In [30]:
```
import pandas as pd
movie_info = pd.read_csv("Movie%20Interests.csv")
```

In [31]:
```
movie_info
```

Out[31]:

	Age	Gender	Interest
0	8	1	Animation
1	11	1	Animation
2	12	1	Animation
3	16	1	Action
4	18	1	Action
5	19	1	Action
6	23	1	Drama
7	26	1	Drama
8	27	1	Drama
9	7	0	Animation
10	9	0	Animation
11	10	0	Animation
12	26	0	Action
13	27	0	Action
14	30	0	Action
15	31	0	Drama
16	34	0	Drama
17	35	0	Drama

In [32]:
```
input_dataset = movie_info.drop(columns = ["Interest"])
```

In [33]:
```
input_dataset
```

Out[33]:

	Age	Gender
0	8	1
1	11	1
2	12	1
3	16	1
4	18	1
5	19	1
6	23	1
7	26	1
8	27	1
9	7	0
10	9	0
11	10	0
12	26	0
13	27	0
14	30	0
15	31	0
16	34	0
17	35	0

In [34]:
```
ouput_dataset = movie_info["Interest"]
```

In [35]:
```
ouput_dataset
```

Out[35]:
```
0     Animation
1     Animation
2     Animation
3        Action
4        Action
5        Action
6         Drama
7         Drama
8         Drama
9     Animation
10    Animation
11    Animation
12       Action
13       Action
14       Action
15        Drama
16        Drama
17        Drama
Name: Interest, dtype: object
```

bulid the ML model (DecisionTreeClassifier)

In [36]:
```
from sklearn.tree import DecisionTreeClassifier
movie_model = DecisionTreeClassifier()
movie_model.fit(input_dataset,ouput_dataset)
```

Out[36]:
```
▾ DecisionTreeClassifier
DecisionTreeClassifier()
```

predict the outcome

In [37]:
```
movie_interst = movie_model.predict([[9,1],[33,0]])
movie_interst
```

```
/opt/homebrew/anaconda3/lib/python3.11/site-packages/sklearn/base.py:464:
UserWarning: X does not have valid feature names, but DecisionTreeClassif
ier was fitted with feature names
  warnings.warn(
```

Out[37]:
```
array(['Animation', 'Drama'], dtype=object)
```

In [38]:
```
movie_interst = movie_model.predict([[42,1],[43,0]])
movie_interst
```

```
/opt/homebrew/anaconda3/lib/python3.11/site-packages/sklearn/base.py:464:
UserWarning: X does not have valid feature names, but DecisionTreeClassif
ier was fitted with feature names
  warnings.warn(
```

Out[38]:
```
array(['Drama', 'Drama'], dtype=object)
```

In [56]:
```
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

input_dataset_train , input_dataset_test , output_dataset_train , output_

movie_model = DecisionTreeClassifier()
movie_model.fit(input_dataset_train , output_dataset_train)

movie_interst = movie_model.predict(input_dataset_test)

accuracy_info = accuracy_score(output_dataset_test,movie_interst)
accuracy_info
```

Out[56]:
```
1.0
```

In [57]:
```
from sklearn.tree import DecisionTreeClassifier
movie_model = DecisionTreeClassifier()
movie_model.fit(input_dataset,ouput_dataset)
movie_interst = movie_model.predict([[9,1],[33,0]])
movie_interst
```

```
/opt/homebrew/anaconda3/lib/python3.11/site-packages/sklearn/base.py:464:
UserWarning: X does not have valid feature names, but DecisionTreeClassif
ier was fitted with feature names
  warnings.warn(
```

Out[57]:
```
array(['Animation', 'Drama'], dtype=object)
```

In [59]:
```
from sklearn.tree import DecisionTreeClassifier
import joblib
movie_model = DecisionTreeClassifier()
movie_model.fit(input_dataset,ouput_dataset)
joblib.dump(movie_model , "Movie-Interst-Identifier")
```

Out[59]:
```
['Movie-Interst-Identifier']
```

In [61]:
```
model_movie_interst = joblib.load("Movie-Interst-Identifier")
movie_interst = model_movie_interst.predict([[9,1],[33,0]])
movie_interst
```

```
/opt/homebrew/anaconda3/lib/python3.11/site-packages/sklearn/base.py:464:
UserWarning: X does not have valid feature names, but DecisionTreeClassif
ier was fitted with feature names
  warnings.warn(
```

Out[61]:
```
array(['Animation', 'Drama'], dtype=object)
```

In [ ]:
```

```