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\C5_health care diabetes - C5_health care diabetes.csv
df
```

Out[2]:

| | Pregnancies | Glucose | BloodPressure | SkinThickness | Insulin | BMI | DiabetesPedigreeFunction | Age | Outcome |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 6 | 148 | 72 | 35 | 0 | 33.6 | 0.627 | 50 | 1 |
| **1** | 1 | 85 | 66 | 29 | 0 | 26.6 | 0.351 | 31 | 0 |
| **2** | 8 | 183 | 64 | 0 | 0 | 23.3 | 0.672 | 32 | 1 |
| **3** | 1 | 89 | 66 | 23 | 94 | 28.1 | 0.167 | 21 | 0 |
| **4** | 0 | 137 | 40 | 35 | 168 | 43.1 | 2.288 | 33 | 1 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **763** | 10 | 101 | 76 | 48 | 180 | 32.9 | 0.171 | 63 | 0 |
| **764** | 2 | 122 | 70 | 27 | 0 | 36.8 | 0.340 | 27 | 0 |
| **765** | 5 | 121 | 72 | 23 | 112 | 26.2 | 0.245 | 30 | 0 |
| **766** | 1 | 126 | 60 | 0 | 0 | 30.1 | 0.349 | 47 | 1 |
| **767** | 1 | 93 | 70 | 31 | 0 | 30.4 | 0.315 | 23 | 0 |

768 rows × 9 columns

In [3]:
```
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 [4]:
```
df['Outcome'].value_counts()
```

Out[4]:
```
0    500
1    268
Name: Outcome, dtype: int64
```

In [5]:
```
df1=df[['Pregnancies','Glucose','BloodPressure','SkinThickness','Insulin','BMI','Age','Outcome
```

In [6]:
```
x=df1.drop('Outcome',axis=1)
y=df['Outcome']
```

In [7]:
```
g1={"1":{'0':1}}
df=df.replace(g1)
print(df)
```

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

     DiabetesPedigreeFunction  Age  Outcome
0                       0.627   50        1
1                       0.351   31        0
2                       0.672   32        1
3                       0.167   21        0
4                       2.288   33        1
..                        ...  ...      ...
763                     0.171   63        0
764                     0.340   27        0
765                     0.245   30        0
766                     0.349   47        1
767                     0.315   23        0

