Assignment of Decision Tree Classifiers

In [ ]:

```
import pandas as pd
df= pd.read_csv('mldata1.csv')
df.head()
```

Out[ ]:

	age	height	weight	gender	likeness
0	27	170.688	76.0	Male	Biryani
1	41	165.000	70.0	Male	Biryani
2	29	171.000	80.0	Male	Biryani
3	27	173.000	102.0	Male	Biryani
4	29	164.000	67.0	Male	Biryani

Converting Likeness column into dummies

In [ ]:

```
df['likeness']= df['likeness'].replace('Biryani',1)
df['likeness']= df['likeness'].replace('Pakora',3)
df['likeness']= df['likeness'].replace('Samosa',2)


df.tail()
```

Out[ ]:

	age	height	weight	gender	likeness
240	31	160.0	60.0	1	3
241	26	172.0	70.0	1	1
242	40	178.0	80.0	1	1
243	25	5.7	65.0	1	1
244	33	157.0	56.0	0	2

Converting Gender dummies to string

In [ ]:

```
df['gender']= df['gender'].replace(1,'Male')
df['gender']= df['gender'].replace(0,'Female')
df.head()
```

Out[ ]:

	age	height	weight	gender	likeness
0	27	170.688	76.0	Male	1
1	41	165.000	70.0	Male	1
2	29	171.000	80.0	Male	1
3	27	173.000	102.0	Male	1
4	29	164.000	67.0	Male	1

predicting the gender using age height weight and likeness

In [ ]:

```
x = df[['age','height', 'weight', 'likeness']]
y = df['gender']
x.head()
```

Out[ ]:

	age	height	weight	likeness
0	27	170.688	76.0	1
1	41	165.000	70.0	1
2	29	171.000	80.0	1
3	27	173.000	102.0	1
4	29	164.000	67.0	1

Creating Model and Prediction through the Model

In [ ]:

```
from sklearn.tree import DecisionTreeClassifier
model= DecisionTreeClassifier()

model= DecisionTreeClassifier().fit(x, y)
```

In [ ]:

```
model.predict([[33,157,66,3]])
```

```
C:\Users\Haier\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\b
ase.py:450: UserWarning: X does not have valid feature names, but DecisionTreeClas
sifier was fitted with feature names
  warnings.warn(
```

Out[ ]:

```
array(['Male'], dtype=object)
```

checking the accuracy of model

In [ ]:

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
x_train, x_test, y_train,y_test=train_test_split(x, y, test_size=0.2, random_state
model= DecisionTreeClassifier()
model.fit(x_train, y_train)
```

Out[ ]:

```
DecisionTreeClassifier()
```

In [ ]:

```
predicted_values = model.predict(x_test)
predicted_values
```

Out[ ]:

```
array(['Female', 'Male', 'Male', 'Male', 'Male', 'Male', 'Male', 'Female',
       'Male', 'Male', 'Female', 'Female', 'Male', 'Male', 'Male', 'Male',
       'Female', 'Male', 'Male', 'Male', 'Male', 'Male', 'Male', 'Male',
       'Female', 'Male', 'Male', 'Female', 'Male', 'Male', 'Female',
       'Male', 'Female', 'Male', 'Male', 'Male', 'Female', 'Male', 'Male',
       'Female', 'Male', 'Male', 'Female', 'Male', 'Male', 'Male', 'Male',
       'Male', 'Female'], dtype=object)
```

Accuracy of the Model

In [ ]:

```
score= accuracy_score(y_test, predicted_values)
score
```

Out[ ]:

```
0.7551020408163265
```

Saving and Exporting model

In [ ]:

```
import pandas as pd
from sklearn.tree import DecisionTreeClassifier
import joblib

model = DecisionTreeClassifier().fit(x,y)

joblib.dump(model, "Gender.joblib")
```

Out[ ]:

```
['Gender.joblib']
```

visualization of model

In [ ]:

```
from sklearn import tree
model= DecisionTreeClassifier().fit(x,y)
## graphic evaluation/look nto what happend
tree.export_graphviz(model,
                     out_file="Gender.dot",
                     feature_names=['age', 'weight','height', 'likeness'],
                     class_names=sorted(y.unique()),
                     label='all',
                     rounded= True,
                     filled= True)
```