RANDOM FOREST - 9

In [44]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [45]:
```
df=pd.read_csv(r"C:\Users\BHOOMISH\Downloads\C9_Data.csv")
df
```

Out[45]:

	row_id	user_id	timestamp	gate_id
0	0	18	2022-07-29 09:08:54	7
1	1	18	2022-07-29 09:09:54	9
2	2	18	2022-07-29 09:09:54	9
3	3	18	2022-07-29 09:10:06	5
4	4	18	2022-07-29 09:10:08	5
...	...	...	...	...
37513	37513	6	2022-12-31 20:38:56	11
37514	37514	6	2022-12-31 20:39:22	6
37515	37515	6	2022-12-31 20:39:23	6
37516	37516	6	2022-12-31 20:39:31	9
37517	37517	6	2022-12-31 20:39:31	9

37518 rows × 4 columns

In [46]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 37518 entries, 0 to 37517
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   row_id     37518 non-null  int64 
 1   user_id    37518 non-null  int64 
 2   timestamp  37518 non-null  object
 3   gate_id    37518 non-null  int64 
dtypes: int64(3), object(1)
memory usage: 1.1+ MB
```

In [ ]:
```

```

In [59]:
```
df=df.dropna()
```

In [60]:
```
df.isnull().sum()
```

Out[60]:
```
row_id       0
user_id      0
timestamp    0
gate_id      0
dtype: int64
```

In [61]:
```
df.describe()
```

Out[61]:

	row_id	user_id	gate_id
count	37518.000000	37518.000000	37518.000000
mean	18758.500000	28.219015	6.819607
std	10830.658036	17.854464	3.197746
min	0.000000	0.000000	-1.000000
25%	9379.250000	12.000000	4.000000
50%	18758.500000	29.000000	6.000000
75%	28137.750000	47.000000	10.000000
max	37517.000000	57.000000	16.000000

In [62]:
```
df.columns
```

Out[62]:
```
Index(['row_id', 'user_id', 'timestamp', 'gate_id'], dtype='object')
```

In [63]:
```
df['user_id'].value_counts()
```

Out[63]:
37	2262
55	2238
6	2013
12	1953
19	1793
15	1756
18	1578
47	1341
53	1311
1	1299
33	1285
11	1281
49	1275
0	1250
39	1144
32	1076
54	1070
9	1034
50	994
29	990
3	989
48	743
14	696
17	677
27	603
35	601
46	502
57	497
24	416
42	359
26	316
34	284
23	261
25	247
40	242
31	191
56	137
41	124
43	124
20	115
22	96
28	64
45	57

```
7       49
36      48
2       39
8       29
10      17
38      13
5       10
30      10
52       5
21       5
44       4
51       3
4        2
Name: user_id, dtype: int64
```

In [ ]:
```
g1={"gate_id":{'6':1,'5':4}}
df=df.replace(g1)
print(df)
```

In [ ]:
```
x=df.drop("row_id",axis=1)
y=df["row_id"]
```

In [ ]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.70)
```

