

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

```
In [3]: df = pd.read_csv(r"C:\Users\user\Downloads\C5_health care diabetes (1).csv")
df
```

Out[3]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction
0	6	148	72	35	0	33.6	0.62
1	1	85	66	29	0	26.6	0.35
2	8	183	64	0	0	23.3	0.67
3	1	89	66	23	94	28.1	0.16
4	0	137	40	35	168	43.1	2.28
...
763	10	101	76	48	180	32.9	0.17
764	2	122	70	27	0	36.8	0.34
765	5	121	72	23	112	26.2	0.24
766	1	126	60	0	0	30.1	0.34
767	1	93	70	31	0	30.4	0.31

768 rows × 9 columns



```
In [4]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Pregnancies         768 non-null    int64
1   Glucose             768 non-null    int64
2   BloodPressure       768 non-null    int64
3   SkinThickness       768 non-null    int64
4   Insulin             768 non-null    int64
5   BMI                 768 non-null    float64
6   DiabetesPedigreeFunction 768 non-null    float64
7   Age                768 non-null    int64
8   Outcome            768 non-null    int64
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

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

```
Out[5]: Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',  
              'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'],  
             dtype='object')
```

```
In [7]: x = df[['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',  
              'BMI', 'DiabetesPedigreeFunction', 'Age']]  
y = df['Outcome']
```

```
In [8]: from sklearn.model_selection import train_test_split
```

```
In [9]: x_train,x_test,y_train,y_test = train_test_split(x,y,train_size=0.70)
```

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

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

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

```
In [12]: parameters = {  
          'max_depth':[11,12,13,14,15],  
          'min_samples_leaf':[15,20,25,30,35],  
          'n_estimators':[10,20,30,40,50]  
        }
```

```
In [13]: from sklearn.model_selection import GridSearchCV
```

```
In [14]: grid_search = GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring='accuracy')  
grid_search.fit(x_train,y_train)
```

```
Out[14]: GridSearchCV(cv=2, estimator=RandomForestClassifier(),  
                      param_grid={'max_depth': [11, 12, 13, 14, 15],  
                                   'min_samples_leaf': [15, 20, 25, 30, 35],  
                                   'n_estimators': [10, 20, 30, 40, 50]},  
                      scoring='accuracy')
```

```
In [15]: grid_search.best_score_
```

```
Out[15]: 0.7765563446706986
```

```
In [16]: from sklearn.tree import plot_tree
```

```
In [17]: rfc_best= grid_search.best_estimator_
```

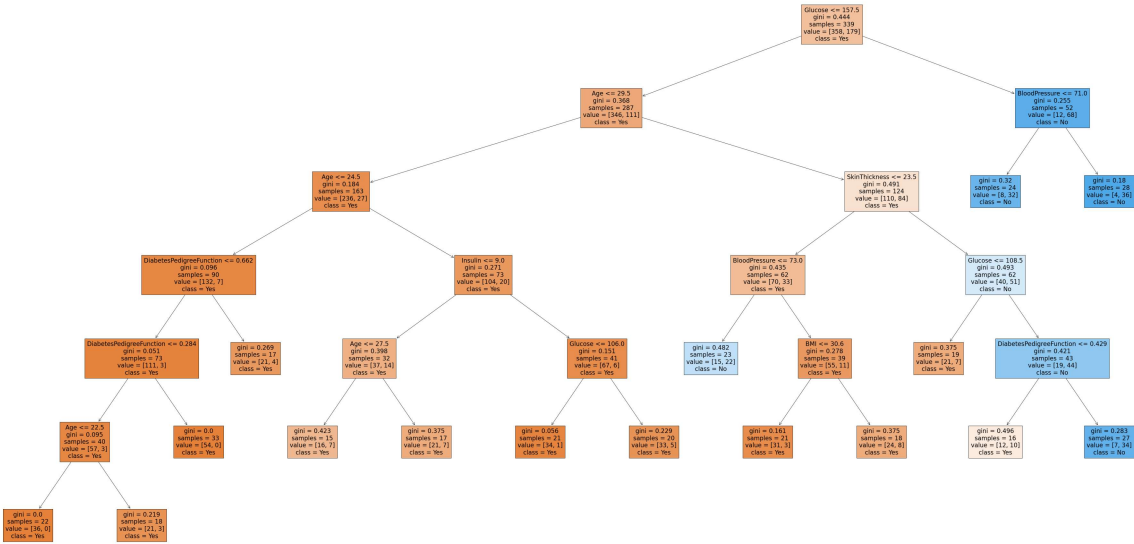
```
In [19]: plt.figure(figsize=(80,40))  
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=["Yes","No"])
```

```

Out[19]: [Text(3215.1428571428573, 2019.0857142857144, 'Glucose <= 157.5\ngini = 0.444\nsamples = 339\nvalue = [358, 179]\nclass = Yes'),
Text(2391.4285714285716, 1708.457142857143, 'Age <= 29.5\ngini = 0.368\nsamples = 287\nvalue = [346, 111]\nclass = Yes'),
Text(1381.7142857142858, 1397.8285714285716, 'Age <= 24.5\ngini = 0.184\nsamples = 163\nvalue = [236, 27]\nclass = Yes'),
Text(850.2857142857143, 1087.2, 'DiabetesPedigreeFunction <= 0.662\ngini = 0.096\nsamples = 90\nvalue = [132, 7]\nclass = Yes'),
Text(637.7142857142858, 776.5714285714287, 'DiabetesPedigreeFunction <= 0.284\ngini = 0.051\nsamples = 73\nvalue = [111, 3]\nclass = Yes'),
Text(425.14285714285717, 465.9428571428573, 'Age <= 22.5\ngini = 0.095\nsamples = 40\nvalue = [57, 3]\nclass = Yes'),
Text(212.57142857142858, 155.3142857142857, 'gini = 0.0\nsamples = 22\nvalue = [36, 0]\nclass = Yes'),
Text(637.7142857142858, 155.3142857142857, 'gini = 0.219\nsamples = 18\nvalue = [21, 3]\nclass = Yes'),
Text(850.2857142857143, 465.9428571428573, 'gini = 0.0\nsamples = 33\nvalue = [54, 0]\nclass = Yes'),
Text(1062.857142857143, 776.5714285714287, 'gini = 0.269\nsamples = 17\nvalue = [21, 4]\nclass = Yes'),
Text(1913.1428571428573, 1087.2, 'Insulin <= 9.0\ngini = 0.271\nsamples = 73\nvalue = [104, 20]\nclass = Yes'),
Text(1488.0, 776.5714285714287, 'Age <= 27.5\ngini = 0.398\nsamples = 32\nvalue = [37, 14]\nclass = Yes'),
Text(1275.4285714285716, 465.9428571428573, 'gini = 0.423\nsamples = 15\nvalue = [16, 7]\nclass = Yes'),
Text(1700.5714285714287, 465.9428571428573, 'gini = 0.375\nsamples = 17\nvalue = [21, 7]\nclass = Yes'),
Text(2338.285714285714, 776.5714285714287, 'Glucose <= 106.0\ngini = 0.151\nsamples = 41\nvalue = [67, 6]\nclass = Yes'),
Text(2125.714285714286, 465.9428571428573, 'gini = 0.056\nsamples = 21\nvalue = [34, 1]\nclass = Yes'),
Text(2550.857142857143, 465.9428571428573, 'gini = 0.229\nsamples = 20\nvalue = [33, 5]\nclass = Yes'),
Text(3401.1428571428573, 1397.8285714285716, 'SkinThickness <= 23.5\ngini = 0.491\nsamples = 124\nvalue = [110, 84]\nclass = Yes'),
Text(2976.0, 1087.2, 'BloodPressure <= 73.0\ngini = 0.435\nsamples = 62\nvalue = [70, 33]\nclass = Yes'),
Text(2763.4285714285716, 776.5714285714287, 'gini = 0.482\nsamples = 23\nvalue = [15, 22]\nclass = No'),
Text(3188.571428571429, 776.5714285714287, 'BMI <= 30.6\ngini = 0.278\nsamples = 39\nvalue = [55, 11]\nclass = Yes'),
Text(2976.0, 465.9428571428573, 'gini = 0.161\nsamples = 21\nvalue = [31, 3]\nclass = Yes'),
Text(3401.1428571428573, 465.9428571428573, 'gini = 0.375\nsamples = 18\nvalue = [24, 8]\nclass = Yes'),
Text(3826.2857142857147, 1087.2, 'Glucose <= 108.5\ngini = 0.493\nsamples = 62\nvalue = [40, 51]\nclass = No'),
Text(3613.714285714286, 776.5714285714287, 'gini = 0.375\nsamples = 19\nvalue = [21, 7]\nclass = Yes'),
Text(4038.857142857143, 776.5714285714287, 'DiabetesPedigreeFunction <= 0.429\ngini = 0.421\nsamples = 43\nvalue = [19, 44]\nclass = No'),
Text(3826.2857142857147, 465.9428571428573, 'gini = 0.496\nsamples = 16\nvalue = [12, 10]\nclass = Yes'),
Text(4251.428571428572, 465.9428571428573, 'gini = 0.283\nsamples = 27\nvalue = [7, 34]\nclass = No'),
Text(4038.857142857143, 1708.457142857143, 'BloodPressure <= 71.0\ngini = 0.

```

```
255\nsamples = 52\nvalue = [12, 68]\nnclass = No'),  
  Text(3826.2857142857147, 1397.8285714285716, 'gini = 0.32\nsamples = 24\nvalue = [8, 32]\nnclass = No'),  
  Text(4251.428571428572, 1397.8285714285716, 'gini = 0.18\nsamples = 28\nvalue = [4, 36]\nnclass = No'))]
```



```
In [ ]:
```