final=df
df_final.head(20)
```

	PassengerId	Survived	Pclass	Name	Sex	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	1	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	1	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	1	0	0	330877	8.4583	NaN	Q
6	7	0	1	McCarthy, Mr. Timothy J	1	0	0	17463	51.8625	E46	S
7	8	0	3	Palsson, Master. Gosta Leonard	1	3	1	349909	21.0750	NaN	S

MODEL TRAINING

```
x=df[['Pclass','Sex']]
y=df['Survived']

12      13      0      3      Saundercok. Mr. William Henry      1      0      0      A/5. 2151      8.0500      NaN      S
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=40)

14      15      0      3      Veselovi, Miss. Anna Anna Anna      0      0      0      330400      7.0342      NaN      S

from sklearn.linear_model import LogisticRegression
log=LogisticRegression(random_state=0)
log.fit(x_train,y_train)
```

LogisticRegression

LogisticRegression(random\_state=0)

MODEL PREDICTION

```
pred=print(log.predict(x_test))

[1 0 1 1 1 0 0 0 0 0 1 0 0 0 1 0 1 1 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 0
 0 1 1 0 0 0 1 1 0 0 0 0 0 0 1 0 1 1 0 1 0 1 1 0 0 0 0 1 0 0 1 1 0 1 1 1 0 0
 0 0 0 1 0 1 1 1 1 0 1 1 0 1 1 1 0 1 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0
 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 0 0 0 1 1 1 1 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0
 1 0 0 0 1 1 0 0 1 1 1 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 1]

print(y_test)

246      0
588      0
472      1
71       0
654      0
..
284      0
727      1
599      1
525      0
457      1
Name: Survived, Length: 179, dtype: int64

import warnings
warnings.filterwarnings('ignore')
res=log.predict([[2,1]])#there first is class pasanger and
#second 1 is for male also use 0 for female
if (res==0):
    print("so sorry not survived")
else:
    print("survived")

so sorry not survived
```