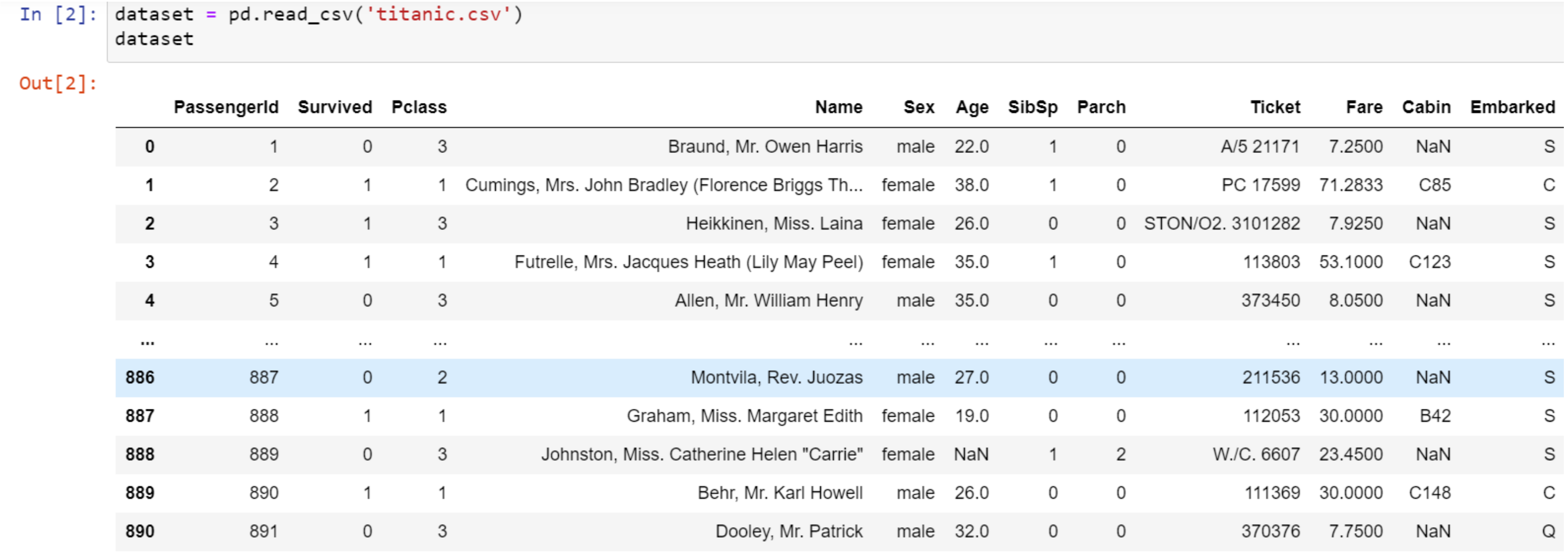
# Tugas Materi 7

Zul Fauzi Oktaviansyah

2110181056

3 – D4 IT - B

# Code 1



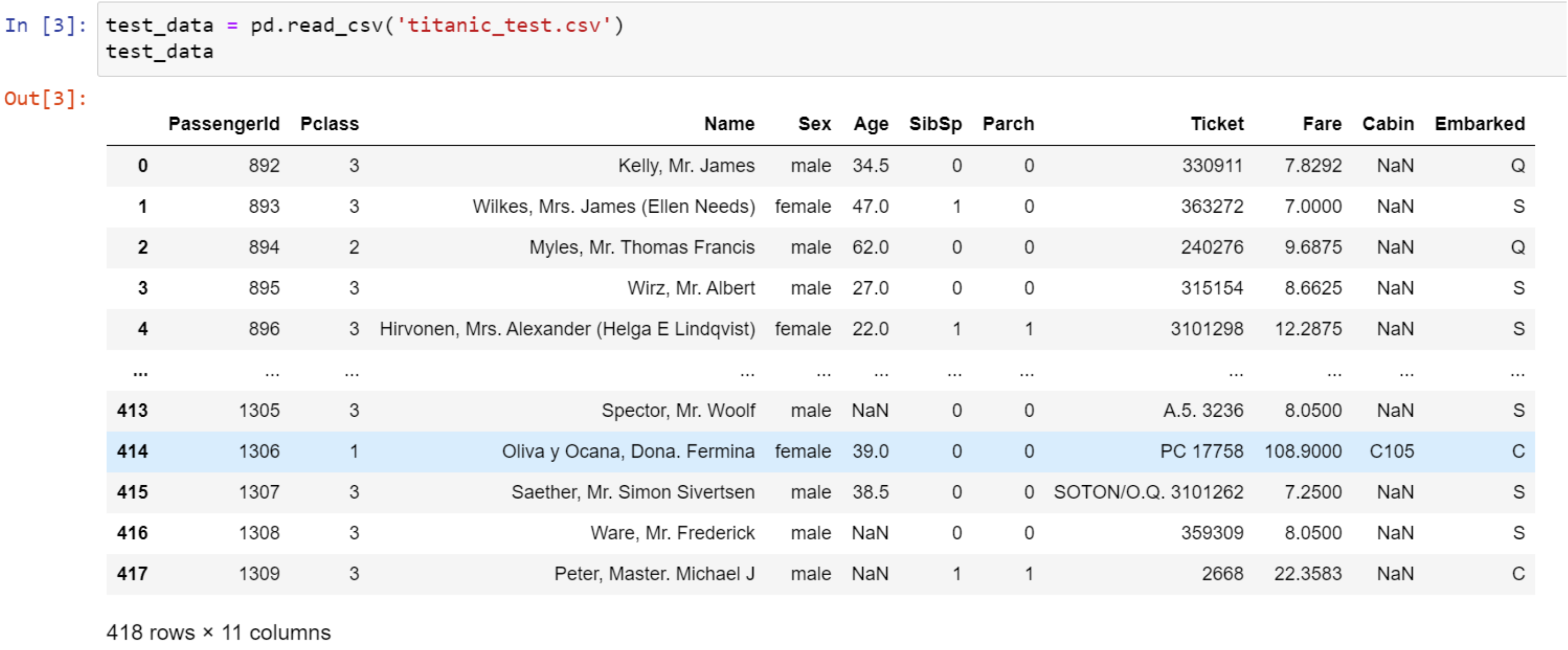
```
In [2]: dataset = pd.read_csv('titanic.csv')
        dataset
```

Out[2]:

| | PassengerId | Survived | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Cabin | Embarked |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 0 | 3 | Braund, Mr. Owen Harris | male | 22.0 | 1 | 0 | A/5 21171 | 7.2500 | NaN | S |
| **1** | 2 | 1 | 1 | Cumings, Mrs. John Bradley (Florence Briggs Th... | female | 38.0 | 1 | 0 | PC 17599 | 71.2833 | C85 | C |
| **2** | 3 | 1 | 3 | Heikkinen, Miss. Laina | female | 26.0 | 0 | 0 | STON/O2. 3101282 | 7.9250 | NaN | S |
| **3** | 4 | 1 | 1 | Futrelle, Mrs. Jacques Heath (Lily May Peel) | female | 35.0 | 1 | 0 | 113803 | 53.1000 | C123 | S |
| **4** | 5 | 0 | 3 | Allen, Mr. William Henry | male | 35.0 | 0 | 0 | 373450 | 8.0500 | NaN | S |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **886** | 887 | 0 | 2 | Montvila, Rev. Juozas | male | 27.0 | 0 | 0 | 211536 | 13.0000 | NaN | S |
| **887** | 888 | 1 | 1 | Graham, Miss. Margaret Edith | female | 19.0 | 0 | 0 | 112053 | 30.0000 | B42 | S |
| **888** | 889 | 0 | 3 | Johnston, Miss. Catherine Helen "Carrie" | female | NaN | 1 | 2 | W./C. 6607 | 23.4500 | NaN | S |
| **889** | 890 | 1 | 1 | Behr, Mr. Karl Howell | male | 26.0 | 0 | 0 | 111369 | 30.0000 | C148 | C |
| **890** | 891 | 0 | 3 | Dooley, Mr. Patrick | male | 32.0 | 0 | 0 | 370376 | 7.7500 | NaN | Q |

Membaca data csv titanic

# Code 2



```
In [3]: test_data = pd.read_csv('titanic_test.csv')
        test_data
```

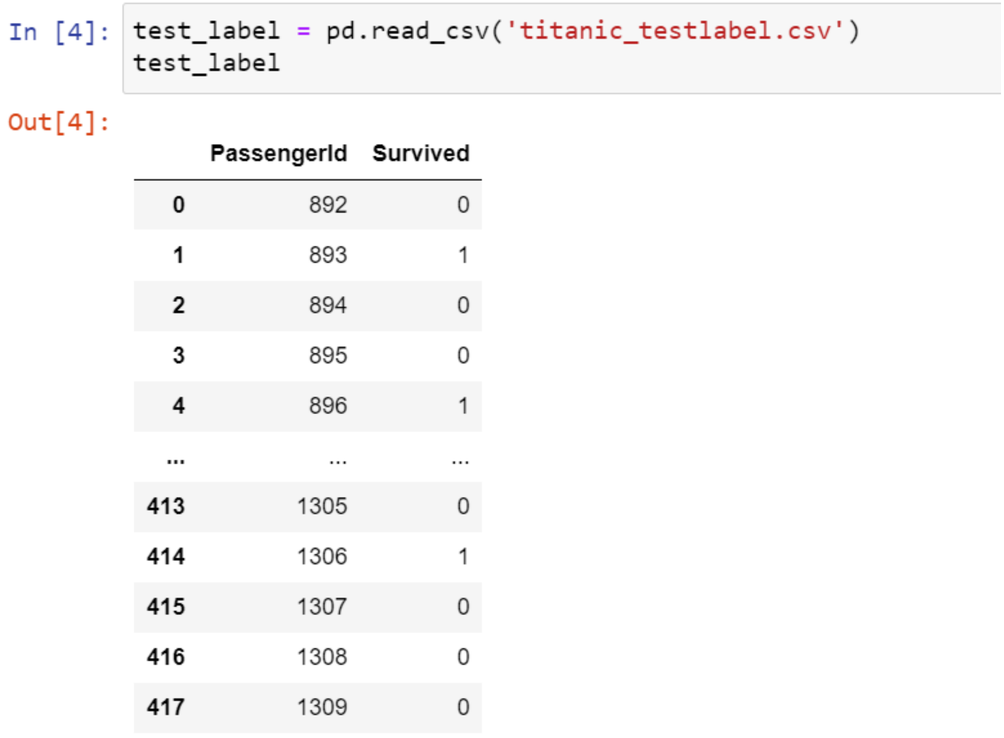
Out[3]:

| | PassengerId | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Cabin | Embarked |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 892 | 3 | Kelly, Mr. James | male | 34.5 | 0 | 0 | 330911 | 7.8292 | NaN | Q |
| **1** | 893 | 3 | Wilkes, Mrs. James (Ellen Needs) | female | 47.0 | 1 | 0 | 363272 | 7.0000 | NaN | S |
| **2** | 894 | 2 | Myles, Mr. Thomas Francis | male | 62.0 | 0 | 0 | 240276 | 9.6875 | NaN | Q |
| **3** | 895 | 3 | Wirz, Mr. Albert | male | 27.0 | 0 | 0 | 315154 | 8.6625 | NaN | S |
| **4** | 896 | 3 | Hirvonen, Mrs. Alexander (Helga E Lindqvist) | female | 22.0 | 1 | 1 | 3101298 | 12.2875 | NaN | S |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **413** | 1305 | 3 | Spector, Mr. Woolf | male | NaN | 0 | 0 | A.5. 3236 | 8.0500 | NaN | S |
| **414** | 1306 | 1 | Oliva y Ocana, Dona. Fermina | female | 39.0 | 0 | 0 | PC 17758 | 108.9000 | C105 | C |
| **415** | 1307 | 3 | Saether, Mr. Simon Sivertsen | male | 38.5 | 0 | 0 | SOTON/O.Q. 3101262 | 7.2500 | NaN | S |
| **416** | 1308 | 3 | Ware, Mr. Frederick | male | NaN | 0 | 0 | 359309 | 8.0500 | NaN | S |
| **417** | 1309 | 3 | Peter, Master. Michael J | male | NaN | 1 | 1 | 2668 | 22.3583 | NaN | C |

418 rows × 11 columns

Membaca data csv titanic_test

# Code 3



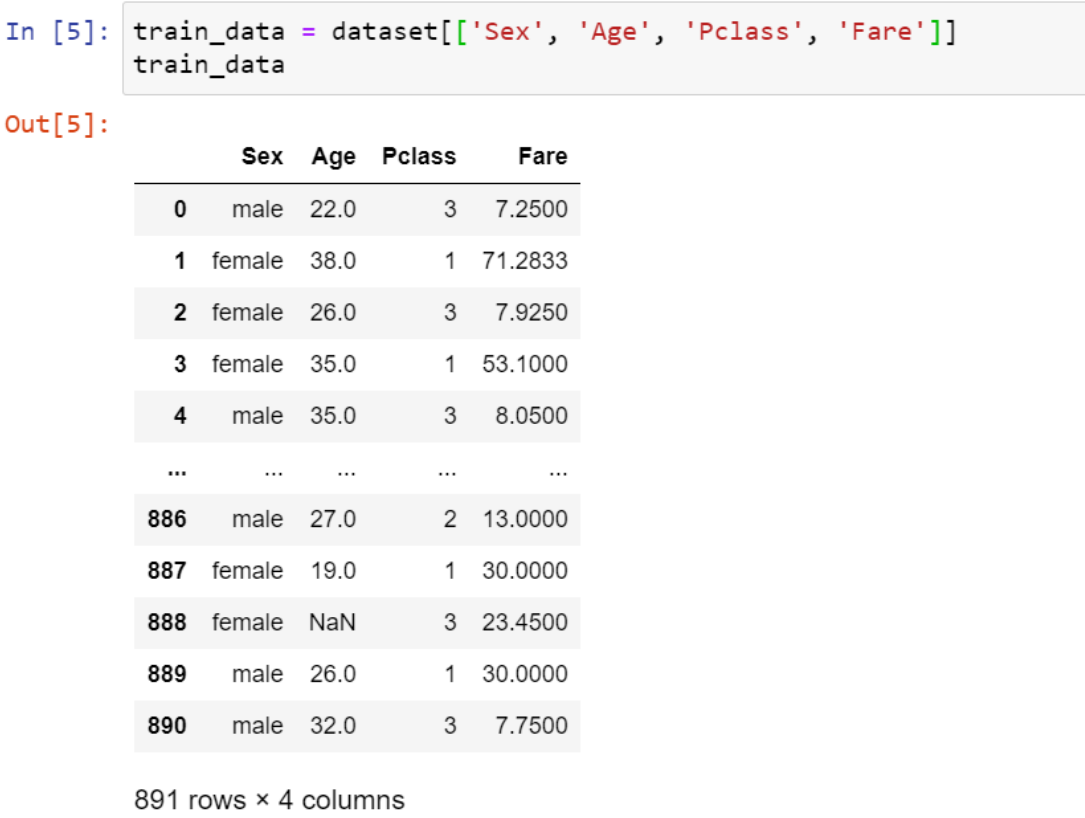
In [4]:
```
test_label = pd.read_csv('titanic_testlabel.csv')
test_label
```

Out[4]:

| | PassengerId | Survived |
|---|---|---|
| **0** | 892 | 0 |
| **1** | 893 | 1 |
| **2** | 894 | 0 |
| **3** | 895 | 0 |
| **4** | 896 | 1 |
| **...** | ... | ... |
| **413** | 1305 | 0 |
| **414** | 1306 | 1 |
| **415** | 1307 | 0 |
| **416** | 1308 | 0 |
| **417** | 1309 | 0 |

Membaca data csv titanic_testlabel

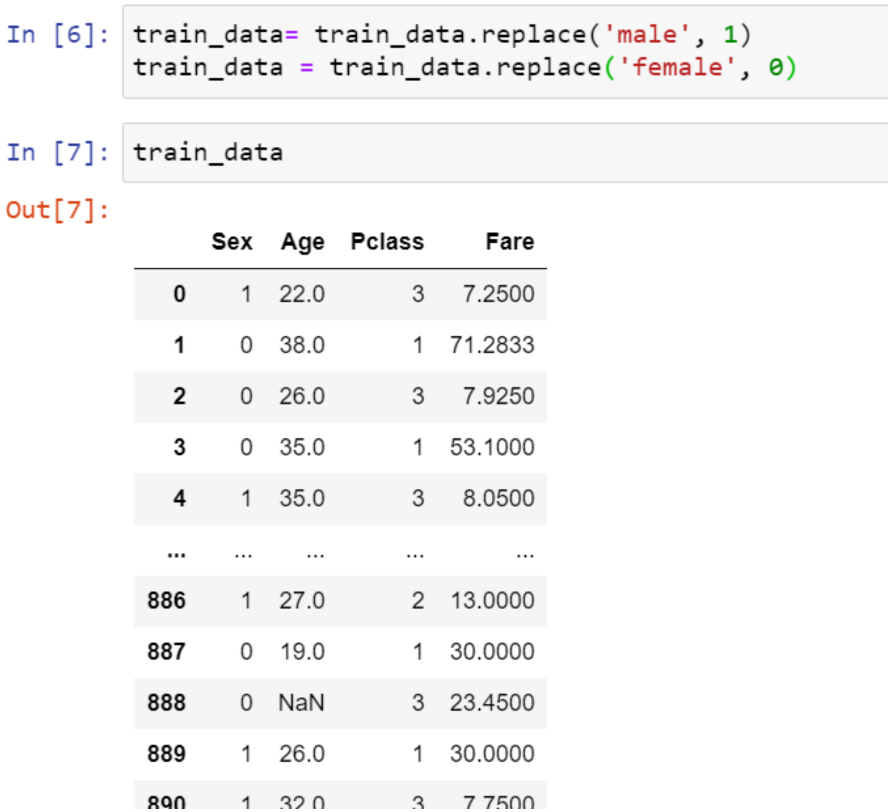
# Code 4



```
In [5]: train_data = dataset[['Sex', 'Age', 'Pclass', 'Fare']]
        train_data

Out[5]:
              Sex   Age  Pclass     Fare
        0    male  22.0       3   7.2500
        1  female  38.0       1  71.2833
        2  female  26.0       3   7.9250
        3  female  35.0       1  53.1000
        4    male  35.0       3   8.0500
      ...     ...   ...     ...      ...
      886    male  27.0       2  13.0000
      887  female  19.0       1  30.0000
      888  female   NaN       3  23.4500
      889    male  26.0       1  30.0000
      890    male  32.0       3   7.7500

     891 rows × 4 columns
```

Mengambil data csv titanic kolom sex age pclass fare

# Code 5



In [6]:
```python
train_data= train_data.replace('male', 1)
train_data = train_data.replace('female', 0)
```

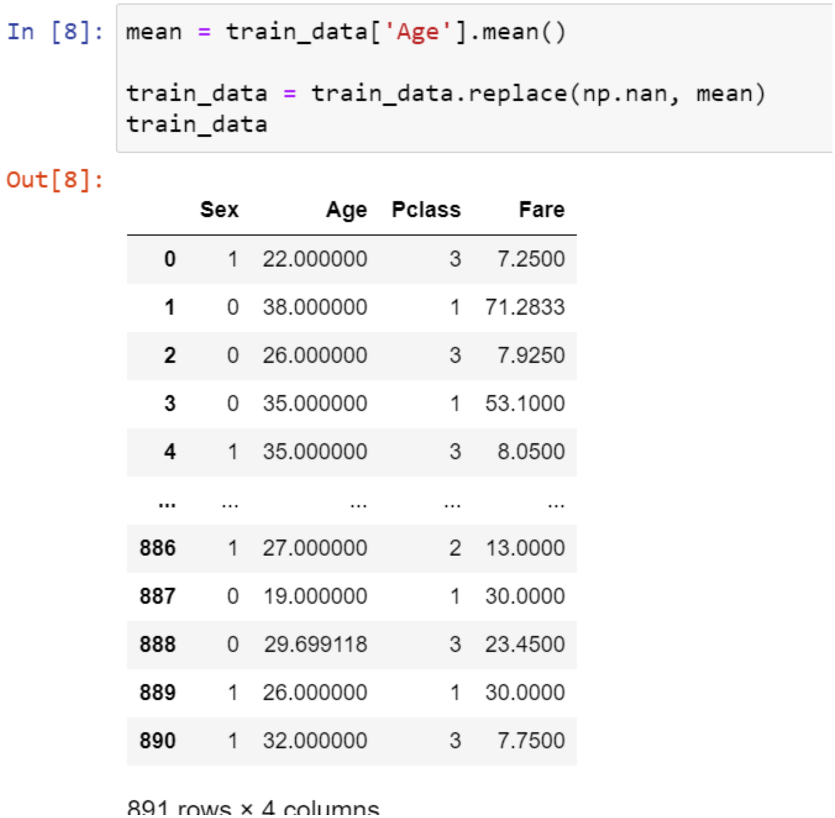
In [7]:
```python
train_data
```

Out[7]:

|  | Sex | Age | Pclass | Fare |
|---|---|---|---|---|
| **0** | 1 | 22.0 | 3 | 7.2500 |
| **1** | 0 | 38.0 | 1 | 71.2833 |
| **2** | 0 | 26.0 | 3 | 7.9250 |
| **3** | 0 | 35.0 | 1 | 53.1000 |
| **4** | 1 | 35.0 | 3 | 8.0500 |
| **...** | ... | ... | ... | ... |
| **886** | 1 | 27.0 | 2 | 13.0000 |
| **887** | 0 | 19.0 | 1 | 30.0000 |
| **888** | 0 | NaN | 3 | 23.4500 |
| **889** | 1 | 26.0 | 1 | 30.0000 |
| **890** | 1 | 32.0 | 3 | 7.7500 |

Mengubah male dan female menjadi 1 dan 0

# Code 6



In [8]:

```
mean = train_data['Age'].mean()

train_data = train_data.replace(np.nan, mean)
train_data
```

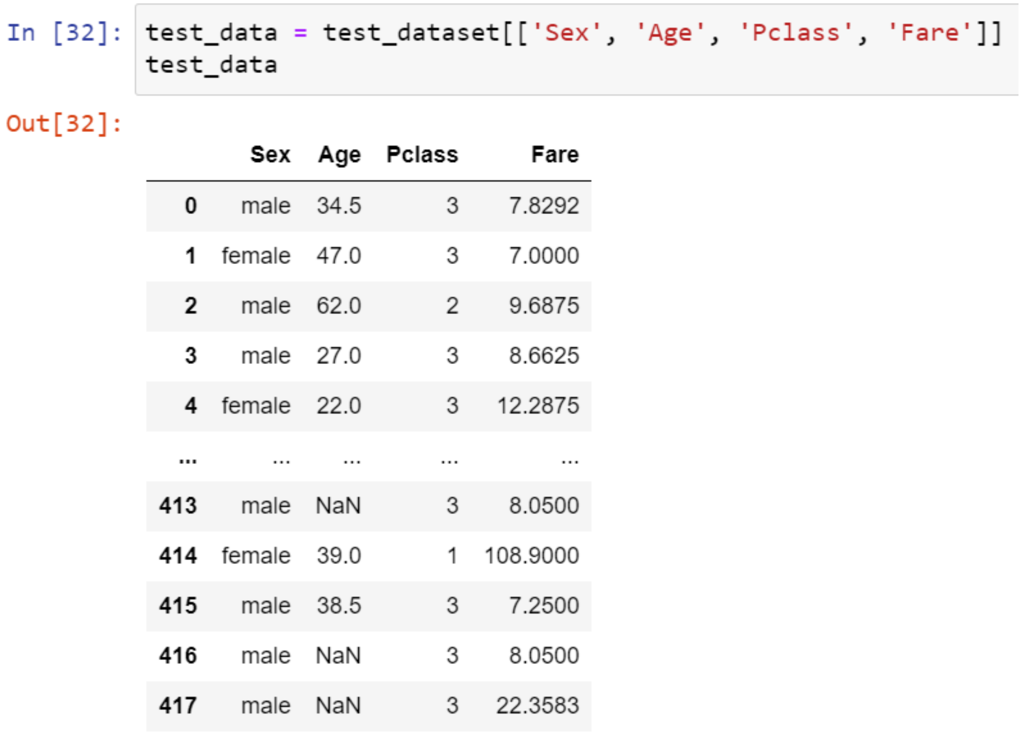
Out[8]:

| | Sex | Age | Pclass | Fare |
|---|---|---|---|---|
| **0** | 1 | 22.000000 | 3 | 7.2500 |
| **1** | 0 | 38.000000 | 1 | 71.2833 |
| **2** | 0 | 26.000000 | 3 | 7.9250 |
| **3** | 0 | 35.000000 | 1 | 53.1000 |
| **4** | 1 | 35.000000 | 3 | 8.0500 |
| **...** | ... | ... | ... | ... |
| **886** | 1 | 27.000000 | 2 | 13.0000 |
| **887** | 0 | 19.000000 | 1 | 30.0000 |
| **888** | 0 | 29.699118 | 3 | 23.4500 |
| **889** | 1 | 26.000000 | 1 | 30.0000 |
| **890** | 1 | 32.000000 | 3 | 7.7500 |

