In [1]:

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

%matplotlib inline

import warnings
warnings.filterwarnings('ignore')
```

In [2]:

```
titanic=pd.read_csv(r"D:\Full Stack Data Science\30 Aug\Titanic dataset ar
titanic.head(3)
```

Out[2]:

	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	I
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	I

In [3]:

```
titanic.describe()
```

Out[3]:

	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 [4]:
```
# Name,Ticket,Fare,Cabin these columns cannot decide survival of person th
del titanic['Name']
del titanic['Ticket']
del titanic['Fare']
del titanic['Cabin']
```

In [5]:
```
titanic.head()
```

Out[5]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Embarked
0	1	0	3	male	22.0	1	0	S
1	2	1	1	female	38.0	1	0	C
2	3	1	3	female	26.0	0	0	S
3	4	1	1	female	35.0	1	0	S
4	5	0	3	male	35.0	0	0	S

In [6]:
```
# Changing value for male ,female string values to numeric values male=1 
titanic['Sex']=titanic.Sex.replace({'male':1,'female':2})
```

In [7]:
```
titanic.head()
```

Out[7]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Embarked
0	1	0	3	1	22.0	1	0	S
1	2	1	1	2	38.0	1	0	C
2	3	1	3	2	26.0	0	0	S
3	4	1	1	2	35.0	1	0	S
4	5	0	3	1	35.0	0	0	S

In [8]:
```
titanic.isnull().sum()
```

Out[8]:
```
PassengerId      0
Survived         0
Pclass           0
Sex              0
Age            177
SibSp            0
Parch            0
Embarked         2
dtype: int64
```

Fill the null values of the Age column. Fill mean Survived age(mean age of the survived people) in the column where the person has survived and mean not Survived age (mean age of the people who have not survived) in the column where person has not survived

In [9]:
```
mean_S=titanic[titanic.Survived==1].Age.mean()
mean_S
```

Out[9]:
```
28.343689655172415
```

Creating a new "Age" column , filling values in it with a condition if goes True then given values (here meanS) is put in place of last values else nothing happens, simply the values are copied from the "Age" column of the dataset

In [10]:
```
e']=np.where(pd.isnull(titanic.Age) & titanic.Survived==1,mean_S,titanic.Age)
d()
```

Out[10]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Embarked	age
0	1	0	3	1	22.0	1	0	S	22.0
1	2	1	1	2	38.0	1	0	C	38.0
2	3	1	3	2	26.0	0	0	S	26.0
3	4	1	1	2	35.0	1	0	S	35.0
4	5	0	3	1	35.0	0	0	S	35.0

In [11]:
```
titanic.isnull().sum()
```

Out[11]:
```
PassengerId      0
Survived         0
Pclass           0
Sex              0
Age            177
SibSp            0
Parch            0
Embarked         2
age            125
dtype: int64
```

In [12]:
```
# Finding the mean age of not survived people
mean_NS=titanic[titanic.Survived==0].Age.mean()
mean_NS
```

Out[12]:
```
30.62617924528302
```

In [13]:
```
titanic.age.fillna(mean_NS,inplace=True)
titanic.tail()
```

Out[13]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Embarked	age
886	887	0	2	1	27.0	0	0	S	27.000000
887	888	1	1	2	19.0	0	0	S	19.000000
888	889	0	3	2	NaN	1	2	S	30.626179
889	890	1	1	1	26.0	0	0	C	26.000000
890	891	0	3	1	32.0	0	0	Q	32.000000

In [14]:
```
titanic.isnull().sum()
```

Out[14]:
```
PassengerId      0
Survived         0
Pclass           0
Sex              0
Age            177
SibSp            0
Parch            0
Embarked         2
age              0
dtype: int64
```

In [15]:
```
# Delete Age column
del titanic['Age']
titanic.head()
```

Out[15]:

	PassengerId	Survived	Pclass	Sex	SibSp	Parch	Embarked	age
0	1	0	3	1	1	0	S	22.0
1	2	1	1	2	1	0	C	38.0
2	3	1	3	2	0	0	S	26.0
3	4	1	1	2	1	0	S	35.0
4	5	0	3	1	0	0	S	35.0

We want to check if "Embarked" column is important for analysis or not, that is whether survival of the person depends on the Embarked column value or not

In [16]:
```
# Finding the number of people who have survived
# Given that they have Embarked or Boarded from a perticular part

## Persons who are survived
survivedQ=titanic[titanic.Embarked=='Q'][titanic.Survived==1].shape[0]
survivedC=titanic[titanic.Embarked=='C'][titanic.Survived==1].shape[0]
survivedS=titanic[titanic.Embarked=='S'][titanic.Survived==1].shape[0]
print(survivedQ)
print(survivedC)
print(survivedS)
```

```
30
93
217
```

In [17]:
```
## Persons who are not survived
survivedQ=titanic[titanic.Embarked=='Q'][titanic.Survived==0].shape[0]
survivedC=titanic[titanic.Embarked=='C'][titanic.Survived==0].shape[0]
survivedS=titanic[titanic.Embarked=='S'][titanic.Survived==0].shape[0]
print(survivedQ)
print(survivedC)
print(survivedS)
```

```
47
75
427
```

Interpretation

- As there are significant changes in the survival rate based on which port the passengers aboard the ship.
- We cannot delete the whole Embarked column,it is useful.

In Embarked column there is only two null values therefore we can delete it beacause they are not affect on our result.

In [18]:
```
titanic.dropna(inplace=True)
titanic.isnull().sum()
```

Out[18]:
```
PassengerId    0
Survived       0
Pclass         0
Sex            0
SibSp          0
Parch          0
Embarked       0
age            0
dtype: int64
```

In [19]:
```
# Rename 'Gender' and 'age' columns
titanic.rename(columns={'Sex':'Gender','age':'Age'},inplace=True)
```

In [20]:
```
titanic.head()
```

Out[20]:

	PassengerId	Survived	Pclass	Gender	SibSp	Parch	Embarked	Age
0	1	0	3	1	1	0	S	22.0
1	2	1	1	2	1	0	C	38.0
2	3	1	3	2	0	0	S	26.0
3	4	1	1	2	1	0	S	35.0
4	5	0	3	1	0	0	S	35.0

In [33]:
```
# Rename Embarked = Embark
titanic=titanic.rename(columns={'Embarked':'Embark','Gender':'Sex'})
```

In [34]:
```
# To convert Embarked column to numerical S=1,Q=2,C=3
titanic.Embark=titanic.Embark.replace({'S':1,'Q':2,'C':3})
```

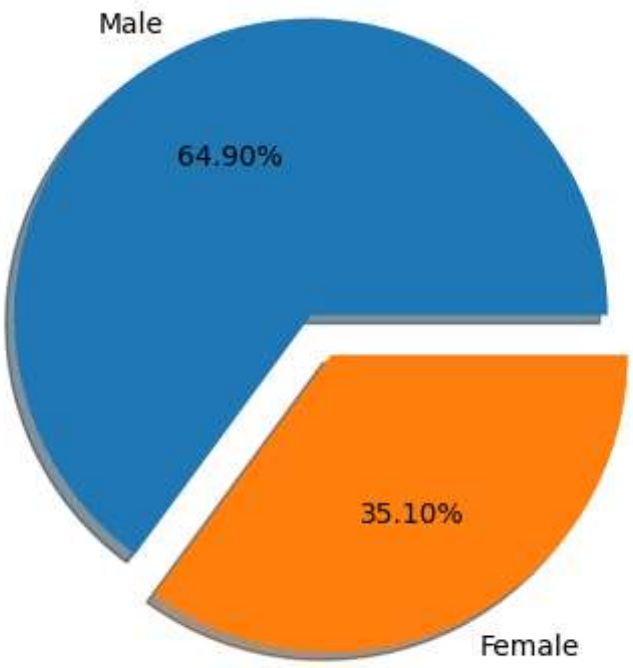
In [35]:
```
titanic.head()
```

Out[35]:

	PassengerId	Survived	Pclass	Sex	SibSp	Parch	Embark	Age
0	1	0	3	1	1	0	1	22.0
1	2	1	1	2	1	0	3	38.0
2	3	1	3	2	0	0	1	26.0
3	4	1	1	2	1	0	1	35.0
4	5	0	3	1	0	0	1	35.0

In [51]:
```
# Draw a pie chart for number of males and females

plot=plt.pie(titanic.Sex.value_counts(),labels=('Male','Female'),autopct=
```

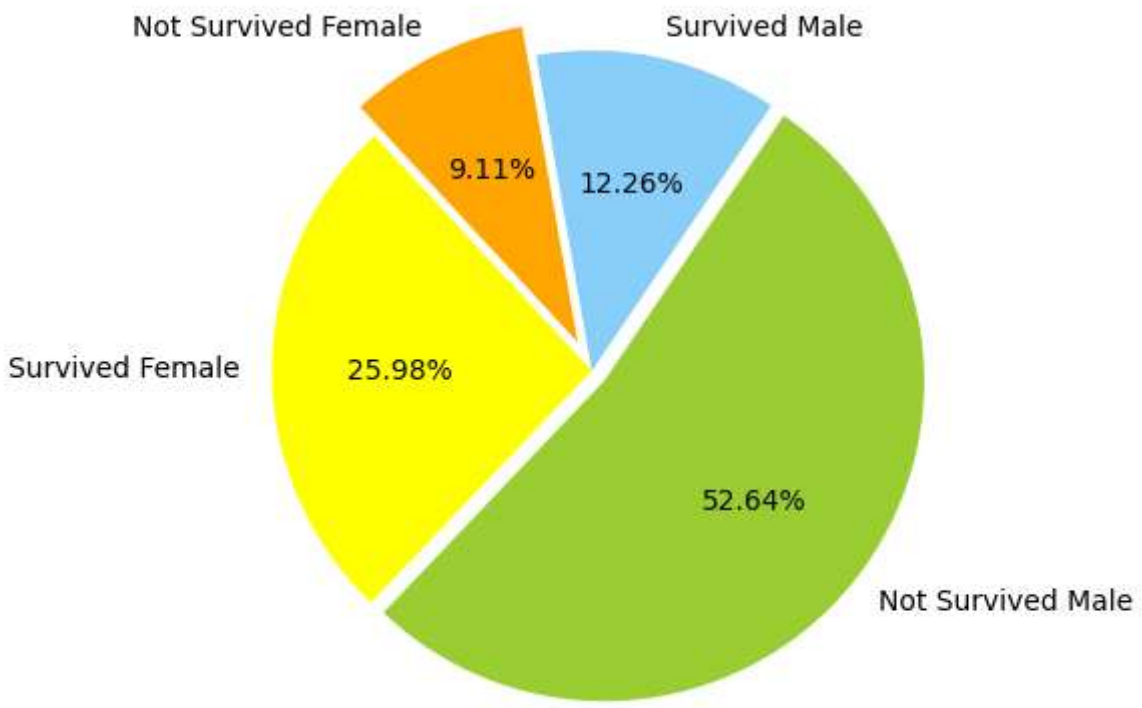


In [52]:
```
# More precise plot
```

In [91]:
```
MaleS=titanic[titanic.Sex==1][titanic.Survived==1].shape[0]
print(MaleS)
MaleNS=titanic[titanic.Sex==1][titanic.Survived==0].shape[0]
print(MaleNS)
FemaleS=titanic[titanic.Sex==2][titanic.Survived==1].shape[0]
print(FemaleS)
FemaleNS=titanic[titanic.Sex==2][titanic.Survived==0].shape[0]
print(FemaleNS)
```

```
109
468
231
81
```

In [92]:
```
t=[MaleS,MaleNS,FemaleS,FemaleNS]
ls=["Survived Male","Not Survived Male","Survived Female","Not Survived Female"
ode=[0,0.05,0,0.1]
pie(chart,labels=labels,colors=colors,explode=explode,startangle=100,counterclo
axis("equal")
show()
```



In [ ]:
```
1
```