

In [2]:

```
#Importing our dataset from csv file
import pandas as pd
dataset=pd.read_csv("titanic-passengers-DT.csv", sep=";")

#Preprocessing our data
def preprocess_data(new_data):
    new_data['Age'].fillna(new_data['Age'].mean(),inplace=True)
    new_data.replace({'Sex':{'male': 1, 'female':0}},inplace=True)
    new_data['Cabin']=new_data.Cabin.fillna('G6')
    new_data.replace({'Survived':{'Yes': 1, 'No':0}},inplace=True)
    return new_data
data=preprocess_data(dataset)
```

In [3]:

```
data.head()
```

Out[3]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	343	0	2	Collander, Mr. Erik Gustaf	1	28.0	0	0	248740	13.0000	G6
1	76	0	3	Moen, Mr. Sigurd Hansen	1	25.0	0	0	348123	7.6500	F G73
2	641	0	3	Jensen, Mr. Hans Peder	1	20.0	0	0	350050	7.8542	G6
3	568	0	3	Palsson, Mrs. Nils (Alma Cornelia Berglund)	0	29.0	0	4	349909	21.0750	G6
4	672	0	1	Davidson, Mr. Thornton	1	31.0	1	0	F.C. 12750	52.0000	B71

In [4]:

```
data.isnull().sum()
```

Out[4]:

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

In [5]:

```
#Decision Tree Prediction
#importing relevant libraries
from sklearn.model_selection import train_test_split
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
#features extraction
x=data.drop(["Survived", "Name", "Cabin", "Ticket","Embarked"], axis=1)
y= data["Survived"]

#splitting data
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.20,random_state=10)

#applying tree algorithm
tree = tree.DecisionTreeClassifier()
tree.fit(x_train, y_train) #fitting our model
y_pred=tree.predict(x_test) # evaluating our model
print("score:{}".format(accuracy_score(y_test, y_pred)))
```

```
score:0.770949720670391
```

In []:

```
pip install graphviz
```

In [6]:

```
#Decision Tree Prediction
#importing relevant libraries
from sklearn.model_selection import train_test_split
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
#features extraction
x=data.drop(["Survived", "Name", "Cabin", "Ticket", "Embarked"], axis=1)
y= data["Survived"]

#splitting data
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.20,random_state=10)

#applying tree algorithm
clf = tree.DecisionTreeClassifier()
clf.fit(x_train, y_train) #fitting our model
```

Out[6]:

DecisionTreeClassifier()

In []:

In [7]:

```
import graphviz
dot_data = tree.export_graphviz(clf, out_file=None,
                                feature_names=x.columns.values,
                                class_names=['Survived','Not Survived'],
                                filled=True, rounded=True,
                                special_characters=True)
graph = graphviz.Source(dot_data)
graph
```

Out[7]:

In [9]:

```
#Decision Tree Pruning Code
dtree=tree.DecisionTreeClassifier(criterion="gini", splitter='random', max_leaf_nodes=10, m
dtree.fit(x_train, y_train)
dot_data = tree.export_graphviz(dtree, out_file=None,
                                feature_names=x.columns.values,
                                class_names=['Survived', 'Not Survived'],
                                filled=True, rounded=True,
                                special_characters=True)
graph = graphviz.Source(dot_data)
graph
```

Out[9]:

Se
gin
sam
value
class

In []: