

```
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\C6_bmi - C6_bmi.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['Index'].value_counts()
```

```
Out[4]: 5    198
4    130
2     69
3     68
1     22
0     13
Name: Index, dtype: int64
```

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

```
Out[5]: Index(['Gender', 'Height', 'Weight', 'Index'], dtype='object')
```

```
In [6]: df1=df[['Height', 'Weight', 'Index']]
```

```
In [7]: x=df1.drop('Index',axis=1)
y=df['Index']
```

```
In [9]: g1={"Index":{'5':1,'4':2,'3':3,'2':4,'1':5}}
df=df.replace(g1)
print(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

[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=0.30)
```

```
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='acc
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]: 0.7742857142857142
```

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

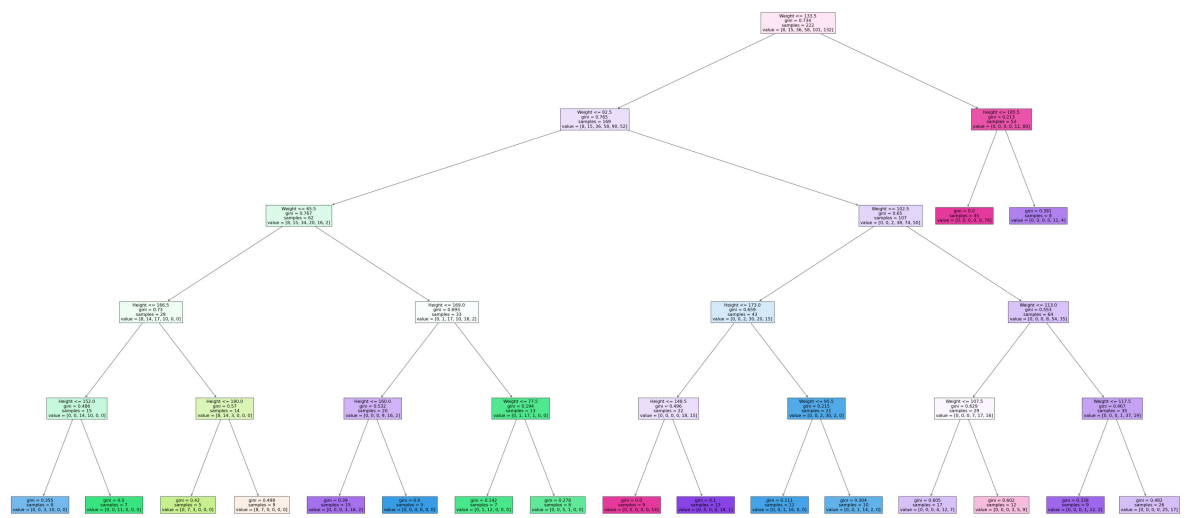
```
In [21]: from sklearn.tree import plot_tree  
  
plt.figure(figsize=(80,40))  
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,filled=True)
```

```

Out[21]: [Text(2999.25, 1993.2, 'Weight <= 133.5\ngini = 0.734\nsamples = 222\nvalue =
[8, 15, 36, 58, 101, 132]'),
Text(2232.0, 1630.8000000000002, 'Weight <= 82.5\ngini = 0.765\nsamples = 16
9\nvalue = [8, 15, 36, 58, 90, 52]'),
Text(1116.0, 1268.4, 'Weight <= 65.5\ngini = 0.767\nsamples = 62\nvalue =
[8, 15, 34, 20, 16, 2]'),
Text(558.0, 906.0, 'Height <= 166.5\ngini = 0.73\nsamples = 29\nvalue = [8,
14, 17, 10, 0, 0]'),
Text(279.0, 543.5999999999999, 'Height <= 152.0\ngini = 0.486\nsamples = 15
\nvalue = [0, 0, 14, 10, 0, 0]'),
Text(139.5, 181.19999999999982, 'gini = 0.355\nsamples = 8\nvalue = [0, 0,
3, 10, 0, 0]'),
Text(418.5, 181.19999999999982, 'gini = 0.0\nsamples = 7\nvalue = [0, 0, 11,
0, 0, 0]'),
Text(837.0, 543.5999999999999, 'Height <= 180.0\ngini = 0.57\nsamples = 14\n
value = [8, 14, 3, 0, 0, 0]'),
Text(697.5, 181.19999999999982, 'gini = 0.42\nsamples = 5\nvalue = [0, 7, 3,
0, 0, 0]'),
Text(976.5, 181.19999999999982, 'gini = 0.498\nsamples = 9\nvalue = [8, 7,
0, 0, 0, 0]'),
Text(1674.0, 906.0, 'Height <= 169.0\ngini = 0.693\nsamples = 33\nvalue =
[0, 1, 17, 10, 16, 2]'),
Text(1395.0, 543.5999999999999, 'Height <= 160.0\ngini = 0.532\nsamples = 20
\nvalue = [0, 0, 0, 9, 16, 2]'),
Text(1255.5, 181.19999999999982, 'gini = 0.39\nsamples = 15\nvalue = [0, 0,
0, 3, 16, 2]'),
Text(1534.5, 181.19999999999982, 'gini = 0.0\nsamples = 5\nvalue = [0, 0, 0,
6, 0, 0]'),
Text(1953.0, 543.5999999999999, 'Weight <= 77.5\ngini = 0.194\nsamples = 13
\nvalue = [0, 1, 17, 1, 0, 0]'),
Text(1813.5, 181.19999999999982, 'gini = 0.142\nsamples = 7\nvalue = [0, 1,
12, 0, 0, 0]'),
Text(2092.5, 181.19999999999982, 'gini = 0.278\nsamples = 6\nvalue = [0, 0,
5, 1, 0, 0]'),
Text(3348.0, 1268.4, 'Weight <= 102.5\ngini = 0.65\nsamples = 107\nvalue =
[0, 0, 2, 38, 74, 50]'),
Text(2790.0, 906.0, 'Height <= 173.0\ngini = 0.659\nsamples = 43\nvalue =
[0, 0, 2, 30, 20, 15]'),
Text(2511.0, 543.5999999999999, 'Height <= 148.5\ngini = 0.496\nsamples = 22
\nvalue = [0, 0, 0, 0, 18, 15]'),
Text(2371.5, 181.19999999999982, 'gini = 0.0\nsamples = 9\nvalue = [0, 0, 0,
0, 0, 14]'),
Text(2650.5, 181.19999999999982, 'gini = 0.1\nsamples = 13\nvalue = [0, 0,
0, 0, 18, 1]'),
Text(3069.0, 543.5999999999999, 'Weight <= 95.5\ngini = 0.215\nsamples = 21
\nvalue = [0, 0, 2, 30, 2, 0]'),
Text(2929.5, 181.19999999999982, 'gini = 0.111\nsamples = 11\nvalue = [0, 0,
1, 16, 0, 0]'),
Text(3208.5, 181.19999999999982, 'gini = 0.304\nsamples = 10\nvalue = [0, 0,
1, 14, 2, 0]'),
Text(3906.0, 906.0, 'Weight <= 113.0\ngini = 0.553\nsamples = 64\nvalue =
[0, 0, 0, 8, 54, 35]'),
Text(3627.0, 543.5999999999999, 'Weight <= 107.5\ngini = 0.629\nsamples = 29
\nvalue = [0, 0, 0, 7, 17, 16]'),
Text(3487.5, 181.19999999999982, 'gini = 0.605\nsamples = 17\nvalue = [0, 0,
0, 4, 12, 7]'),
Text(3766.5, 181.19999999999982, 'gini = 0.602\nsamples = 12\nvalue = [0, 0,

```

```
0, 3, 5, 9]'),
  Text(4185.0, 543.5999999999999, 'Weight <= 117.5\ngini = 0.467\nsamples = 35\n\nvalue = [0, 0, 0, 1, 37, 19]'),
  Text(4045.5, 181.19999999999982, 'gini = 0.338\nsamples = 9\n\nvalue = [0, 0, 0, 1, 12, 2]'),
  Text(4324.5, 181.19999999999982, 'gini = 0.482\nsamples = 26\n\nvalue = [0, 0, 0, 0, 25, 17]'),
  Text(3766.5, 1630.8000000000002, 'Height <= 185.5\ngini = 0.213\nsamples = 53\n\nvalue = [0, 0, 0, 0, 11, 80]'),
  Text(3627.0, 1268.4, 'gini = 0.0\nsamples = 45\n\nvalue = [0, 0, 0, 0, 0, 76]'),
  Text(3906.0, 1268.4, 'gini = 0.391\nsamples = 8\n\nvalue = [0, 0, 0, 0, 11, 4]')]]
```



```
In [ ]:
```

```
In [ ]:
```