In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
import re
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.tree import plot_tree
```

In [2]:
```
df=pd.read_csv("C2_train.gender_submission - C2_train.gender_submission.csv")
df
```

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	female	19.0	0	0	112053	30.0000

In [3]:
```
df1=df.fillna(value=0)
df1
```

Out[3]:

	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	0.0	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

In [4]:
```
df1.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          891 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        891 non-null    object 
 11  Embarked     891 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

In [5]:
```
df1.columns
```

Out[5]:
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

In [6]:
```
df2=df1[['PassengerId', 'Survived', 'Pclass', 'SibSp','Parch' ,'Embarked']]
df2
```

Out[6]:

	PassengerId	Survived	Pclass	SibSp	Parch	Embarked
0	1	0	3	1	0	S
1	2	1	1	1	0	C
2	3	1	3	0	0	S
3	4	1	1	1	0	S
4	5	0	3	0	0	S
...	...	...	...	...	...	...
886	887	0	2	0	0	S
887	888	1	1	0	0	S
888	889	0	3	1	2	S
889	890	1	1	0	0	C
890	891	0	3	0	0	Q

891 rows × 6 columns

In [7]:
```
df2['Embarked'].value_counts()
```

Out[7]:
```
S    644
C    168
Q     77
0      2
Name: Embarked, dtype: int64
```

In [8]:
```
x=df2.drop('Embarked',axis=1)
y=df2['Embarked']
```

In [9]:
```
g1={"Embarked":{'S':1,"C":2,"Q":3}}
df2=df2.replace(g1)
print(df2)
```

```
     PassengerId  Survived  Pclass  SibSp  Parch  Embarked
0              1         0       3      1      0         1
1              2         1       1      1      0         2
2              3         1       3      0      0         1
3              4         1       1      1      0         1
4              5         0       3      0      0         1
..           ...       ...     ...    ...    ...       ...
886          887         0       2      0      0         1
887          888         1       1      0      0         1
888          889         0       3      1      2         1
889          890         1       1      0      0         2
890          891         0       3      0      0         3

[891 rows x 6 columns]
```

In [10]:
```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.70)
```

In [11]:
```
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[11]:
```
RandomForestClassifier()
```

In [12]:
```
parameters = {'max_depth':[1,2,3,4,5],
              'min_samples_leaf':[5,10,15,20,25],
               'n_estimators':[10,20,30,40,50]}
```

In [13]:
```
grid_search = GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring='ac
grid_search.fit(x_train,y_train)
```

Out[13]:
```
GridSearchCV(cv=2, estimator=RandomForestClassifier(),
             param_grid={'max_depth': [1, 2, 3, 4, 5],
                         'min_samples_leaf': [5, 10, 15, 20, 25],
                         'n_estimators': [10, 20, 30, 40, 50]},
             scoring='accuracy')
```

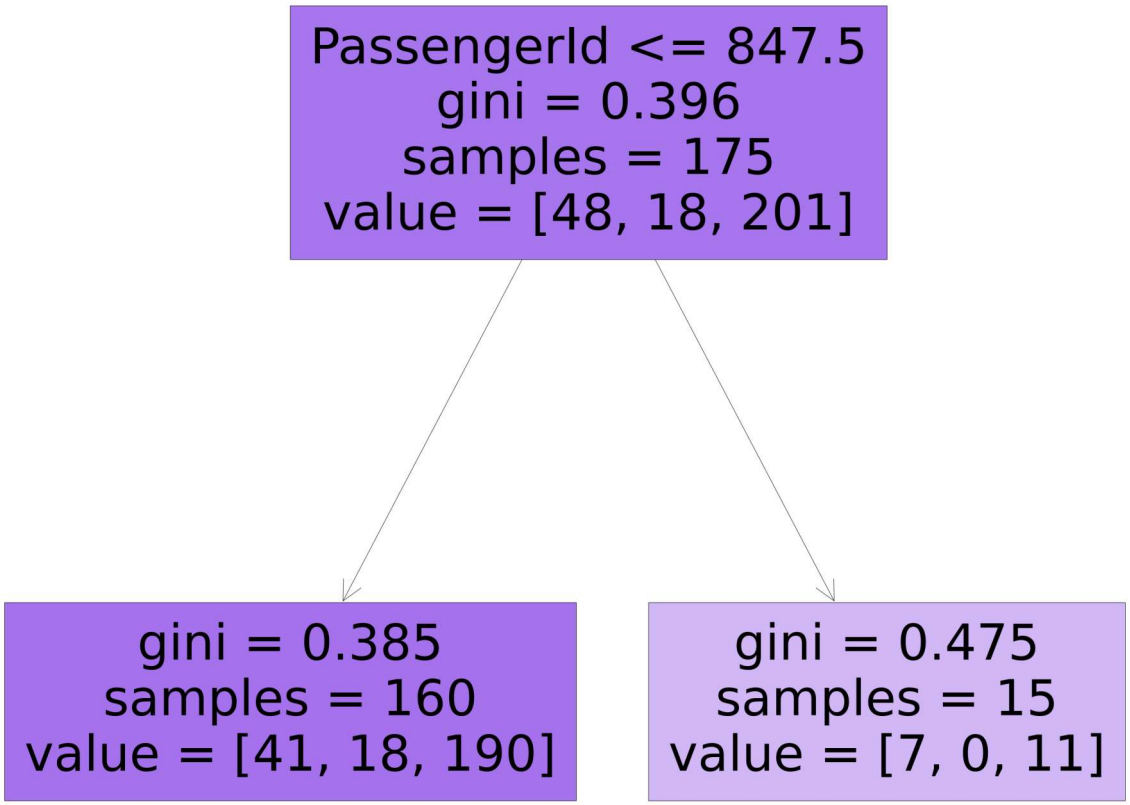
In [14]:
```
grid_search.best_score_
```

Out[14]:
```
0.7715464033217372
```

In [15]:
```
rfc_best =grid_search.best_estimator_
```

In [41]:
```
plt.figure(figsize=(50,49))
plot_tree(rfc_best.estimators_[3],feature_names=x.columns,filled=True)
```

Out[41]:
```
[Text(1395.0, 1997.73, 'PassengerId <= 847.5\ngini = 0.396\nsamples = 175\nva
lue = [48, 18, 201]'),
 Text(697.5, 665.9099999999999, 'gini = 0.385\nsamples = 160\nvalue = [41, 1
8, 190]'),
 Text(2092.5, 665.9099999999999, 'gini = 0.475\nsamples = 15\nvalue = [7, 0,
11]')]
```



In [ ]: