Titanic project

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import accuracy_score
```

In [5]:

```
df = pd.read_csv("train.csv")
test = pd.read_csv("test.csv")
```

In [6]:

```
df.head(3)
```

Out[6]:

	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

In [7]:

```
df.columns
```

Out[7]:

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

In [8]:

```
df.shape
```

Out[8]:

```
(891, 12)
```

In [9]:

```
df.Survived.count()
```

Out[9]:

```
891
```

In [10]:

```
df.Survived.sum()
```

Out[10]:

```
342
```

In [11]:

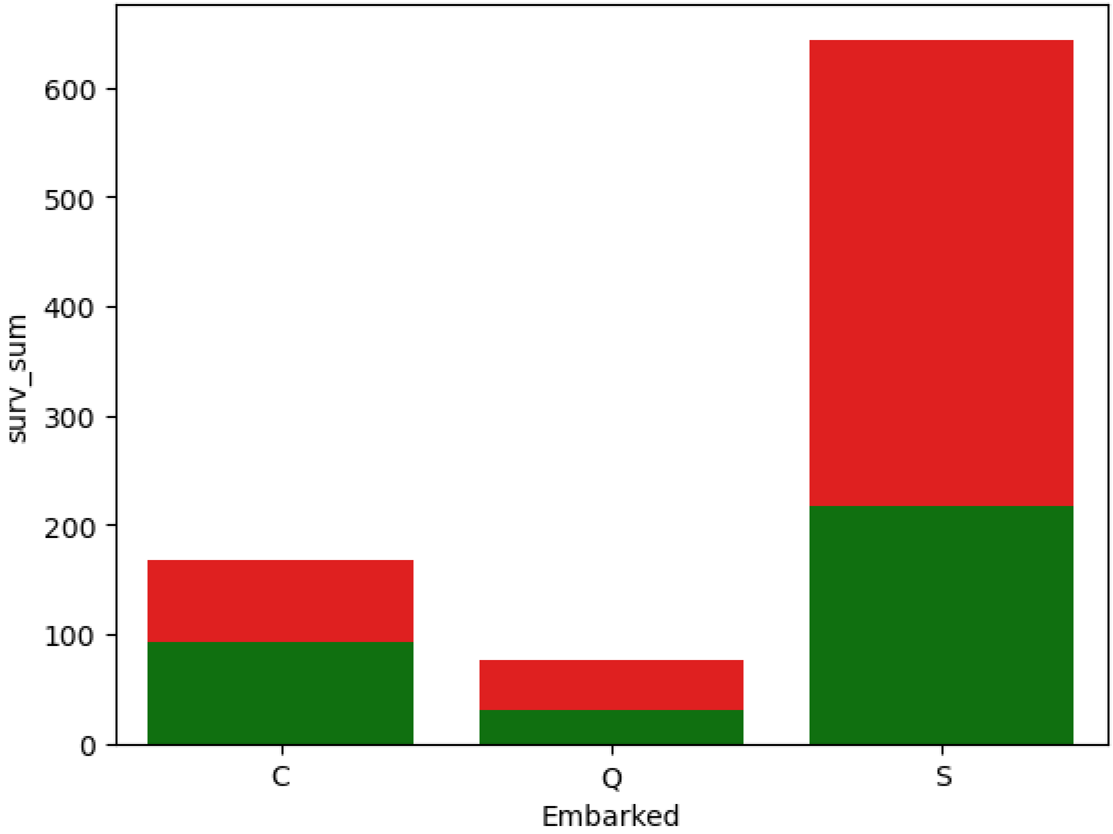
```
agg_func_bar ={'Survived':['sum',"count"]}
df_bar = df.groupby(by = "Embarked").agg(agg_func_bar)
df_bar.columns = ["surv_sum" , "surv_count"]
df_bar = df_bar.reset_index()
df_bar.head(3)
```

Out[11]:

	Embarked	surv_sum	surv_count
0	C	93	168

1	Q	30	77
2	S	217	644

In [12]:
```
bar_embarked = plt.subplots()
bar_embarked = sns.barplot(x = "Embarked" , y = 'surv_count' , data = df_bar , color = "
bar_embarked = sns.barplot( x ="Embarked" , y = 'surv_sum' , data = df_bar, color = "gre
plt.show()
```



In [13]:
```
df.head(3)
```

Out[13]:

	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

In [14]:
```
df.columns
```

Out[14]:
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

In [15]:
```
df = df.drop(columns = ["PassengerId" , "Name", "Ticket", "Embarked", "Cabin"])
```

In [16]:

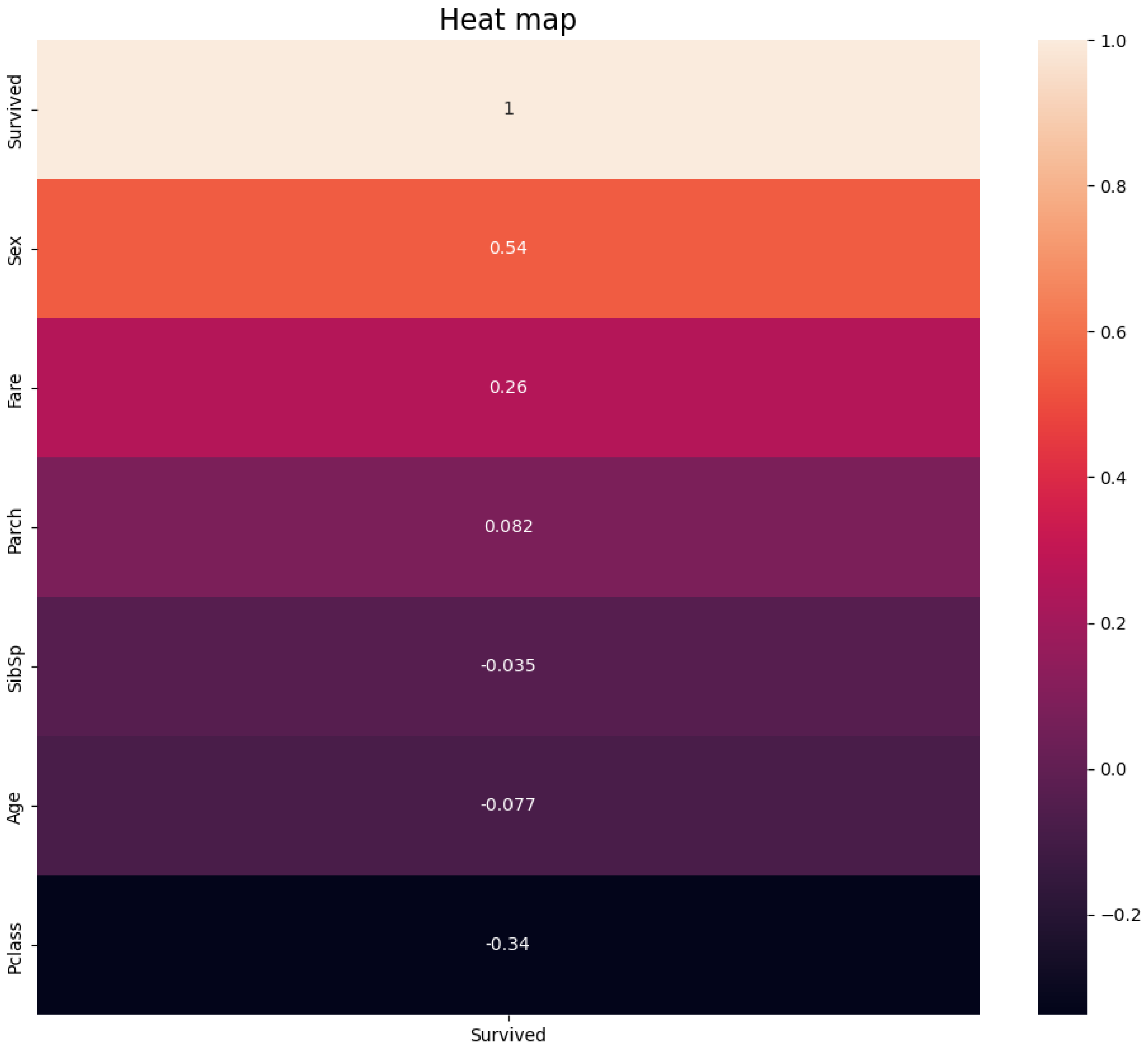
```
filter_sex = {"male": 0 , "female": 1}
df["Sex"] = df["Sex"].map(filter_sex)
df.head(3)
```

Out[16]:

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare
0	0	3	0	22.0	1	0	7.2500
1	1	1	1	38.0	1	0	71.2833
2	1	3	1	26.0	0	0	7.9250

In [17]:

```
plt.figure(figsize = (12,10))
heat_map = sns.heatmap(df.corr()[["Survived"]].sort_values(by = "Survived", ascending =
heat_map.set_title("Heat map", fontdict = {"fontsize":16})
plt.show()
```



I have high concerns about age correlation lets chech it

Data which we will use to predict survived rate from most important to less important:

1.Sex

2.Pclass

3.Fare

In [18]:

```
df.drop(columns = ["Age","SibSp","Parch"], inplace = True)
```

In [19]:

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

Out[19]:

```
Survived    0
Pclass      0
Sex         0
Fare        0
dtype: int64
```

Linear model accuracy around 79%

In [20]:

```
x_train, x_test, y_train, y_test = train_test_split(df[["Sex","Pclass","Fare"]], df[["Su
```

In [21]:

```
lin_reg = LinearRegression().fit(x_train,y_train)
y_pred = lin_reg.predict(x_test)
y_pred = np.round(y_pred)
accuracy = accuracy_score(y_test,y_pred)
print(accuracy)
```

```
0.8097014925373134
```