In [52]:

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

In [53]:

```
df=pd.read_csv(r"C:\Users\user\Downloads\C10_loan1 - C10_loan1.csv")
df
```

Out[53]:

	Home Owner	Marital Status	Annual Income	Defaulted Borrower
0	Yes	Single	125	No
1	No	Married	100	No
2	No	Single	70	No
3	Yes	Married	120	No
4	No	Divorced	95	Yes
5	No	Married	60	No
6	Yes	Divorced	220	No
7	No	Single	85	Yes
8	No	Married	75	No
9	No	Single	90	Yes

In [54]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 4 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   Home Owner          10 non-null     object
 1   Marital Status      10 non-null     object
 2   Annual Income       10 non-null     int64 
 3   Defaulted Borrower  10 non-null     object
dtypes: int64(1), object(3)
memory usage: 448.0+ bytes
```

In [55]:

```
df['Defaulted Borrower'].value_counts()
```

Out[55]:

```
No     7
Yes    3
Name: Defaulted Borrower, dtype: int64
```

In [56]:

```
df.columns
```

Out[56]:

```
Index(['Home Owner', 'Marital Status', 'Annual Income', 'Defaulted Borrowe
r'], dtype='object')
```

In [57]:
```
df1=df[['Home Owner', 'Marital Status', 'Annual Income', 'Defaulted Borrower']
```

In [58]:
```
x=df1[['Annual Income']]
y=df['Defaulted Borrower']
```

In [59]:
```
g1={'Defaulted Borrower':{'Yes':0,'No':1}}
df1=df1.replace(g1)
print(df1)
```

```
  Home Owner Marital Status  Annual Income  Defaulted Borrower
0        Yes         Single            125                   1
1         No        Married            100                   1
2         No         Single             70                   1
3        Yes        Married            120                   1
4         No       Divorced             95                   0
5         No        Married             60                   1
6        Yes       Divorced            220                   1
7         No         Single             85                   0
8         No        Married             75                   1
9         No         Single             90                   0
```

In [60]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.30)
```

In [61]:
```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[61]:
```
RandomForestClassifier()
```

In [62]:
```
parameters={'max_depth':[1,2,3,4,5],
           'min_samples_leaf':[5,10,15,20,25],
           'n_estimators':[10,20,30,40,50]}
```

In [63]:
```
from sklearn.model_selection import GridSearchCV
grid_search=GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring='acc
grid_search.fit(x_train,y_train)
```

Out[63]:
```
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 [64]:
```
grid_search.best_score_
```

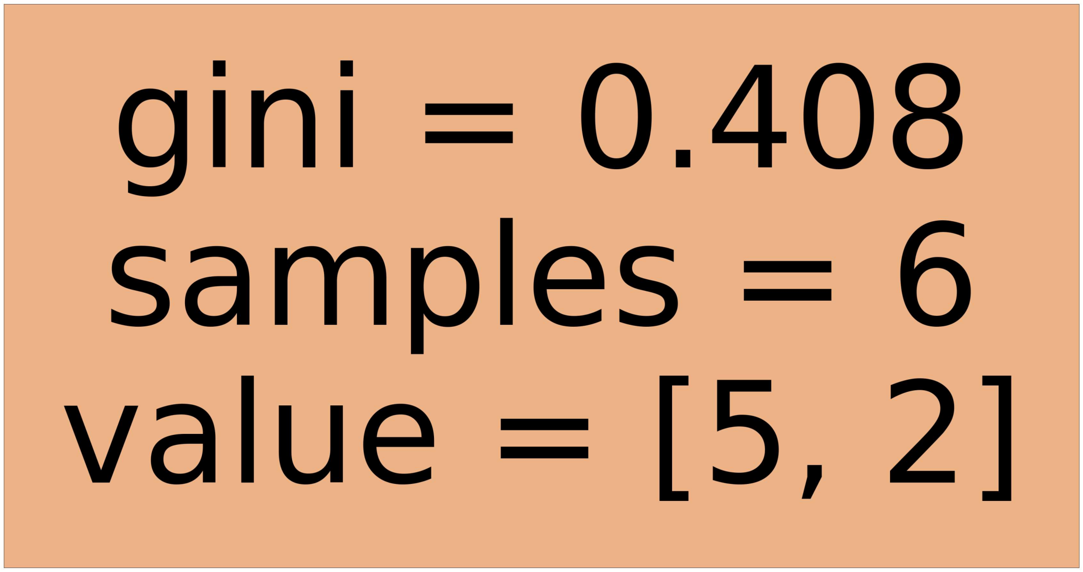
Out[64]:
```
0.7083333333333333
```

```
In [65]: rfc_best=grid_search.best_estimator_
```

```
In [66]: from sklearn.tree import plot_tree

         plt.figure(figsize=(80,40))
         plot_tree(rfc_best.estimators_[5],feature_names=x.columns,filled=True)
```

```
Out[66]: [Text(2232.0, 1087.2, 'gini = 0.408\nsamples = 6\nvalue = [5, 2]')]
```



```
In [ ]:
```

```
In [ ]:
```