

In [1]:

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

In [2]:

```
df=pd.read_csv('bmi.csv')
df
```

	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

In [3]:

```
df.columns
```

Out[3]:

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

In [5]:

```
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 [6]:

```
df['Gender'].value_counts()
```

Out[6]:

```
Female    255
Male      245
Name: Gender, dtype: int64
```

In [8]:

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

In [9]:

```
d={"Gender":{"Female":1, 'Male':2}}
df=df.replace(df)
print(df)
```

	Gender	Height	Weight	Index
0	Male	197	69	3
1	Male	158	117	4
2	Female	165	70	3
3	Female	145	140	3
4	Male	168	143	3
..
495	Female	196	140	3
496	Female	192	122	3
497	Female	167	156	3
498	Male	196	156	3
499	Male	196	70	3

[500 rows x 4 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=0.70)
```

In [11]:

```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[11]:

RandomForestClassifier()

Depth of Tree

In [12]:

```
parameters={"max_depth":[1,2,3,4,5], "min_samples_leaf":[5,23,45,76,78], 'n_estimators':[10
```

Cross Validate

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, 23, 45, 76, 78],
                         'n_estimators': [10, 23, 45, 65, 7]}},
             scoring='accuracy')
```

Score

In [14]:

```
grid_search.best_score_
```

Out[14]:

0.56

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'],filled
```

Out[16]:

```
[Text(1785.6, 1812.0, 'Weight <= 83.5\n gini = 0.498\n samples = 93\n value = [80, 70]\n nclass = Yes'),
 Text(892.8, 1087.2, 'gini = 0.463\n samples = 28\n value = [35, 20]\n nclass = Yes'),
 Text(2678.3999999999996, 1087.2, 'Height <= 163.0\n gini = 0.499\n samples = 65\n value = [45, 50]\n nclass = No'),
 Text(1785.6, 362.39999999999986, 'gini = 0.4\n samples = 23\n value = [8, 21]\n nclass = No'),
 Text(3571.2, 362.39999999999986, 'gini = 0.493\n samples = 42\n value = [37, 29]\n nclass = Yes')]
```