[768 rows x 9 columns]
```

In [8]:
```
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 [9]:
```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[9]:
```
RandomForestClassifier()
```

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

In [11]:
```
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[11]:
```
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 [12]:
```
grid_search.best_score_
```

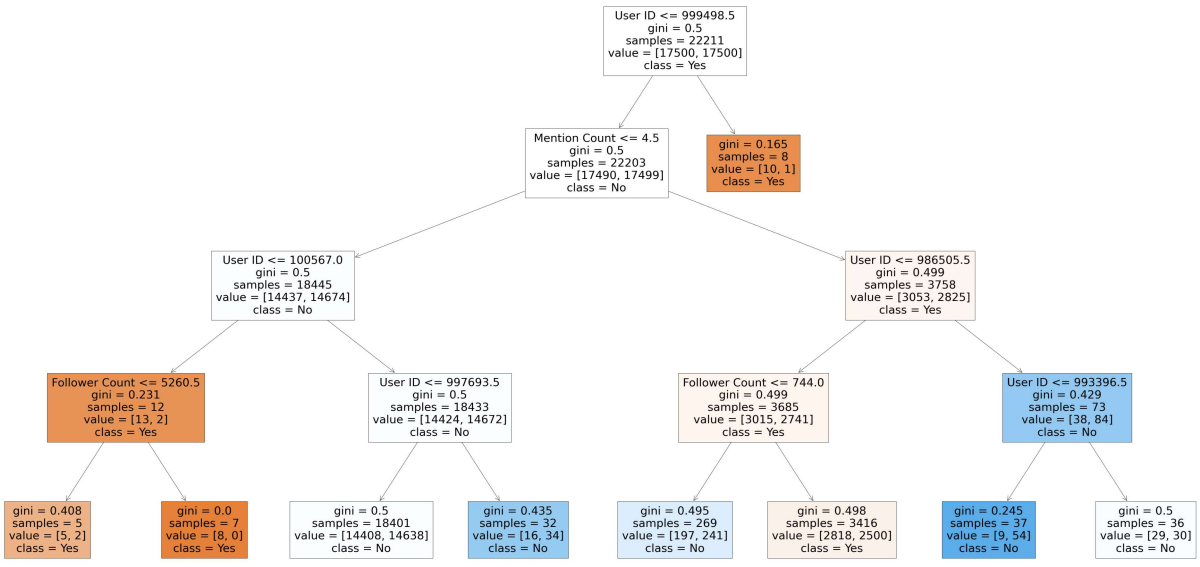
Out[12]:
```
0.7747045441935305
```

In [13]:
```
rfc_best=grid_search.best_estimator_
```

In [14]:
```
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=Tru
```

Out[14]:
```
[Text(2511.0, 1956.96, 'User ID <= 999498.5\ngini = 0.5\nsamples = 22211\nvalue = [17500, 17
500]\nclass = Yes'),
 Text(2232.0, 1522.0800000000002, 'Mention Count <= 4.5\ngini = 0.5\nsamples = 22203\nvalue
= [17490, 17499]\nclass = No'),
 Text(1116.0, 1087.2, 'User ID <= 100567.0\ngini = 0.5\nsamples = 18445\nvalue = [14437, 146
74]\nclass = No'),
 Text(558.0, 652.3200000000002, 'Follower Count <= 5260.5\ngini = 0.231\nsamples = 12\nvalue
= [13, 2]\nclass = Yes'),
 Text(279.0, 217.44000000000005, 'gini = 0.408\nsamples = 5\nvalue = [5, 2]\nclass = Yes'),
 Text(837.0, 217.44000000000005, 'gini = 0.0\nsamples = 7\nvalue = [8, 0]\nclass = Yes'),
 Text(1674.0, 652.3200000000002, 'User ID <= 997693.5\ngini = 0.5\nsamples = 18433\nvalue =
[14424, 14672]\nclass = No'),
 Text(1395.0, 217.44000000000005, 'gini = 0.5\nsamples = 18401\nvalue = [14408, 14638]\nclas
s = No'),
 Text(1953.0, 217.44000000000005, 'gini = 0.435\nsamples = 32\nvalue = [16, 34]\nclass = N
o'),
 Text(3348.0, 1087.2, 'User ID <= 986505.5\ngini = 0.499\nsamples = 3758\nvalue = [3053, 282
5]\nclass = Yes'),
 Text(2790.0, 652.3200000000002, 'Follower Count <= 744.0\ngini = 0.499\nsamples = 3685\nval
ue = [3015, 2741]\nclass = Yes'),
 Text(2511.0, 217.44000000000005, 'gini = 0.495\nsamples = 269\nvalue = [197, 241]\nclass =
No'),
 Text(3069.0, 217.44000000000005, 'gini = 0.498\nsamples = 3416\nvalue = [2818, 2500]\nclass
= Yes'),
 Text(3906.0, 652.3200000000002, 'User ID <= 993396.5\ngini = 0.429\nsamples = 73\nvalue =
[38, 84]\nclass = No'),
 Text(3627.0, 217.44000000000005, 'gini = 0.245\nsamples = 37\nvalue = [9, 54]\nclass = N
o'),
 Text(4185.0, 217.44000000000005, 'gini = 0.5\nsamples = 36\nvalue = [29, 30]\nclass = No'),
 Text(2790.0, 1522.0800000000002, 'gini = 0.165\nsamples = 8\nvalue = [10, 1]\nclass = Ye
s')]
```




In [ ]:
```
```