In [ ]:
```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

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

In [ ]:
```
from sklearn.model_selection import GridSearchCV
grid_search=GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring="accuracy")
grid_search.fit(x_train,y_train)
```

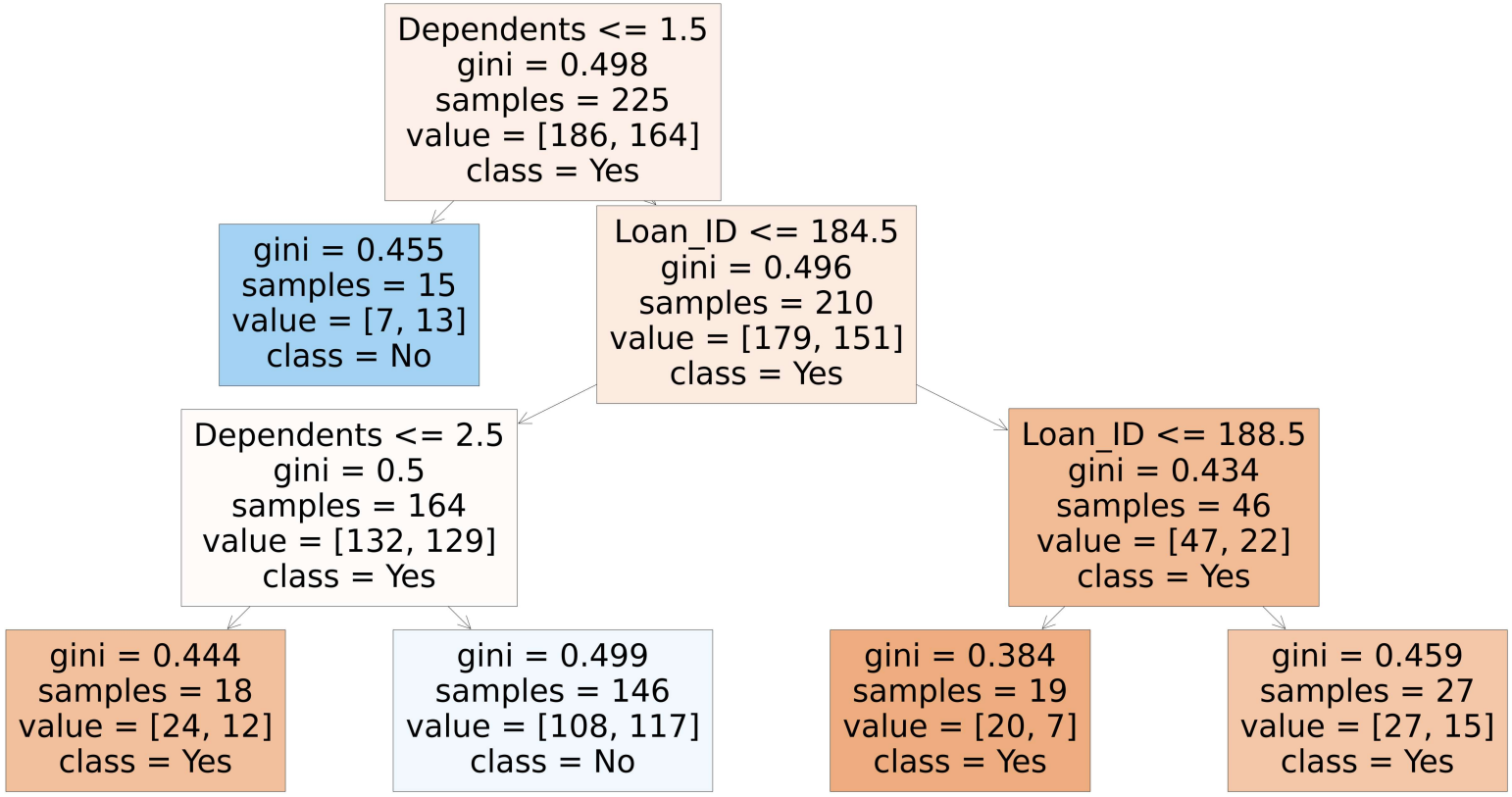
In [ ]:
```
grid_search.best_score_
```

In [41]:
```
parameters={'max_depth':[1,2,3,4,5],
            'min_samples_leaf':[5,10,15,20,25],
            'n_estimators':[10,20,30,40,50]}
```

In [42]:
```
rfc_best=grid_search.best_estimator_
```

In [43]:
```
from sklearn.tree import plot_tree
plt.figure(figsize=(80,40))
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=['Yes','No'],filled=True)
```

Out[43]:
```
[Text(0.375, 0.875, 'Dependents <= 1.5\ngini = 0.498\nsamples = 225\nvalue = [186, 164]\nclass = Yes'),
 Text(0.25, 0.625, 'gini = 0.455\nsamples = 15\nvalue = [7, 13]\nclass = No'),
 Text(0.5, 0.625, 'Loan_ID <= 184.5\ngini = 0.496\nsamples = 210\nvalue = [179, 151]\nclass = Yes'),
 Text(0.25, 0.375, 'Dependents <= 2.5\ngini = 0.5\nsamples = 164\nvalue = [132, 129]\nclass = Yes'),
 Text(0.125, 0.125, 'gini = 0.444\nsamples = 18\nvalue = [24, 12]\nclass = Yes'),
 Text(0.375, 0.125, 'gini = 0.499\nsamples = 146\nvalue = [108, 117]\nclass = No'),
 Text(0.75, 0.375, 'Loan_ID <= 188.5\ngini = 0.434\nsamples = 46\nvalue = [47, 22]\nclass = Yes'),
 Text(0.625, 0.125, 'gini = 0.384\nsamples = 19\nvalue = [20, 7]\nclass = Yes'),
 Text(0.875, 0.125, 'gini = 0.459\nsamples = 27\nvalue = [27, 15]\nclass = Yes')]
```



In [ ]: