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

In [2]:
```
df = pd.read_csv(r"C:\Users\user\Downloads\C6_bmi (1).csv")
df
```

Out[2]:

	Gender	Height	Weight	Index
0	Male	174	96	4
1	Male	189	87	2
2	Female	185	110	4
3	Female	195	104	3
4	Male	149	61	3
...	...	...	...	...
495	Female	150	153	5
496	Female	184	121	4
497	Female	141	136	5
498	Male	150	95	5
499	Male	173	131	5

500 rows × 4 columns

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   Gender  500 non-null    object
 1   Height  500 non-null    int64 
 2   Weight  500 non-null    int64 
 3   Index   500 non-null    int64 
dtypes: int64(3), object(1)
memory usage: 15.8+ KB
```

In [4]:
```
df.columns
```

Out[4]:
```
Index(['Gender', 'Height', 'Weight', 'Index'], dtype='object')
```

In [5]:
```
x = df[['Height', 'Weight', 'Index']]
y = df['Gender']
```

In [7]:
```
e1 = {"Gender":{'Female':0,'Male':1}}
df1 = df.replace(e1)
df1
```

Out[7]:

	Gender	Height	Weight	Index
0	1	174	96	4
1	1	189	87	2
2	0	185	110	4
3	0	195	104	3
4	1	149	61	3
...	...	...	...	...
495	0	150	153	5
496	0	184	121	4
497	0	141	136	5
498	1	150	95	5
499	1	173	131	5

500 rows × 4 columns

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_
```

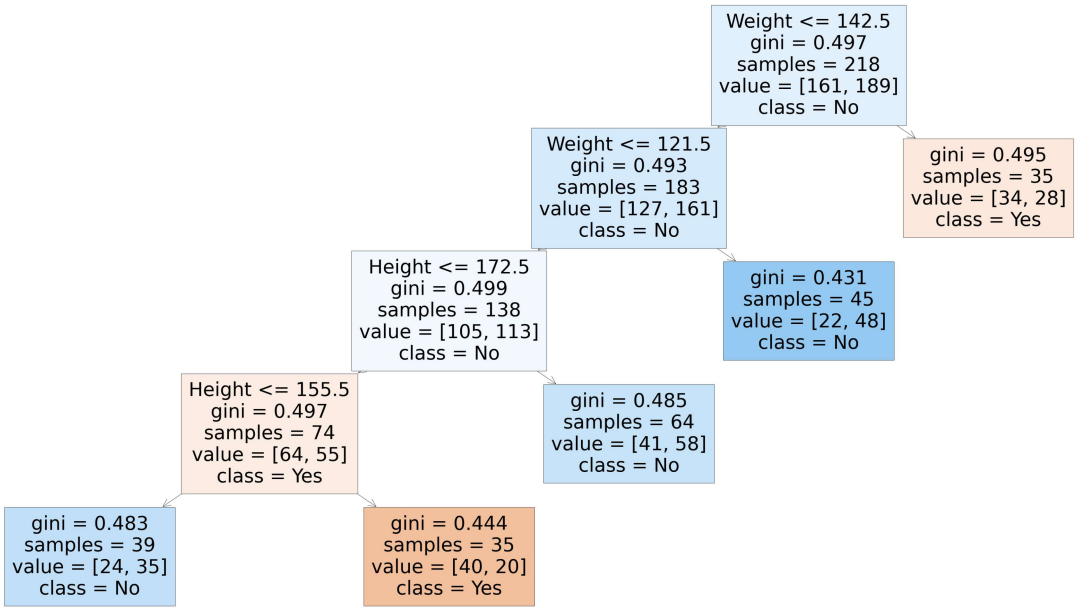
Out[15]:
```
0.5085714285714286
```

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

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

In [18]:
```
plt.figure(figsize=(80,40))
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=["Yes","No"],filled=True
```

Out[18]:
```
[Text(3188.5714285714284, 1956.96, 'Weight <= 142.5\ngini = 0.497\nsamples = 218\nvalue = [16
1, 189]\nclass = No'),
 Text(2550.8571428571427, 1522.0800000000002, 'Weight <= 121.5\ngini = 0.493\nsamples = 183\n
value = [127, 161]\nclass = No'),
 Text(1913.1428571428569, 1087.2, 'Height <= 172.5\ngini = 0.499\nsamples = 138\nvalue = [10
5, 113]\nclass = No'),
 Text(1275.4285714285713, 652.3200000000002, 'Height <= 155.5\ngini = 0.497\nsamples = 74\nva
lue = [64, 55]\nclass = Yes'),
 Text(637.7142857142857, 217.44000000000005, 'gini = 0.483\nsamples = 39\nvalue = [24, 35]\nc
lass = No'),
 Text(1913.1428571428569, 217.44000000000005, 'gini = 0.444\nsamples = 35\nvalue = [40, 20]\n
class = Yes'),
 Text(2550.8571428571427, 652.3200000000002, 'gini = 0.485\nsamples = 64\nvalue = [41, 58]\nc
lass = No'),
 Text(3188.5714285714284, 1087.2, 'gini = 0.431\nsamples = 45\nvalue = [22, 48]\nclass = N
o'),
 Text(3826.2857142857138, 1522.0800000000002, 'gini = 0.495\nsamples = 35\nvalue = [34, 28]\n
class = Yes')]
```



In [ ]:
```

```