891 rows × 4 columns

Mengisi missing value pada kolom age dengan rata2 kolom age

# Code 7

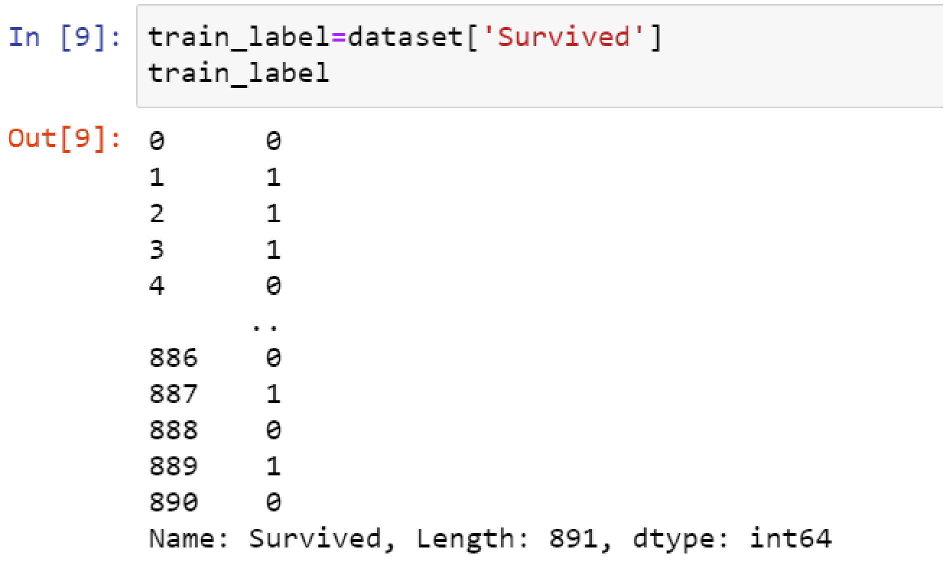


```
In [32]: test_data = test_dataset[['Sex', 'Age', 'Pclass', 'Fare']]
         test_data
```

```
Out[32]:
              Sex   Age  Pclass      Fare
         0   male  34.5       3    7.8292
         1 female  47.0       3    7.0000
         2   male  62.0       2    9.6875
         3   male  27.0       3    8.6625
         4 female  22.0       3   12.2875
       ...    ...   ...     ...       ...
       413   male   NaN       3    8.0500
       414 female  39.0       1  108.9000
       415   male  38.5       3    7.2500
       416   male   NaN       3    8.0500
       417   male   NaN       3   22.3583
```

Mengambil test_dataset kolom sex age pclass fare

# Code 8

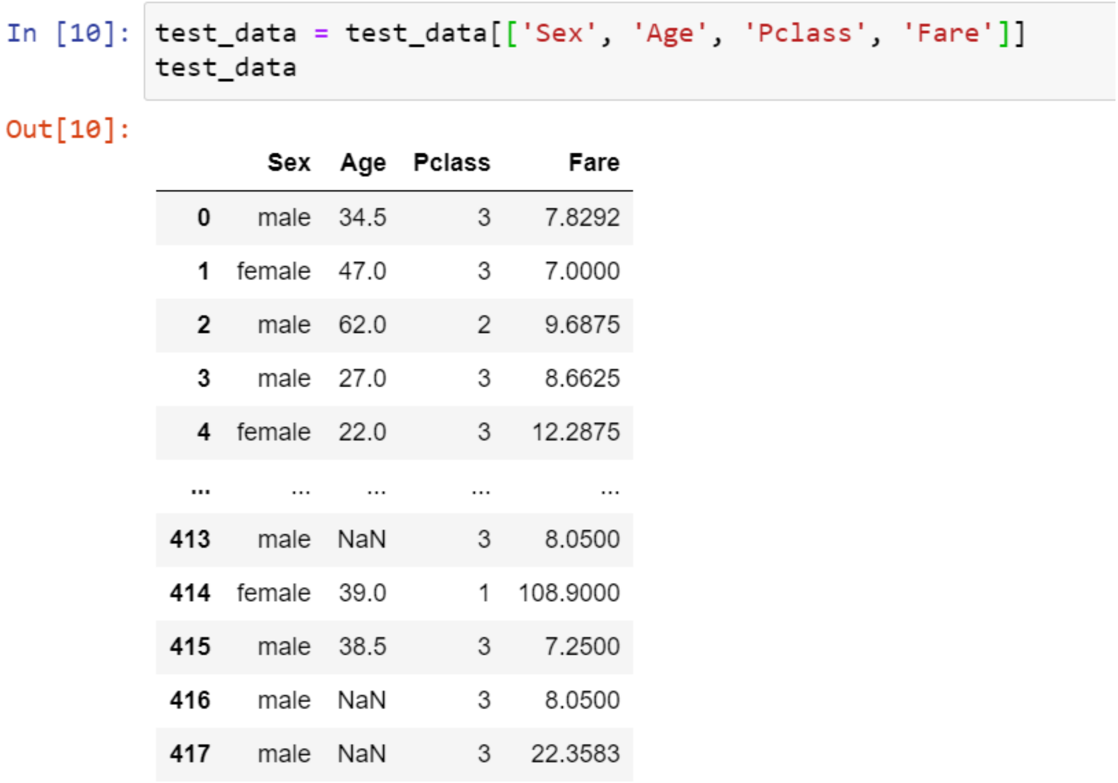


```
In [9]: train_label=dataset['Survived']
        train_label
```

```
Out[9]: 0      0
        1      1
        2      1
        3      1
        4      0
              ..
        886    0
        887    1
        888    0
        889    1
        890    0
        Name: Survived, Length: 891, dtype: int64
```

Mengambil train_label, kolom survived

# Code 9



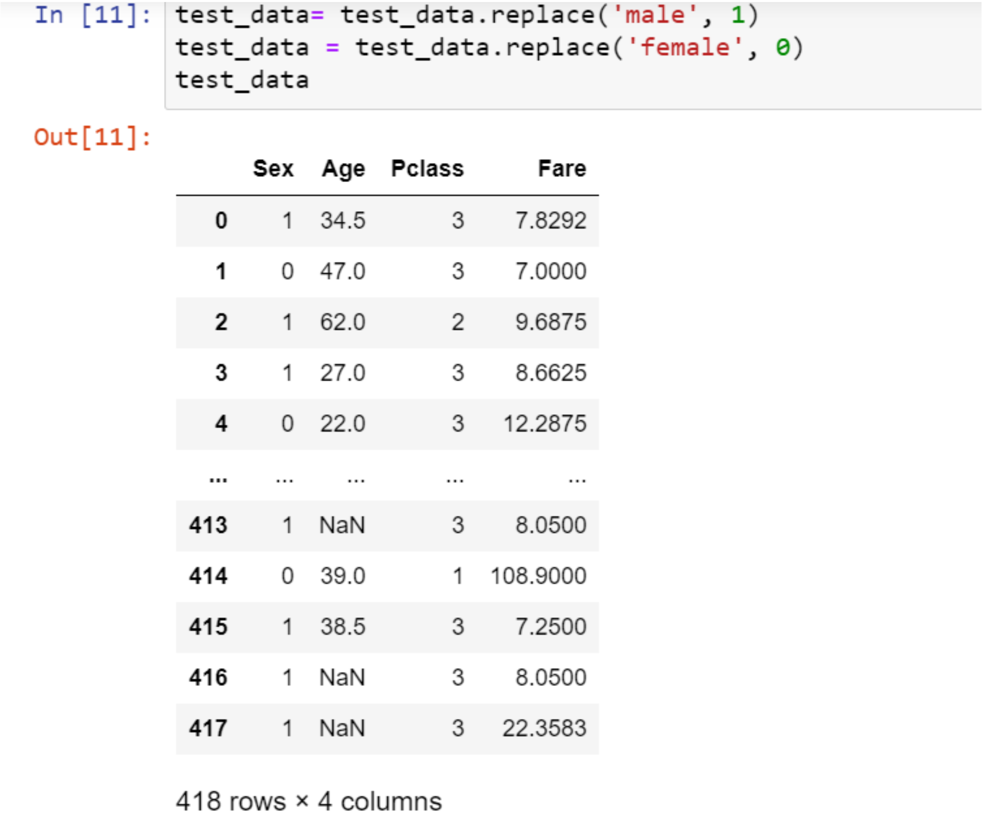
```
In [10]: test_data = test_data[['Sex', 'Age', 'Pclass', 'Fare']]
         test_data
```

Out[10]:

| | Sex | Age | Pclass | Fare |
|---|---|---|---|---|
| **0** | male | 34.5 | 3 | 7.8292 |
| **1** | female | 47.0 | 3 | 7.0000 |
| **2** | male | 62.0 | 2 | 9.6875 |
| **3** | male | 27.0 | 3 | 8.6625 |
| **4** | female | 22.0 | 3 | 12.2875 |
| **...** | ... | ... | ... | ... |
| **413** | male | NaN | 3 | 8.0500 |
| **414** | female | 39.0 | 1 | 108.9000 |
| **415** | male | 38.5 | 3 | 7.2500 |
| **416** | male | NaN | 3 | 8.0500 |
| **417** | male | NaN | 3 | 22.3583 |

Mengambil test data fitur sex age pclass fare

# Code 10



```
In [11]: test_data= test_data.replace('male', 1)
         test_data = test_data.replace('female', 0)
         test_data
```

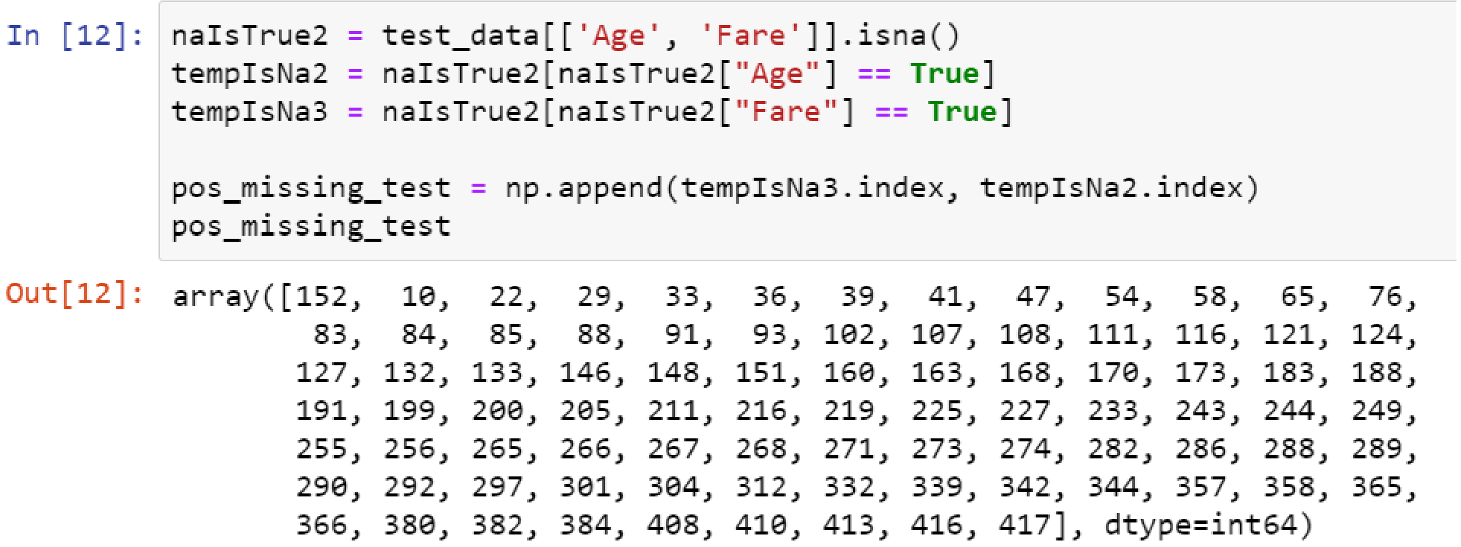
Out[11]:

|  | Sex | Age | Pclass | Fare |
|---|---|---|---|---|
| **0** | 1 | 34.5 | 3 | 7.8292 |
| **1** | 0 | 47.0 | 3 | 7.0000 |
| **2** | 1 | 62.0 | 2 | 9.6875 |
| **3** | 1 | 27.0 | 3 | 8.6625 |
| **4** | 0 | 22.0 | 3 | 12.2875 |
| **...** | ... | ... | ... | ... |
| **413** | 1 | NaN | 3 | 8.0500 |
| **414** | 0 | 39.0 | 1 | 108.9000 |
| **415** | 1 | 38.5 | 3 | 7.2500 |
| **416** | 1 | NaN | 3 | 8.0500 |
| **417** | 1 | NaN | 3 | 22.3583 |

418 rows × 4 columns

Mengubah fitur male female menjadi 1 dan 0

# Code 11

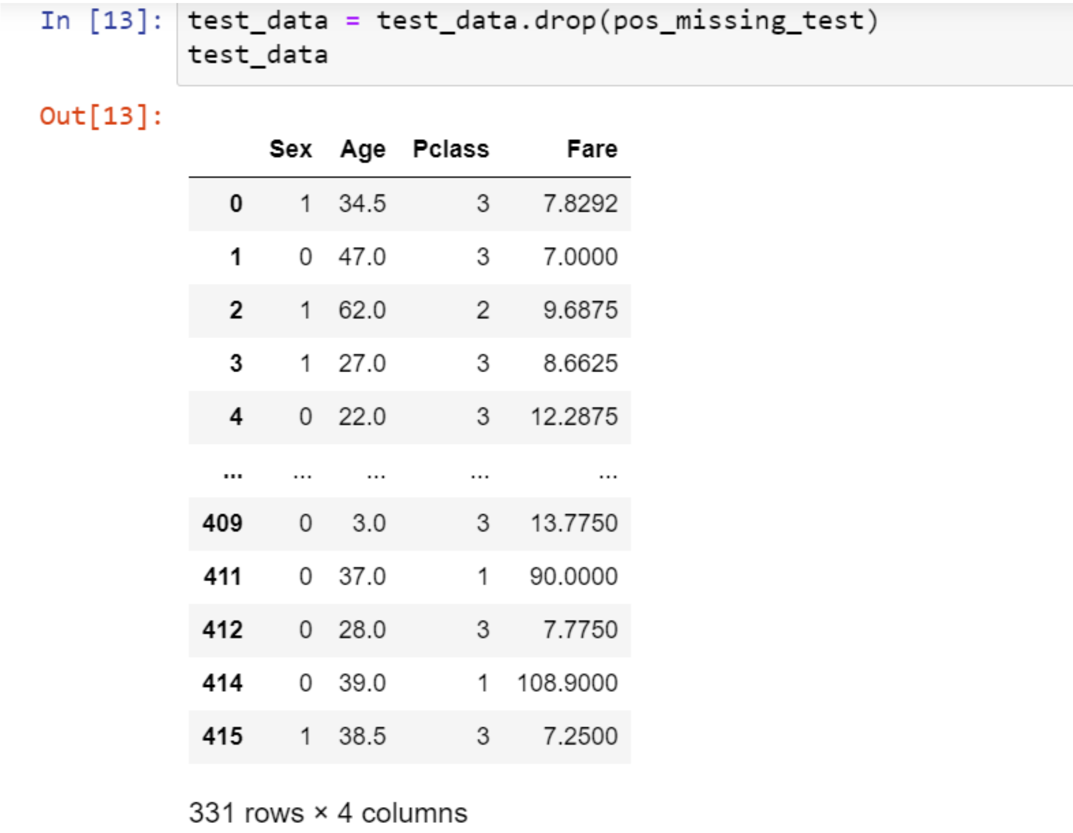


```
In [12]: naIsTrue2 = test_data[['Age', 'Fare']].isna()
         tempIsNa2 = naIsTrue2[naIsTrue2["Age"] == True]
         tempIsNa3 = naIsTrue2[naIsTrue2["Fare"] == True]

         pos_missing_test = np.append(tempIsNa3.index, tempIsNa2.index)
         pos_missing_test
```

```
Out[12]: array([152,  10,  22,  29,  33,  36,  39,  41,  47,  54,  58,  65,  76,
                 83,  84,  85,  88,  91,  93, 102, 107, 108, 111, 116, 121, 124,
                127, 132, 133, 146, 148, 151, 160, 163, 168, 170, 173, 183, 188,
                191, 199, 200, 205, 211, 216, 219, 225, 227, 233, 243, 244, 249,
                255, 256, 265, 266, 267, 268, 271, 273, 274, 282, 286, 288, 289,
                290, 292, 297, 301, 304, 312, 332, 339, 342, 344, 357, 358, 365,
                366, 380, 382, 384, 408, 410, 413, 416, 417], dtype=int64)
```

Mengambil posisi index missing value pada test data

# Code 12

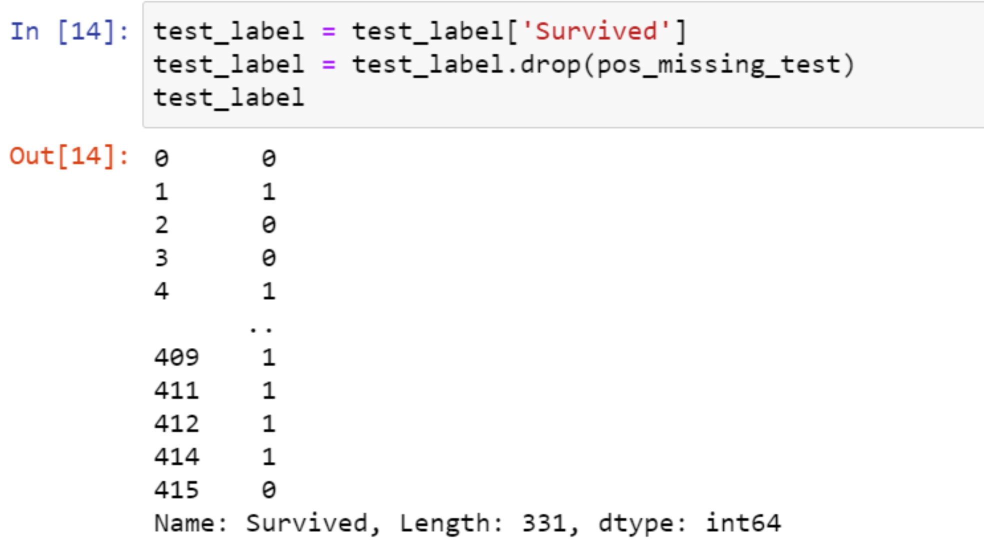


```
In [13]: test_data = test_data.drop(pos_missing_test)
         test_data
```

```
Out[13]:
              Sex   Age  Pclass      Fare
         0      1  34.5       3    7.8292
         1      0  47.0       3    7.0000
         2      1  62.0       2    9.6875
         3      1  27.0       3    8.6625
         4      0  22.0       3   12.2875
       ...    ...   ...     ...       ...
       409      0   3.0       3   13.7750
       411      0  37.0       1   90.0000
       412      0  28.0       3    7.7750
       414      0  39.0       1  108.9000
       415      1  38.5       3    7.2500

       331 rows × 4 columns
```

Menghapus data test yang memiliki missing value

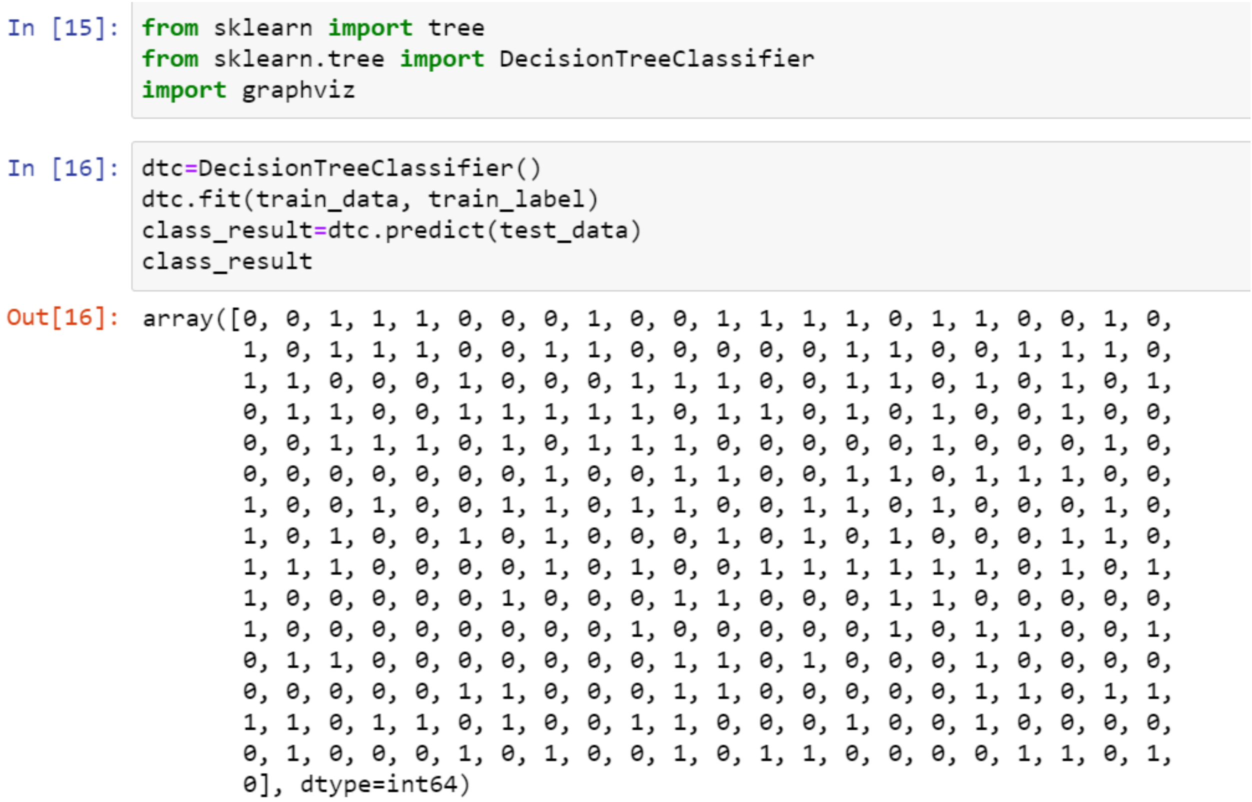
# Code 13



```
In [14]: test_label = test_label['Survived']
         test_label = test_label.drop(pos_missing_test)
         test_label

Out[14]: 0      0
         1      1
         2      0
         3      0
         4      1
               ..
         409    1
         411    1
         412    1
         414    1
         415    0
         Name: Survived, Length: 331, dtype: int64
```

Menghapus data pada testlabel yang memiliki missing value

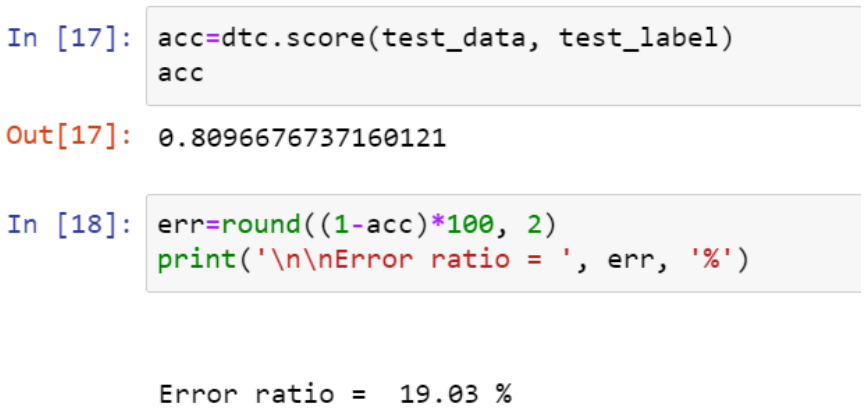
# Code 14



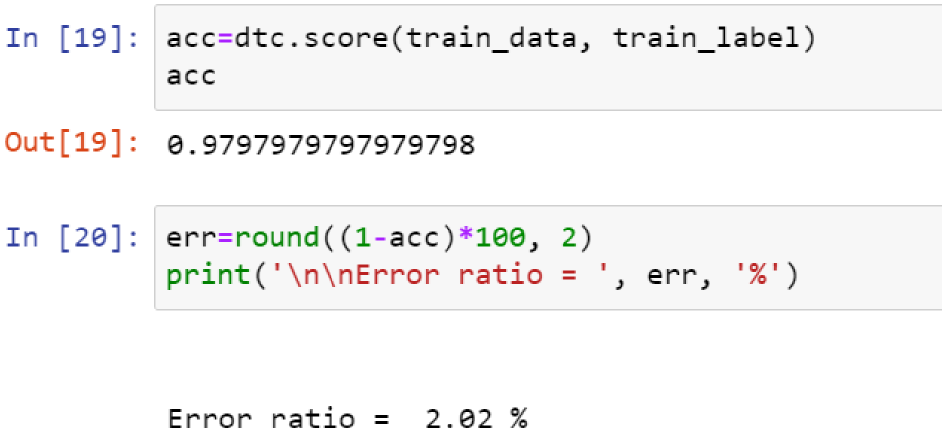
```
In [15]: from sklearn import tree
         from sklearn.tree import DecisionTreeClassifier
         import graphviz

In [16]: dtc=DecisionTreeClassifier()
         dtc.fit(train_data, train_label)
         class_result=dtc.predict(test_data)
         class_result

Out[16]: array([0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0,
                1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0,
                1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1,
                0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0,
                0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0,
                0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0,
                1, 0, 0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0,
                1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0,
                1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1,
                1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0,
                1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1,
                0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1,
                1, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0,
                0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1,
                0], dtype=int64)
```

Membuat model decision tree menggunakan library sklearn

# Code 15



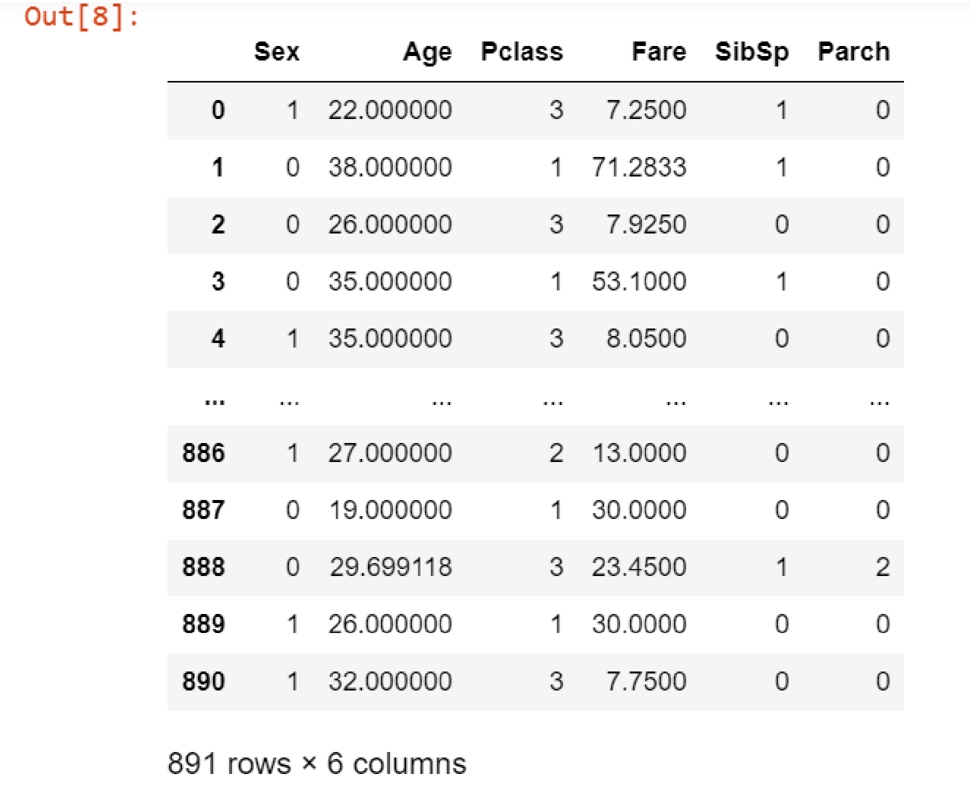
```
In [17]: acc=dtc.score(test_data, test_label)
         acc

Out[17]: 0.8096676737160121

In [18]: err=round((1-acc)*100, 2)
         print('\n\nError ratio = ', err, '%')


         Error ratio =  19.03 %
```



```
In [19]: acc=dtc.score(train_data, train_label)
         acc

Out[19]: 0.9797979797979798

In [20]: err=round((1-acc)*100, 2)
         print('\n\nError ratio = ', err, '%')


         Error ratio =  2.02 %
```

Menghitung nilai akurasi dan error rasio pada test data dan train data

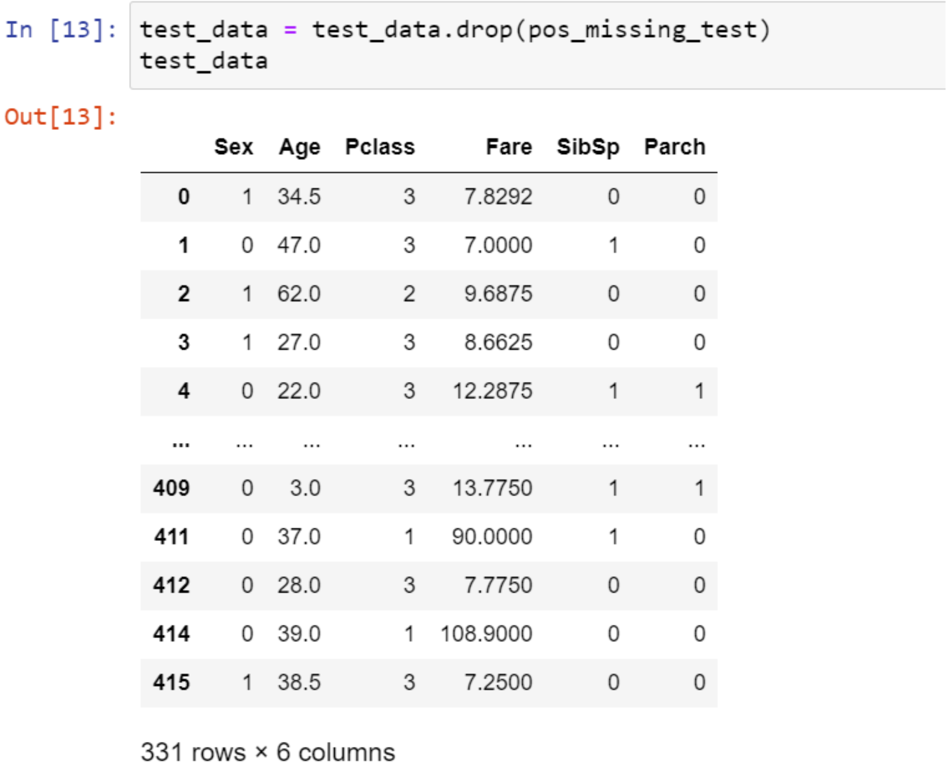
# Code 16



Out[8]:

| | Sex | Age | Pclass | Fare | SibSp | Parch |
|---|---|---|---|---|---|---|
| **0** | 1 | 22.000000 | 3 | 7.2500 | 1 | 0 |
| **1** | 0 | 38.000000 | 1 | 71.2833 | 1 | 0 |
| **2** | 0 | 26.000000 | 3 | 7.9250 | 0 | 0 |
| **3** | 0 | 35.000000 | 1 | 53.1000 | 1 | 0 |
| **4** | 1 | 35.000000 | 3 | 8.0500 | 0 | 0 |
| **...** | ... | ... | ... | ... | ... | ... |
| **886** | 1 | 27.000000 | 2 | 13.0000 | 0 | 0 |
| **887** | 0 | 19.000000 | 1 | 30.0000 | 0 | 0 |
| **888** | 0 | 29.699118 | 3 | 23.4500 | 1 | 2 |
| **889** | 1 | 26.000000 | 1 | 30.0000 | 0 | 0 |
| **890** | 1 | 32.000000 | 3 | 7.7500 | 0 | 0 |

891 rows × 6 columns



In [13]:
```
test_data = test_data.drop(pos_missing_test)
test_data
```

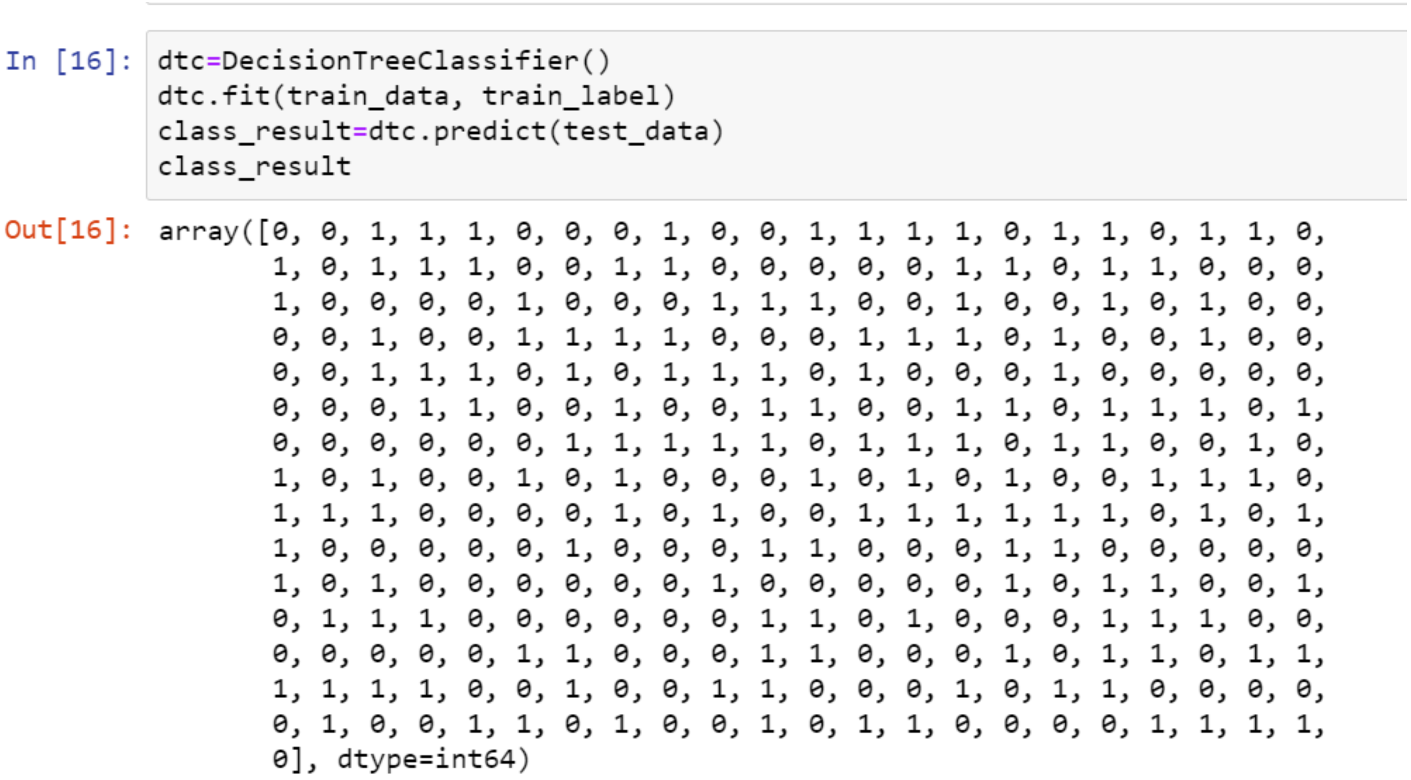
Out[13]:

| | Sex | Age | Pclass | Fare | SibSp | Parch |
|---|---|---|---|---|---|---|
| **0** | 1 | 34.5 | 3 | 7.8292 | 0 | 0 |
| **1** | 0 | 47.0 | 3 | 7.0000 | 1 | 0 |
| **2** | 1 | 62.0 | 2 | 9.6875 | 0 | 0 |
| **3** | 1 | 27.0 | 3 | 8.6625 | 0 | 0 |
| **4** | 0 | 22.0 | 3 | 12.2875 | 1 | 1 |
| **...** | ... | ... | ... | ... | ... | ... |
| **409** | 0 | 3.0 | 3 | 13.7750 | 1 | 1 |
| **411** | 0 | 37.0 | 1 | 90.0000 | 1 | 0 |
| **412** | 0 | 28.0 | 3 | 7.7750 | 0 | 0 |
| **414** | 0 | 39.0 | 1 | 108.9000 | 0 | 0 |
| **415** | 1 | 38.5 | 3 | 7.2500 | 0 | 0 |

331 rows × 6 columns

Menambahkan fitur sibsp dan parch

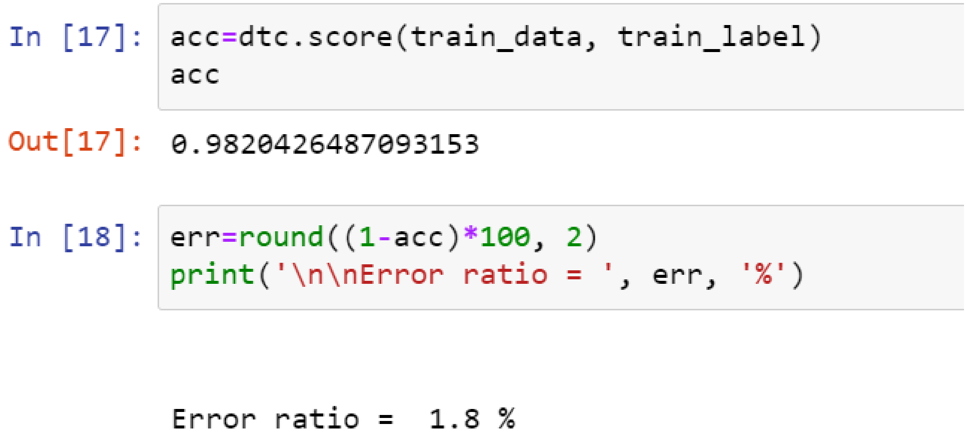
# Code 17



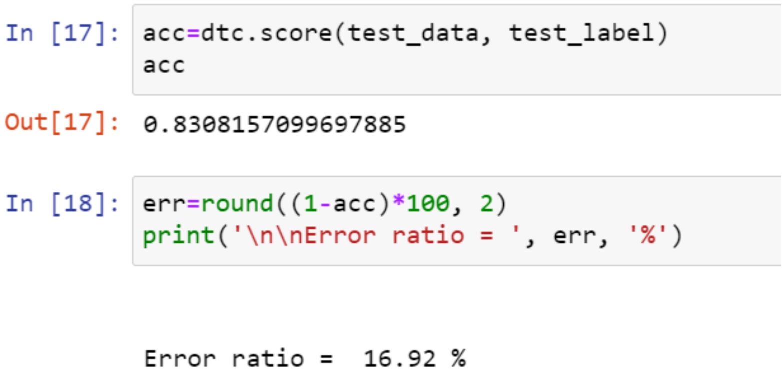
```
In [16]: dtc=DecisionTreeClassifier()
         dtc.fit(train_data, train_label)
         class_result=dtc.predict(test_data)
         class_result

Out[16]: array([0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0,
                1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0,
                1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0,
                0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 0,
                0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0,
                0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1,
                0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0,
                1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0,
                1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1,
                1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0,
                1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1,
                0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0,
                0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1,
                1, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0,
                0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1,
                0], dtype=int64)
```

Membuat ulang model dan prediksi dengan tambahan fitur baru

# Code 18



```
In [17]: acc=dtc.score(train_data, train_label)
         acc

Out[17]: 0.9820426487093153

In [18]: err=round((1-acc)*100, 2)
         print('\n\nError ratio = ', err, '%')


         Error ratio =  1.8 %
```



```
In [17]: acc=dtc.score(test_data, test_label)
         acc

Out[17]: 0.8308157099697885

In [18]: err=round((1-acc)*100, 2)
         print('\n\nError ratio = ', err, '%')


         Error ratio =  16.92 %
```

Dapat kita ketahui dengan menambahkan fitur kita memperoleh penurunan error ratio