

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

```
In [18]: data=pd.read_csv(r"C:\Users\user\Desktop\vicky\C6_bmi.csv")
```

```
In [19]: data.head()
```

Out[19]:

	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

```
In [37]: data["Gender"].fillna(value="Boy")
```

Out[37]:

0	Boy
1	Boy
2	Girl
3	Girl
4	Boy
	...
495	Girl
496	Girl
497	Girl
498	Boy
499	Boy

Name: Gender, Length: 500, dtype: object

```
In [38]: data['Gender'].value_counts()
```

Out[38]:

Girl	255
Boy	245

Name: Gender, dtype: int64

```
In [39]: x=data.drop('Gender',axis=1)
y=data['Gender']
```

```
In [40]: g1={"Gender":{"Female":"Girl","Male":"Boy"},}
data=data.replace(g1)
print(data)
```

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

[500 rows x 4 columns]

```
In [ ]:
```

```
In [42]: data.isna()
```

```
Out[42]:
```

	Gender	Height	Weight	Index
0	False	False	False	False
1	False	False	False	False
2	False	False	False	False
3	False	False	False	False
4	False	False	False	False
...
495	False	False	False	False
496	False	False	False	False
497	False	False	False	False
498	False	False	False	False
499	False	False	False	False

500 rows x 4 columns

```
In [43]: from sklearn.model_selection import train_test_split
```

```
In [44]: x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.30)
```

```
In [45]: from sklearn.ensemble import RandomForestClassifier
```

```
In [46]: rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

```
Out[46]: RandomForestClassifier()
```

```
In [47]: parameters = {'max_depth':[1,2,3,4,5],
                        'min_samples_leaf':[5,10,15,20,25],
                        'n_estimators':[10,20,30,40,50]
                      }
```

```
In [49]: 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[49]: 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 [50]: grid_search.best_score_
```

```
Out[50]: 0.5666666666666667
```

```
In [51]: from sklearn.tree import plot_tree
```

```
In [52]: rfc_best=grid_search.best_estimator_
```

```
In [53]: plt.figure(figsize=(80,40))
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=['Yes','No'],filled=True)
```

```
Out[53]: [Text(2232.0, 1812.0, 'Index <= 4.5\ngini = 0.499\nsamples = 94\nvalue = [71, 79]\nclass = N
o'),
Text(1116.0, 1087.2, 'Height <= 180.5\ngini = 0.492\nsamples = 54\nvalue = [37, 48]\nclass =
No'),
Text(558.0, 362.39999999999986, 'gini = 0.418\nsamples = 28\nvalue = [14, 33]\nclass = No'),
Text(1674.0, 362.39999999999986, 'gini = 0.478\nsamples = 26\nvalue = [23, 15]\nclass = Ye
s'),
Text(3348.0, 1087.2, 'Weight <= 133.5\ngini = 0.499\nsamples = 40\nvalue = [34, 31]\nclass =
Yes'),
Text(2790.0, 362.39999999999986, 'gini = 0.459\nsamples = 16\nvalue = [18, 10]\nclass = Ye
s'),
Text(3906.0, 362.39999999999986, 'gini = 0.491\nsamples = 24\nvalue = [16, 21]\nclass = N
o')]
```

In []: