In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df=pd.read_csv(r"C:\Users\user\Downloads\C4_framingham - C4_framingham.csv")
df
```

Out[2]:

r	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate
)	0.0	0.0	0	0	0	195.0	106.0	70.0	26.97	80.(
)	0.0	0.0	0	0	0	250.0	121.0	81.0	28.73	95.(
1	20.0	0.0	0	0	0	245.0	127.5	80.0	25.34	75.(
1	30.0	0.0	0	1	0	225.0	150.0	95.0	28.58	65.(
1	23.0	0.0	0	0	0	285.0	130.0	84.0	23.10	85.(
.	...	...	...	...	...	...	...	...	...	..
1	1.0	0.0	0	1	0	313.0	179.0	92.0	25.97	66.(
1	43.0	0.0	0	0	0	207.0	126.5	80.0	19.71	65.(
1	20.0	NaN	0	0	0	248.0	131.0	72.0	22.00	84.(
1	15.0	0.0	0	0	0	210.0	126.5	87.0	19.16	86.(
)	0.0	0.0	0	0	0	269.0	133.5	83.0	21.47	80.(

In [3]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4238 entries, 0 to 4237
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   male             4238 non-null   int64  
 1   age              4238 non-null   int64  
 2   education        4133 non-null   float64
 3   currentSmoker    4238 non-null   int64  
 4   cigsPerDay       4209 non-null   float64
 5   BPMeds           4185 non-null   float64
 6   prevalentStroke  4238 non-null   int64  
 7   prevalentHyp     4238 non-null   int64  
 8   diabetes         4238 non-null   int64  
 9   totChol          4188 non-null   float64
 10  sysBP            4238 non-null   float64
 11  diaBP            4238 non-null   float64
 12  BMI              4219 non-null   float64
 13  heartRate        4237 non-null   float64
 14  glucose          3850 non-null   float64
 15  TenYearCHD       4238 non-null   int64  
dtypes: float64(9), int64(7)
memory usage: 529.9 KB
```

In [4]:
```
df['TenYearCHD'].value_counts()
```

Out[4]:
```
0    3594
1     644
Name: TenYearCHD, dtype: int64
```

In [7]:
```
df1=df[['male','age','currentSmoker','currentSmoker','prevalentHyp','prevalentStrok
```

In [8]:
```
x=df1.drop('TenYearCHD',axis=1)
y=df['TenYearCHD']
```

In [9]:
```
g1={"1":{'0':1}}
df=df.replace(g1)
print(df)
```

```
      male  age  education  currentSmoker  cigsPerDay  BPMeds  \
0        1   39        4.0              0         0.0     0.0
1        0   46        2.0              0         0.0     0.0
2        1   48        1.0              1        20.0     0.0
3        0   61        3.0              1        30.0     0.0
4        0   46        3.0              1        23.0     0.0
...    ...  ...        ...            ...         ...     ...
4233     1   50        1.0              1         1.0     0.0
4234     1   51        3.0              1        43.0     0.0
4235     0   48        2.0              1        20.0     NaN
4236     0   44        1.0              1        15.0     0.0
4237     0   52        2.0              0         0.0     0.0

      prevalentStroke  prevalentHyp  diabetes  totChol  sysBP  diaBP    BMI  \
0                   0             0         0    195.0  106.0   70.0  26.97
1                   0             0         0    250.0  121.0   81.0  28.73
2                   0             0         0    245.0  127.5   80.0  25.34
3                   0             1         0    225.0  150.0   95.0  28.58
4                   0             0         0    285.0  130.0   84.0  23.10
...               ...           ...       ...      ...    ...    ...    ...
4233                0             1         0    313.0  179.0   92.0  25.97
4234                0             0         0    207.0  126.5   80.0  19.71
4235                0             0         0    248.0  131.0   72.0  22.00
4236                0             0         0    210.0  126.5   87.0  19.16
4237                0             0         0    269.0  133.5   83.0  21.47

      heartRate  glucose  TenYearCHD
0          80.0     77.0           0
1          95.0     76.0           0
2          75.0     70.0           0
3          65.0    103.0           1
4          85.0     85.0           0
...         ...      ...         ...
4233       66.0     86.0           1
4234       65.0     68.0           0
4235       84.0     86.0           0
4236       86.0      NaN           0
4237       80.0    107.0           0

[4238 rows x 16 columns]
```

In [10]:
```
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 [11]:
```
from sklearn.ensemble import RandomForestClassifier
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]:
```
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)
```

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_
```

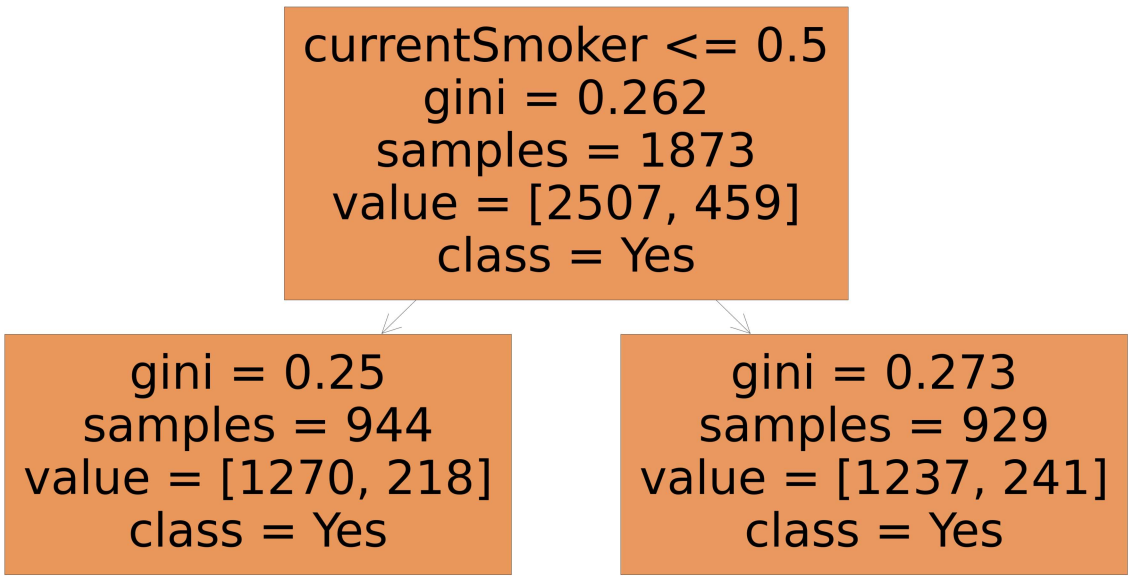
Out[14]:
```
0.8506405933917734
```

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

In [16]:
```
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'],
```

Out[16]:
```
[Text(2232.0, 1630.8000000000002, 'currentSmoker <= 0.5\ngini = 0.262\nsamples = 1
873\nvalue = [2507, 459]\nclass = Yes'),
 Text(1116.0, 543.5999999999999, 'gini = 0.25\nsamples = 944\nvalue = [1270, 218]
\nclass = Yes'),
 Text(3348.0, 543.5999999999999, 'gini = 0.273\nsamples = 929\nvalue = [1237, 241]
\nclass = Yes')]
```



In [ ]:

In [ ]: