```
import numpy as np # linear algebra
import pandas as pd
```

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
```

```
df = pd.read_csv('/content/milknew.csv')
df
```

	pH	Temprature	Taste	Odor	Fat	Turbidity	Colour	Grade
0	6.6	35	1	0	1	0	254	high
1	6.6	36	0	1	0	1	253	high
2	8.5	70	1	1	1	1	246	low
3	9.5	34	1	1	0	1	255	low
4	6.6	37	0	0	0	0	255	medium
...	...	...	...	...	...	...	...	...
1054	6.7	45	1	1	0	0	247	medium
1055	6.7	38	1	0	1	0	255	high
1056	3.0	40	1	1	1	1	255	low
1057	6.8	43	1	0	1	0	250	high
1058	8.6	55	0	1	1	1	255	low

1059 rows × 8 columns

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1059 entries, 0 to 1058
Data columns (total 8 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   pH          1059 non-null   float64
 1   Temprature  1059 non-null   int64  
 2   Taste       1059 non-null   int64  
 3   Odor        1059 non-null   int64  
 4   Fat         1059 non-null   int64  
 5   Turbidity   1059 non-null   int64  
 6   Colour      1059 non-null   int64  
 7   Grade       1059 non-null   object 
dtypes: float64(1), int64(6), object(1)
memory usage: 66.3+ KB
```

```
df.describe()
```

	pH	Temprature	Taste	Odor	Fat	Turbidity	Colour
count	1059.000000	1059.000000	1059.000000	1059.000000	1059.000000	1059.000000	1059.000000
mean	6.630123	44.226629	0.546742	0.432483	0.671388	0.491029	251.840415
std	1.399679	10.098364	0.498046	0.495655	0.469930	0.500156	4.307424
min	3.000000	34.000000	0.000000	0.000000	0.000000	0.000000	240.000000
25%	6.500000	38.000000	0.000000	0.000000	0.000000	0.000000	250.000000
50%	6.700000	41.000000	1.000000	0.000000	1.000000	0.000000	255.000000
75%	6.800000	45.000000	1.000000	1.000000	1.000000	1.000000	255.000000
max	9.500000	90.000000	1.000000	1.000000	1.000000	1.000000	255.000000

```
X = df.iloc[:, :-1].values
Y = df.iloc[:, -1].values


#Split data
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=42)


#Modelling
model_Decission_tree = DecisionTreeClassifier(criterion='entropy', random_state=42)
model_Decission_tree = model_Decission_tree.fit(X_train, Y_train)


#Predict data train and test
Y_pred_train = model_Decission_tree.predict(X_train)
Y_pred_test = model_Decission_tree.predict(X_test)


#Evaluation report model
grade = ['high','medium', 'low']

print('Classsification Report Model : \n')
print(classification_report(Y_test, Y_pred_test, target_names=grade))
```

```
Classsification Report Model :

                  precision    recall  f1-score   support

            high       1.00      0.98      0.99        48
          medium       1.00      0.99      0.99        78
             low       0.98      1.00      0.99        86

        accuracy                           0.99       212
       macro avg       0.99      0.99      0.99       212
    weighted avg       0.99      0.99      0.99       212
```

```
#Entering new data to get milk quality grade

pH = float(input('Input PH = '))
Temprature = float(input('Input Temprature = '))
Taste = bool(input('Input Tastte (True or False) = '))
Odor = bool(input('Input Odor (True or False) = '))
Fat = bool(input('Input Fat (True or False) = '))
Turbidity = bool(input('Input Turbidity (True or False) = '))
Colour= int(input('Input Colour = '))
grade_milk = [[pH, Temprature, Taste, Odor, Fat, Turbidity, Colour]]

predict_data = model_Decission_tree.predict(grade_milk)
print('Quality of the milk', predict_data)
```

```
Input PH = 9
Input Temprature = 45
Input Tastte (True or False) = 1
Input Odor (True or False) = 0.45
Input Fat (True or False) = 0.78
Input Turbidity (True or False) = 87
Input Colour = 255
Quality of the milk ['low']
```