

```
In [15]: import numpy as np
import pandas as pd
```

```
In [16]: df=pd.read_csv('weight-height.csv')
```

```
In [17]: df
```

```
Out[17]:
```

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470
4	Male	69.881796	206.349801
...
9995	Female	66.172652	136.777454
9996	Female	67.067155	170.867906
9997	Female	63.867992	128.475319
9998	Female	69.034243	163.852461
9999	Female	61.944246	113.649103

10000 rows × 3 columns

```
In [18]: df.head(4)
```

```
Out[18]:
```

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470

```
In [19]: df.isnull().sum()
```

```
Out[19]: Gender      0
Height      0
Weight      0
dtype: int64
```

```
In [20]: df['Gender'].value_counts()
```

```
Out[20]: Male      5000
Female    5000
Name: Gender, dtype: int64
```

```
In [21]: len(df['Gender'].value_counts())
```

```
Out[21]: 2
```

```
In [22]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  -
 0   Gender  10000 non-null   object
 1   Height  10000 non-null   float64
 2   Weight  10000 non-null   float64
dtypes: float64(2), object(1)
memory usage: 234.5+ KB
```

```
In [23]: X1 = df.drop(['Gender'], axis = 1)
        Y1 = df['Gender']
```

```
In [24]: X1.head(2)
```

```
Out[24]:
```

	Height	Weight
0	73.847017	241.893563
1	68.781904	162.310473

```
In [25]: from sklearn.model_selection import train_test_split
```

```
In [26]: X_train, X_test, Y_train, Y_test = train_test_split(X1,Y1,test_size=0.2,random_state=42)
```

```
In [27]: X_train.shape, X_test.shape
```

```
Out[27]: ((8000, 2), (2000, 2))
```

```
In [28]: from sklearn.neighbors import KNeighborsClassifier
```

```
In [29]: knn = KNeighborsClassifier(n_neighbors=3)
```

```
In [30]: knn.fit(X_train, Y_train)
```

```
Out[30]: KNeighborsClassifier(n_neighbors=3)
```

```
In [31]: X_test
```

Out[31]:

	Height	Weight
--	--------	--------

2077	69.010642	177.586828
9207	65.725572	130.568403
6755	61.835671	118.361357
6704	63.593709	144.419187
9528	66.442292	145.653809
...
8656	66.423595	153.000559
9133	64.725015	130.409987
7566	57.553505	108.151688
3970	70.225090	196.949126
3602	68.994328	199.782401

2000 rows × 2 columns

```
In [32]: y_predict1 = knn.predict(X_test)
y_predict1
```

Out[32]: array(['Male', 'Female', 'Female', ..., 'Female', 'Male', 'Male'],
dtype=object)

```
In [33]: X_test.head(2)
```

Out[33]:

	Height	Weight
--	--------	--------

2077	69.010642	177.586828
9207	65.725572	130.568403

```
In [34]: knn.predict([[73.847017, 241.893563]])
```

C:\Users\hi\anaconda3\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but KNeighborsClassifier was fitted with feature names

warnings.warn(

Out[34]: array(['Male'], dtype=object)

```
In [35]: Y_test[:2]
```

Out[35]: 2077 Male
9207 Female
Name: Gender, dtype: object

```
In [36]: df.tail()
```

Out[36]:

	Gender	Height	Weight
9995	Female	66.172652	136.777454
9996	Female	67.067155	170.867906
9997	Female	63.867992	128.475319
9998	Female	69.034243	163.852461
9999	Female	61.944246	113.649103

In [37]: `from sklearn.metrics import confusion_matrix, accuracy_score`

In [38]: `confusion_matrix(Y_test, y_predict1)`

Out[38]: `array([[879, 97],
 [104, 920]], dtype=int64)`

In [39]: `accuracy_score(Y_test, y_predict1)`

Out[39]: `0.8995`

In []:

In []:

In []: