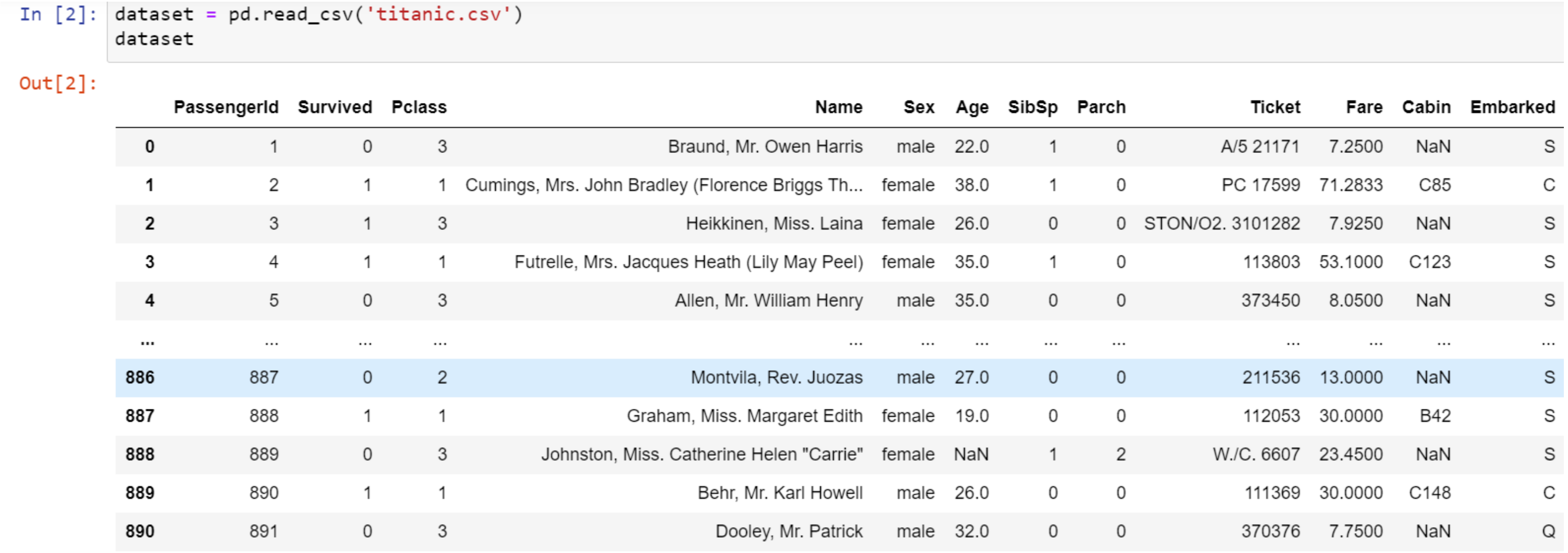
Tugas Materi 8

Zul Fauzi Oktaviansyah

2110181056

3 – D4 IT - B

Code 1



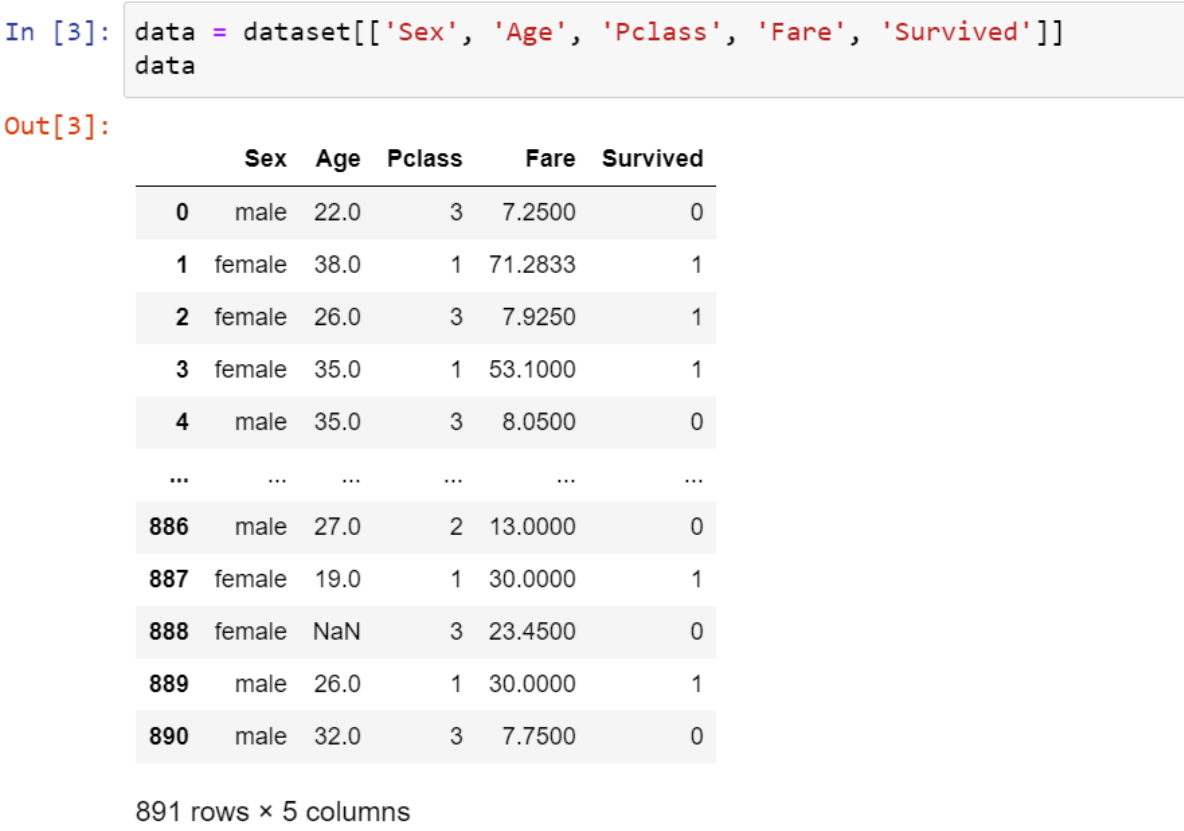
```
In [2]: dataset = pd.read_csv('titanic.csv')
        dataset
```

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]:

```
data = dataset[['Sex', 'Age', 'Pclass', 'Fare', 'Survived']]
data
```

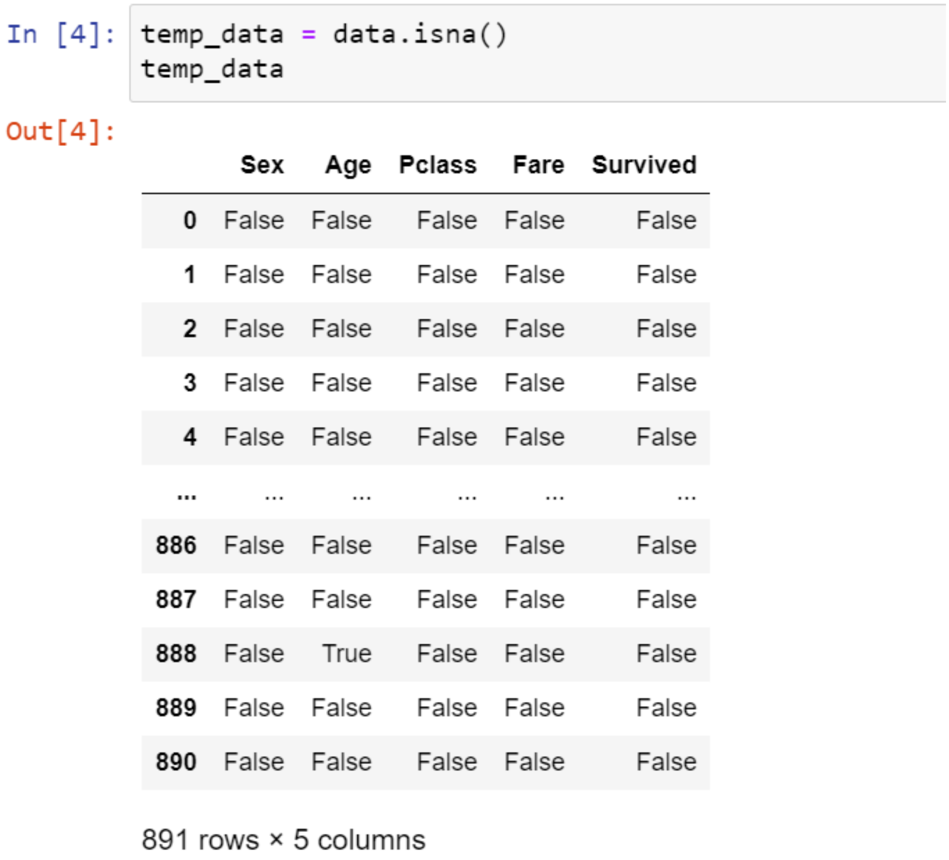
Out[3]:

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

891 rows × 5 columns

Mengambil dataset fitur sex, age, pclass, fare, survived

Code 3



```
In [4]: temp_data = data.isna()
        temp_data
```

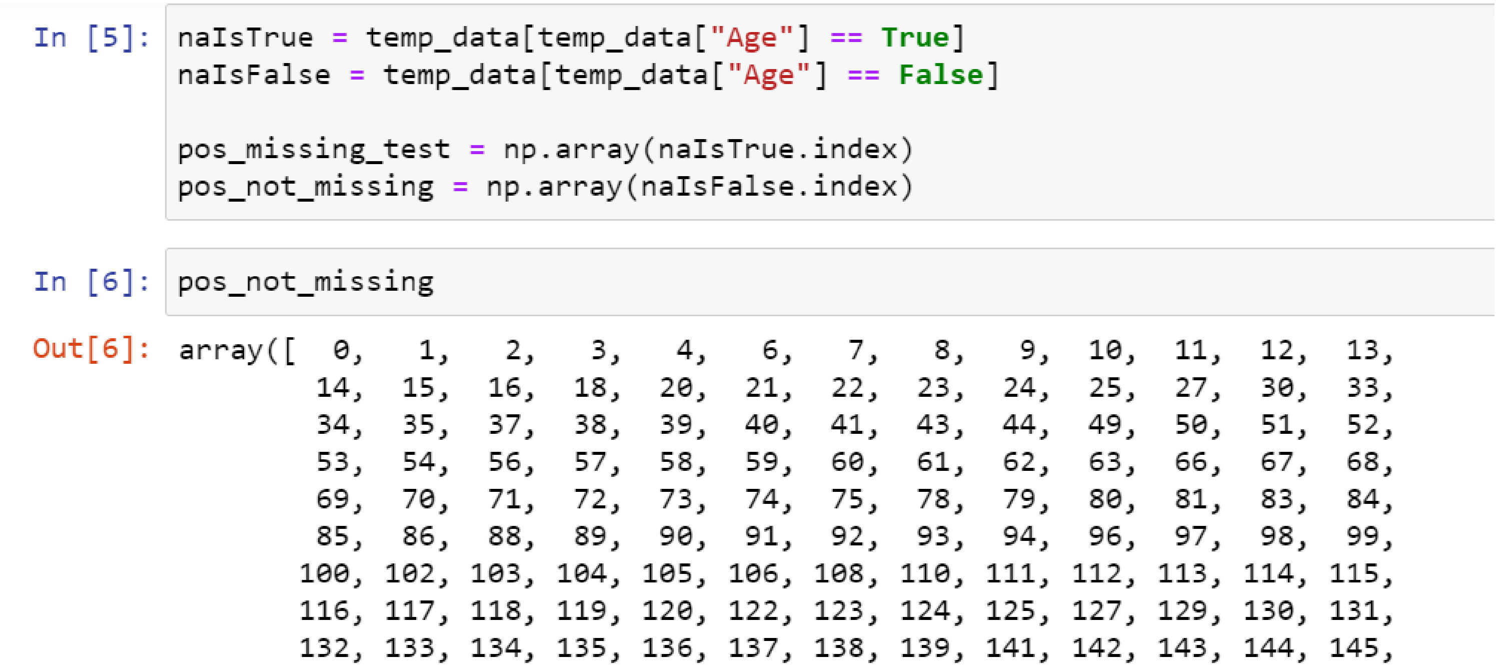
Out[4]:

	Sex	Age	Pclass	Fare	Survived
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...	...	...	...	...	...
886	False	False	False	False	False
887	False	False	False	False	False
888	False	True	False	False	False
889	False	False	False	False	False
890	False	False	False	False	False

891 rows × 5 columns

Membuat value missing menjadi true dan not missing menjadi false

Code 4



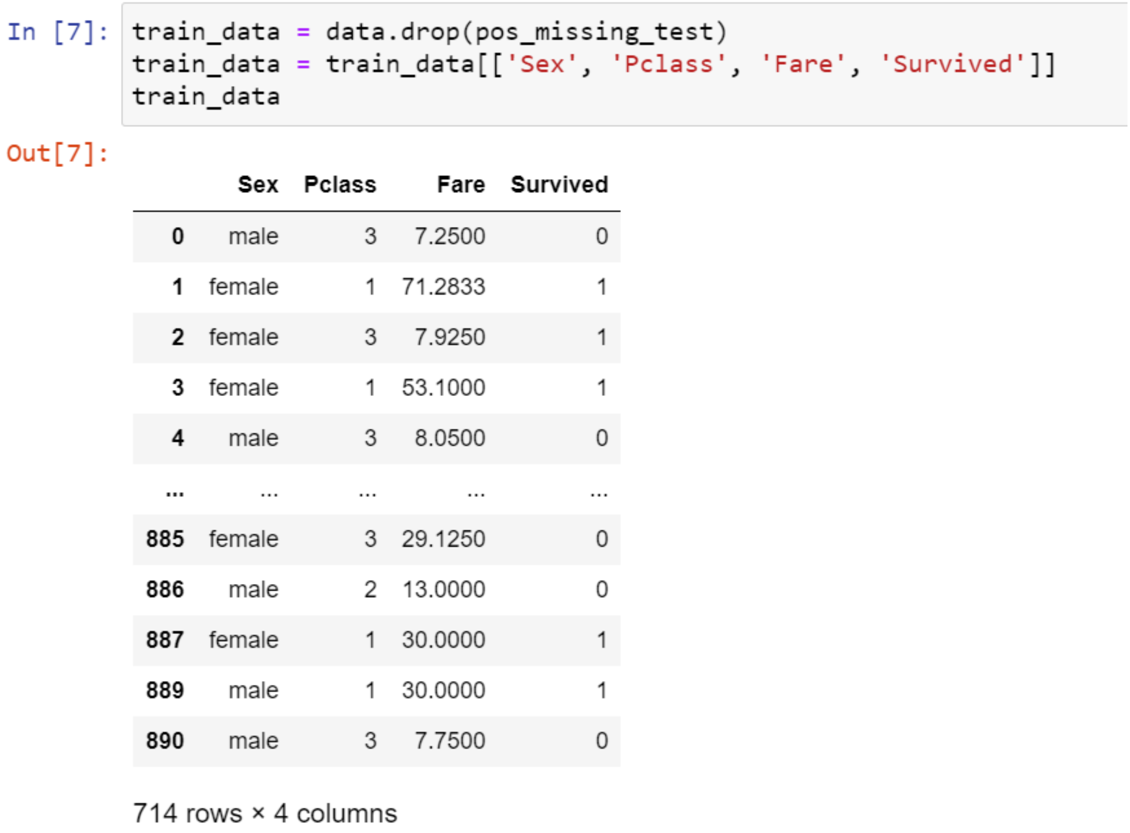
```
In [5]: naIsTrue = temp_data[temp_data["Age"] == True]
        naIsFalse = temp_data[temp_data["Age"] == False]

        pos_missing_test = np.array(naIsTrue.index)
        pos_not_missing = np.array(naIsFalse.index)
```

```
In [6]: pos_not_missing
```

```
Out[6]: array([  0,   1,   2,   3,   4,   6,   7,   8,   9,  10,  11,  12,  13,
                14,  15,  16,  18,  20,  21,  22,  23,  24,  25,  27,  30,  33,
                34,  35,  37,  38,  39,  40,  41,  43,  44,  49,  50,  51,  52,
                53,  54,  56,  57,  58,  59,  60,  61,  62,  63,  66,  67,  68,
                69,  70,  71,  72,  73,  74,  75,  78,  79,  80,  81,  83,  84,
                85,  86,  88,  89,  90,  91,  92,  93,  94,  96,  97,  98,  99,
               100, 102, 103, 104, 105, 106, 108, 110, 111, 112, 113, 114, 115,
               116, 117, 118, 119, 120, 122, 123, 124, 125, 127, 129, 130, 131,
               132, 133, 134, 135, 136, 137, 138, 139, 141, 142, 143, 144, 145,
```

Mengambil posisi index dari data fitur age yang missing dan yang tidak

Code 5



In [7]:

```
train_data = data.drop(pos_missing_test)
train_data = train_data[['Sex', 'Pclass', 'Fare', 'Survived']]
train_data
```

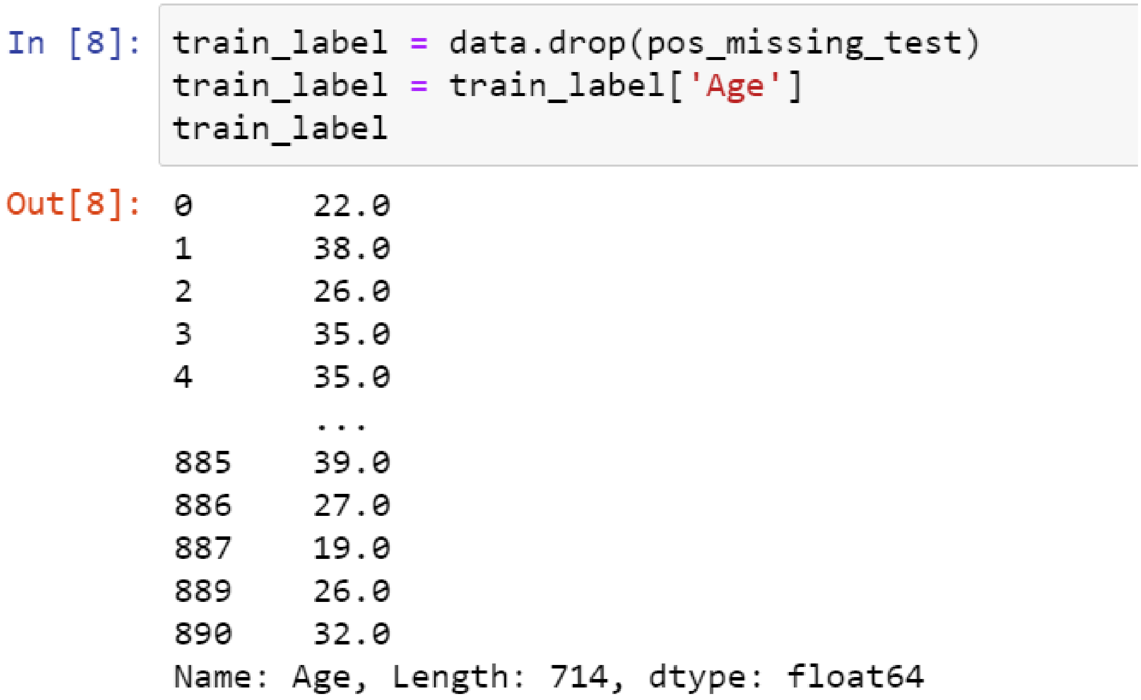
Out[7]:

	Sex	Pclass	Fare	Survived
0	male	3	7.2500	0
1	female	1	71.2833	1
2	female	3	7.9250	1
3	female	1	53.1000	1
4	male	3	8.0500	0
...	...	...	...	...
885	female	3	29.1250	0
886	male	2	13.0000	0
887	female	1	30.0000	1
889	male	1	30.0000	1
890	male	3	7.7500	0

714 rows × 4 columns

Data yang tidak missing pada fitur age dijadikan sebagai train data
Terdiri atas fitur selain age

Code 6

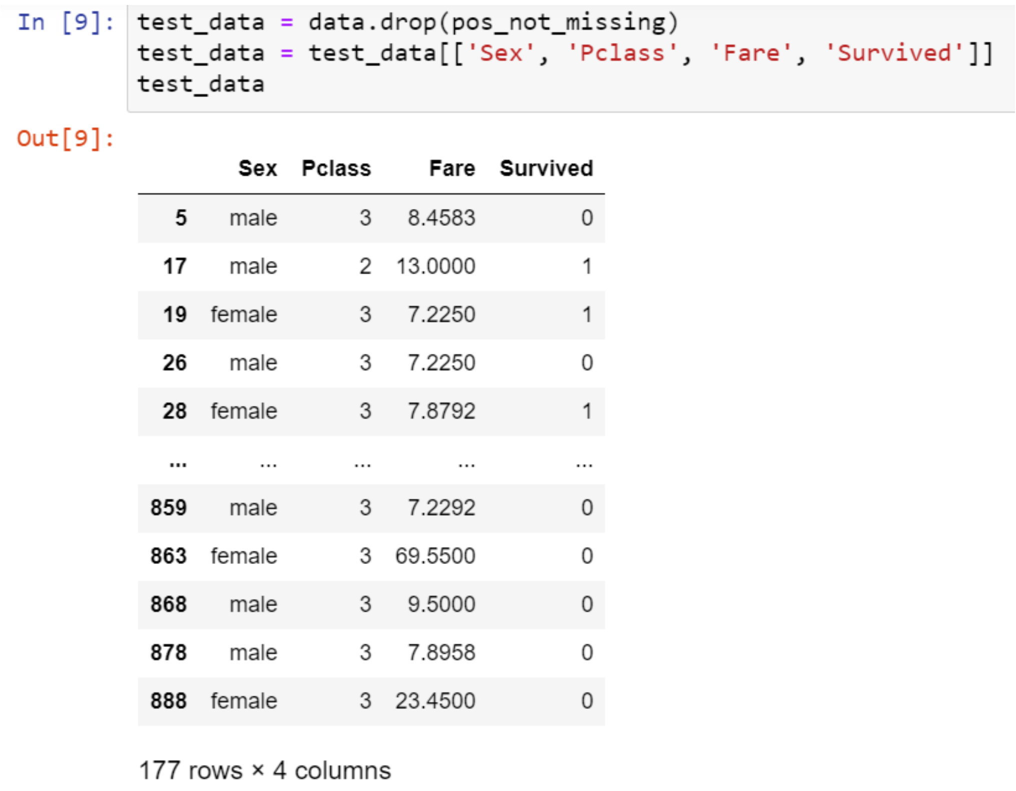


```
In [8]: train_label = data.drop(pos_missing_test)
        train_label = train_label['Age']
        train_label
```

```
Out[8]: 0      22.0
        1      38.0
        2      26.0
        3      35.0
        4      35.0
               ...
        885    39.0
        886    27.0
        887    19.0
        889    26.0
        890    32.0
        Name: Age, Length: 714, dtype: float64
```

Data fitur age yang tidak missing dijadikan train label

Code 7



```
In [9]: test_data = data.drop(pos_not_missing)
        test_data = test_data[['Sex', 'Pclass', 'Fare', 'Survived']]
        test_data
```

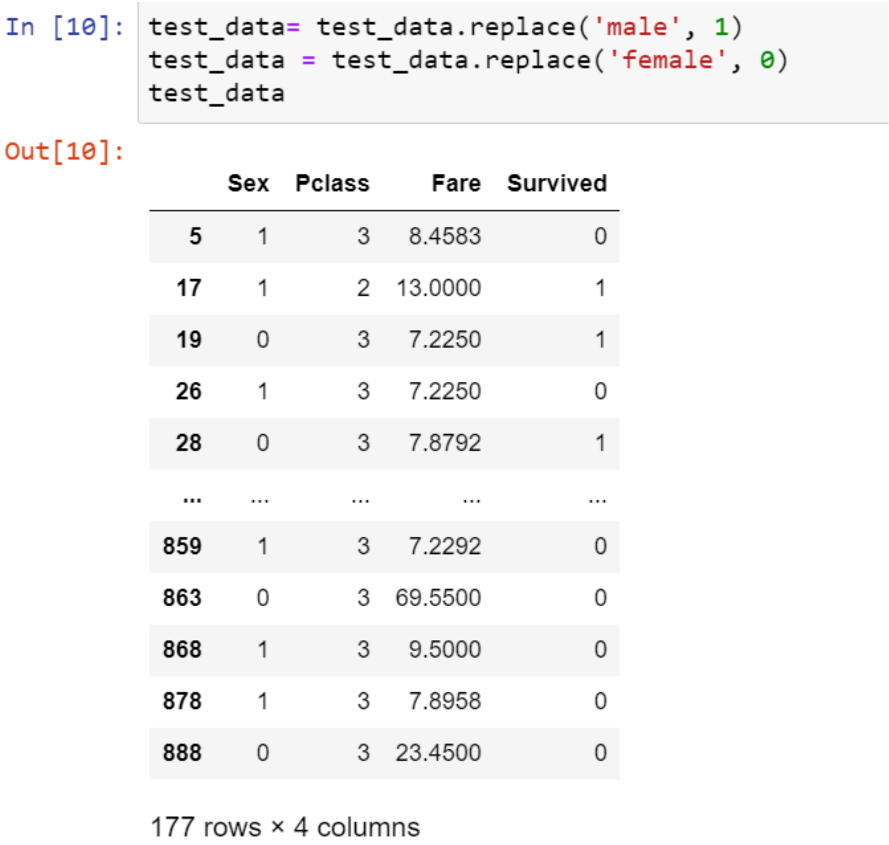
Out[9]:

	Sex	Pclass	Fare	Survived
5	male	3	8.4583	0
17	male	2	13.0000	1
19	female	3	7.2250	1
26	male	3	7.2250	0
28	female	3	7.8792	1
...	...	...	...	...
859	male	3	7.2292	0
863	female	3	69.5500	0
868	male	3	9.5000	0
878	male	3	7.8958	0
888	female	3	23.4500	0

177 rows × 4 columns

Data yang memiliki missing pada fitur age dijadikan test_data
Nantinya hasil age akan diprediksi

Code 8



In [10]:

```
test_data= test_data.replace('male', 1)
test_data = test_data.replace('female', 0)
test_data
```

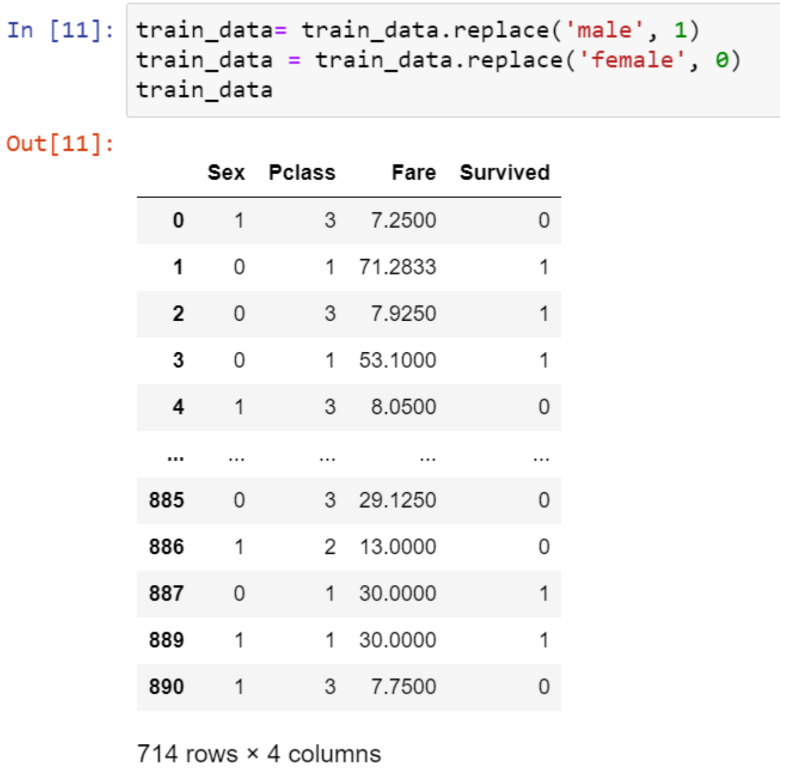
Out[10]:

	Sex	Pclass	Fare	Survived
5	1	3	8.4583	0
17	1	2	13.0000	1
19	0	3	7.2250	1
26	1	3	7.2250	0
28	0	3	7.8792	1
...	...	...	...	...
859	1	3	7.2292	0
863	0	3	69.5500	0
868	1	3	9.5000	0
878	1	3	7.8958	0
888	0	3	23.4500	0

177 rows × 4 columns

Mengganti male dan female menjadi 1 dan 0 pada test data

Code 9



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

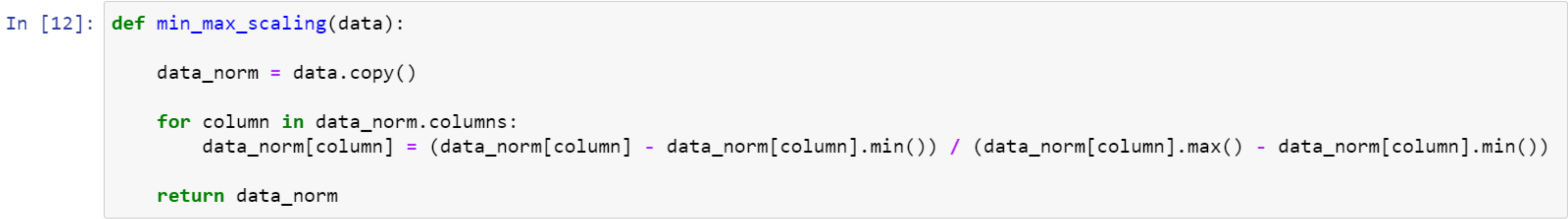
Out[11]:

	Sex	Pclass	Fare	Survived
0	1	3	7.2500	0
1	0	1	71.2833	1
2	0	3	7.9250	1
3	0	1	53.1000	1
4	1	3	8.0500	0
...	...	...	...	...
885	0	3	29.1250	0
886	1	2	13.0000	0
887	0	1	30.0000	1
889	1	1	30.0000	1
890	1	3	7.7500	0

714 rows × 4 columns

Mengganti male dan female menjadi 1 dan 0 pada train data

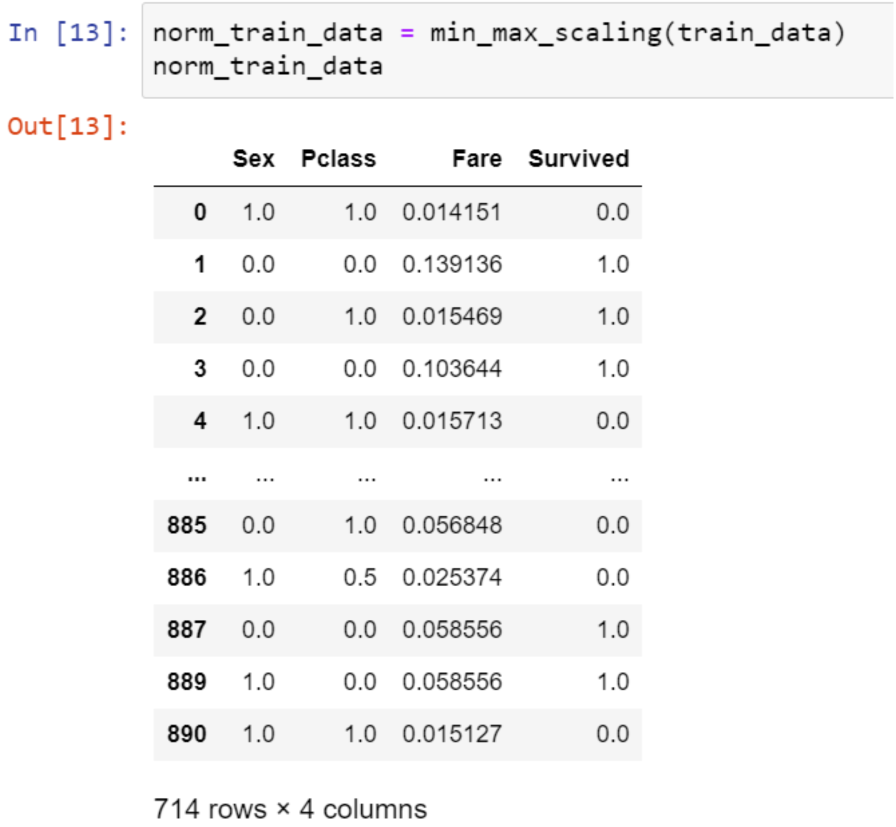
Code 10



```
In [12]: def min_max_scaling(data):

             data_norm = data.copy()

             for column in data_norm.columns:
                 data_norm[column] = (data_norm[column] - data_norm[column].min()) / (data_norm[column].max() - data_norm[column].min())

             return data_norm
```

Mendefinisikan fungsi untuk mencari scala minmax

Code 11



```
In [13]: norm_train_data = min_max_scaling(train_data)
         norm_train_data
```

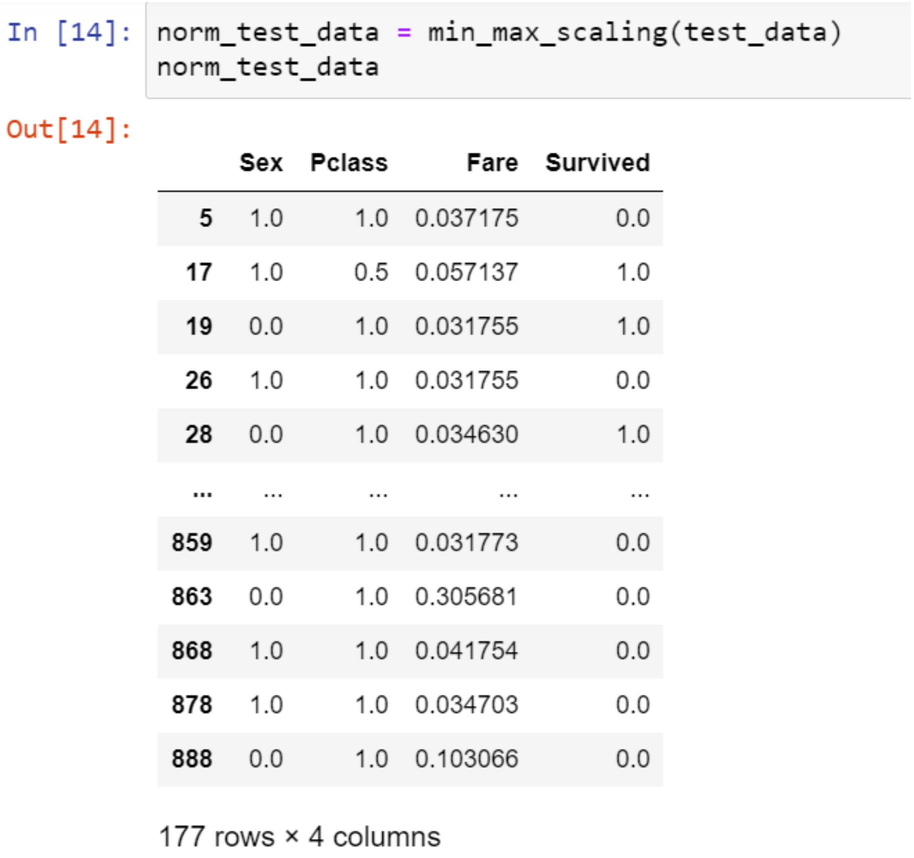
Out[13]:

	Sex	Pclass	Fare	Survived
0	1.0	1.0	0.014151	0.0
1	0.0	0.0	0.139136	1.0
2	0.0	1.0	0.015469	1.0
3	0.0	0.0	0.103644	1.0
4	1.0	1.0	0.015713	0.0
...	...	...	...	...
885	0.0	1.0	0.056848	0.0
886	1.0	0.5	0.025374	0.0
887	0.0	0.0	0.058556	1.0
889	1.0	0.0	0.058556	1.0
890	1.0	1.0	0.015127	0.0

714 rows × 4 columns

Mengubah nilai menjadi normalisasi pada train data

Code 12



In [14]:

```
norm_test_data = min_max_scaling(test_data)
norm_test_data
```

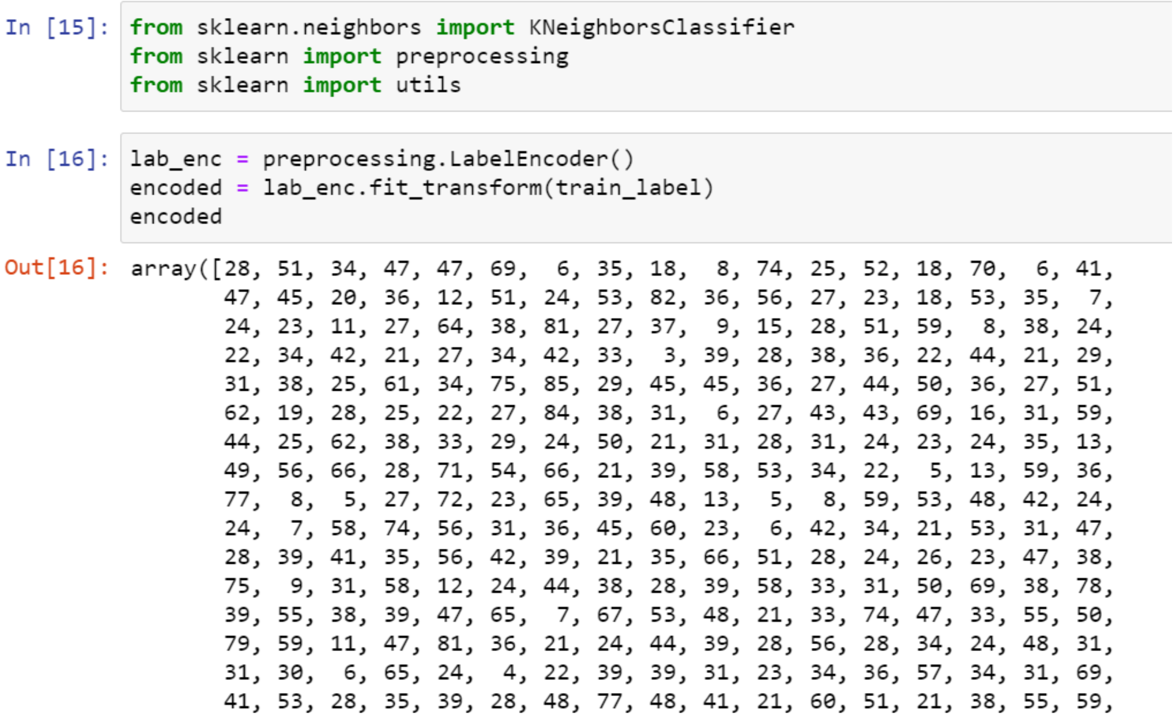
Out[14]:

	Sex	Pclass	Fare	Survived
5	1.0	1.0	0.037175	0.0
17	1.0	0.5	0.057137	1.0
19	0.0	1.0	0.031755	1.0
26	1.0	1.0	0.031755	0.0
28	0.0	1.0	0.034630	1.0
...	...	...	...	...
859	1.0	1.0	0.031773	0.0
863	0.0	1.0	0.305681	0.0
868	1.0	1.0	0.041754	0.0
878	1.0	1.0	0.034703	0.0
888	0.0	1.0	0.103066	0.0

177 rows × 4 columns

Mengubah nilai menjadi normalisasi pada test data

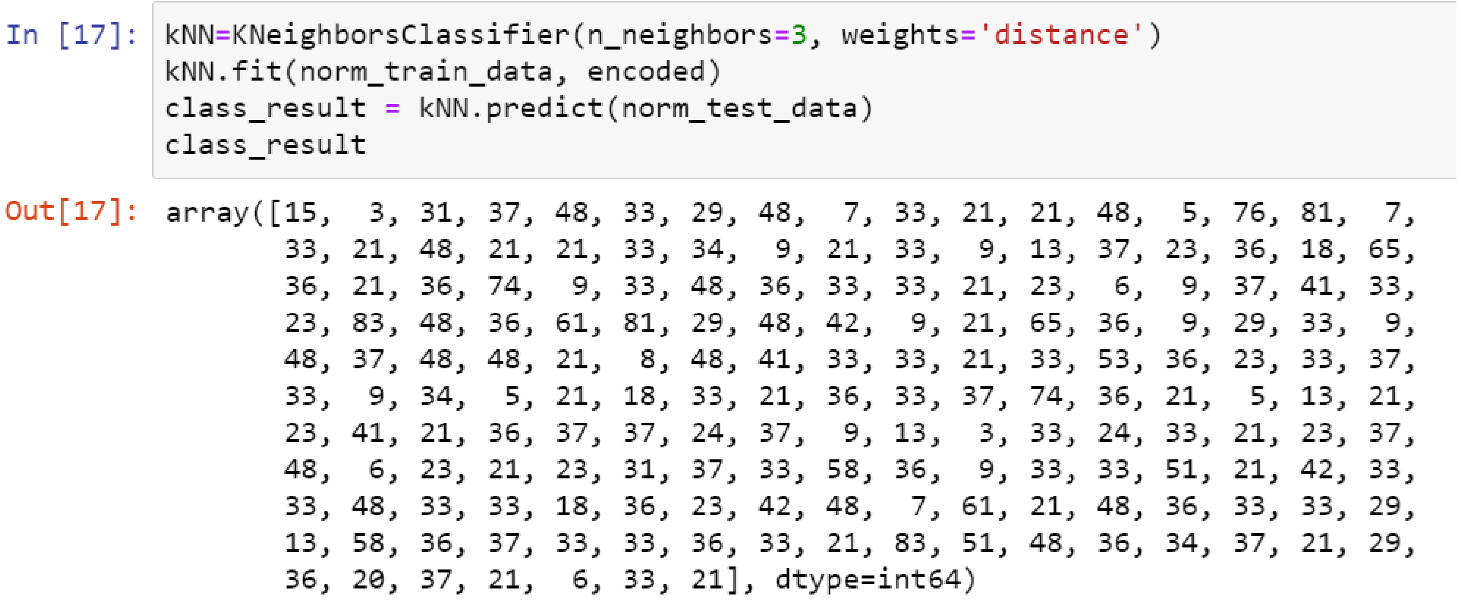
Code 13



```
In [15]: from sklearn.neighbors import KNeighborsClassifier
         from sklearn import preprocessing
         from sklearn import utils

In [16]: lab_enc = preprocessing.LabelEncoder()
         encoded = lab_enc.fit_transform(train_label)
         encoded

Out[16]: array([28, 51, 34, 47, 47, 69,  6, 35, 18,  8, 74, 25, 52, 18, 70,  6, 41,
                47, 45, 20, 36, 12, 51, 24, 53, 82, 36, 56, 27, 23, 18, 53, 35,  7,
                24, 23, 11, 27, 64, 38, 81, 27, 37,  9, 15, 28, 51, 59,  8, 38, 24,
                22, 34, 42, 21, 27, 34, 42, 33,  3, 39, 28, 38, 36, 22, 44, 21, 29,
                31, 38, 25, 61, 34, 75, 85, 29, 45, 45, 36, 27, 44, 50, 36, 27, 51,
                62, 19, 28, 25, 22, 27, 84, 38, 31,  6, 27, 43, 43, 69, 16, 31, 59,
                44, 25, 62, 38, 33, 29, 24, 50, 21, 31, 28, 31, 24, 23, 24, 35, 13,
                49, 56, 66, 28, 71, 54, 66, 21, 39, 58, 53, 34, 22,  5, 13, 59, 36,
                77,  8,  5, 27, 72, 23, 65, 39, 48, 13,  5,  8, 59, 53, 48, 42, 24,
                24,  7, 58, 74, 56, 31, 36, 45, 60, 23,  6, 42, 34, 21, 53, 31, 47,
                28, 39, 41, 35, 56, 42, 39, 21, 35, 66, 51, 28, 24, 26, 23, 47, 38,
                75,  9, 31, 58, 12, 24, 44, 38, 28, 39, 58, 33, 31, 50, 69, 38, 78,
                39, 55, 38, 39, 47, 65,  7, 67, 53, 48, 21, 33, 74, 47, 33, 55, 50,
                79, 59, 11, 47, 81, 36, 21, 24, 44, 39, 28, 56, 28, 34, 24, 48, 31,
                31, 30,  6, 65, 24,  4, 22, 39, 39, 31, 23, 34, 36, 57, 34, 31, 69,
                41, 53, 28, 35, 39, 28, 48, 77, 48, 41, 21, 60, 51, 21, 38, 55, 59,
```

Mengubah train label fitur age yang tadinya bertipe float menjadi integer

Code 14

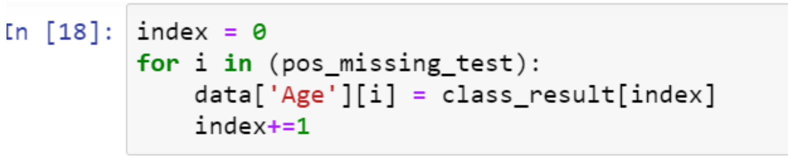


```
In [17]: kNN=KNeighborsClassifier(n_neighbors=3, weights='distance')
         kNN.fit(norm_train_data, encoded)
         class_result = kNN.predict(norm_test_data)
         class_result
```

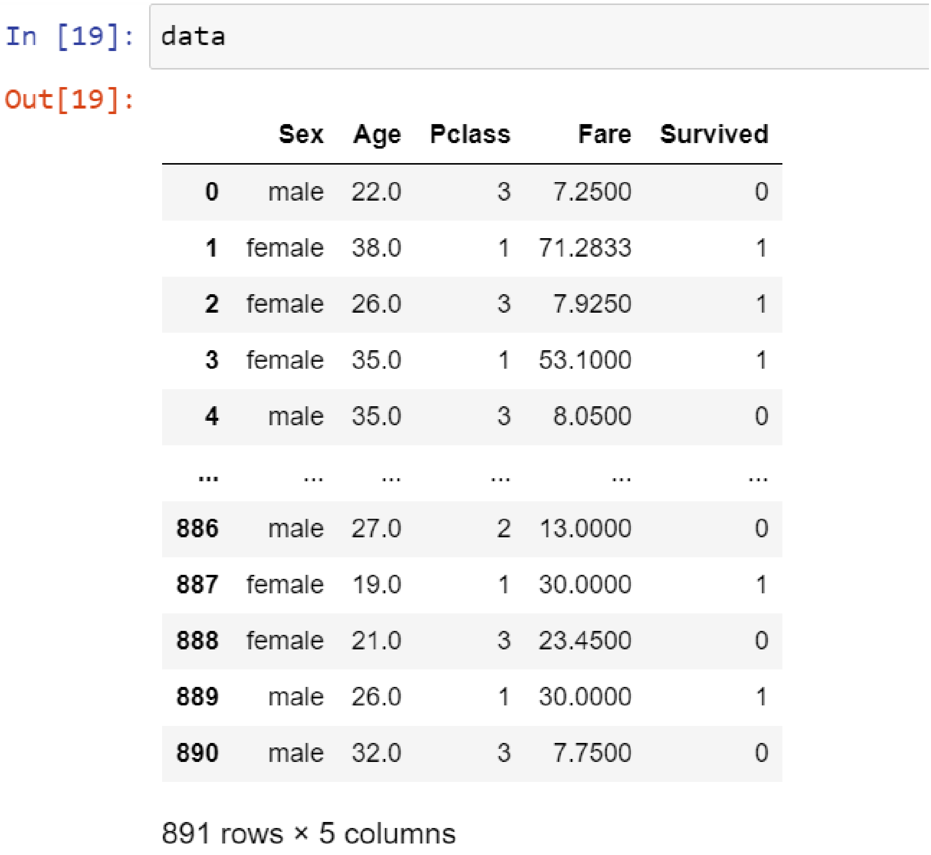
```
Out[17]: array([15,  3, 31, 37, 48, 33, 29, 48,  7, 33, 21, 21, 48,  5, 76, 81,  7,
                33, 21, 48, 21, 21, 33, 34,  9, 21, 33,  9, 13, 37, 23, 36, 18, 65,
                36, 21, 36, 74,  9, 33, 48, 36, 33, 33, 21, 23,  6,  9, 37, 41, 33,
                23, 83, 48, 36, 61, 81, 29, 48, 42,  9, 21, 65, 36,  9, 29, 33,  9,
                48, 37, 48, 48, 21,  8, 48, 41, 33, 33, 21, 33, 53, 36, 23, 33, 37,
                33,  9, 34,  5, 21, 18, 33, 21, 36, 33, 37, 74, 36, 21,  5, 13, 21,
                23, 41, 21, 36, 37, 37, 24, 37,  9, 13,  3, 33, 24, 33, 21, 23, 37,
                48,  6, 23, 21, 23, 31, 37, 33, 58, 36,  9, 33, 33, 51, 21, 42, 33,
                33, 48, 33, 33, 18, 36, 23, 42, 48,  7, 61, 21, 48, 36, 33, 33, 29,
                13, 58, 36, 37, 33, 33, 36, 33, 21, 83, 51, 48, 36, 34, 37, 21, 29,
                36, 20, 37, 21,  6, 33, 21], dtype=int64)
```

Membuat model KNN dengan k=3 dan menghasilkan umur yang diprediksi

Code 15



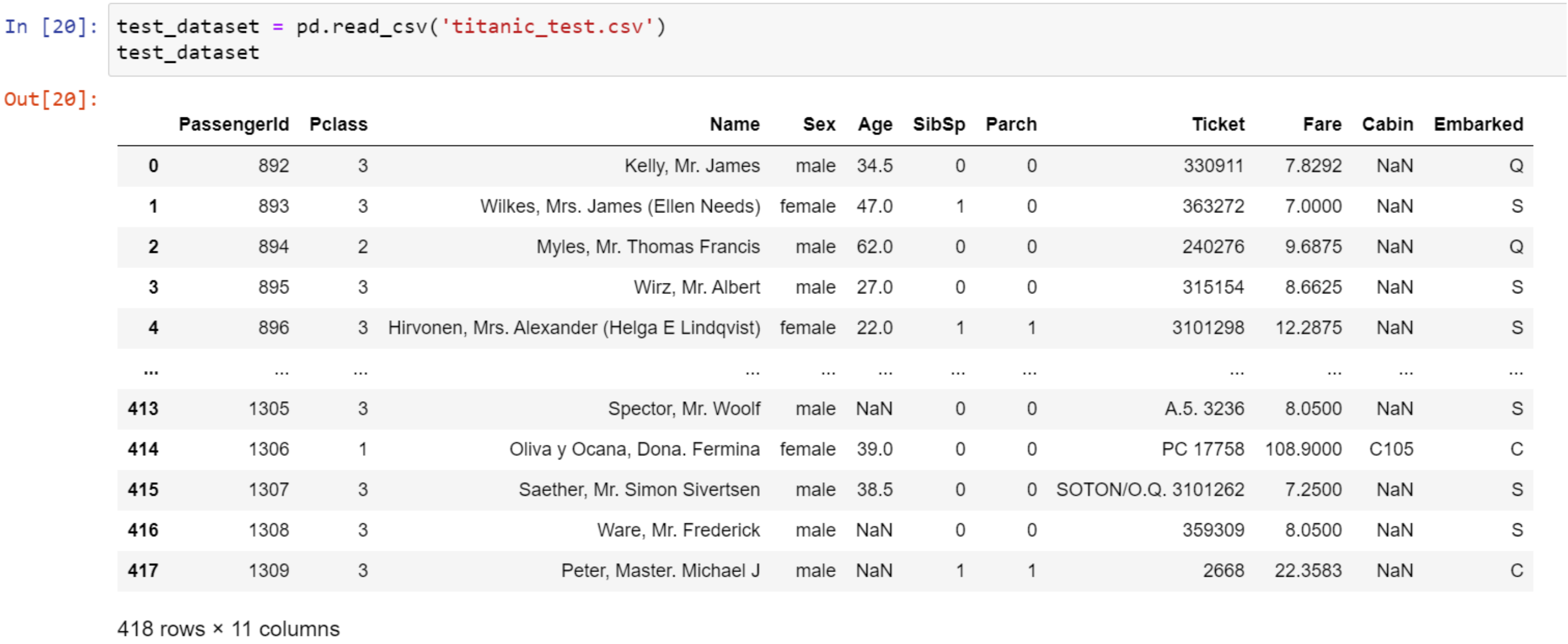
```
In [18]: index = 0
         for i in (pos_missing_test):
             data['Age'][i] = class_result[index]
             index+=1
```



```
In [19]: data

Out[19]:
            Sex   Age  Pclass     Fare  Survived
      0    male  22.0       3   7.2500         0
      1  female  38.0       1  71.2833         1
      2  female  26.0       3   7.9250         1
      3  female  35.0       1  53.1000         1
      4    male  35.0       3   8.0500         0
    ...     ...   ...     ...      ...       ...
    886    male  27.0       2  13.0000         0
    887  female  19.0       1  30.0000         1
    888  female  21.0       3  23.4500         0
    889    male  26.0       1  30.0000         1
    890    male  32.0       3   7.7500         0

891 rows × 5 columns
```

Mereplace nilai missing pada fitur age pada data asli

Code 16



```
In [20]: test_dataset = pd.read_csv('titanic_test.csv')
         test_dataset
```

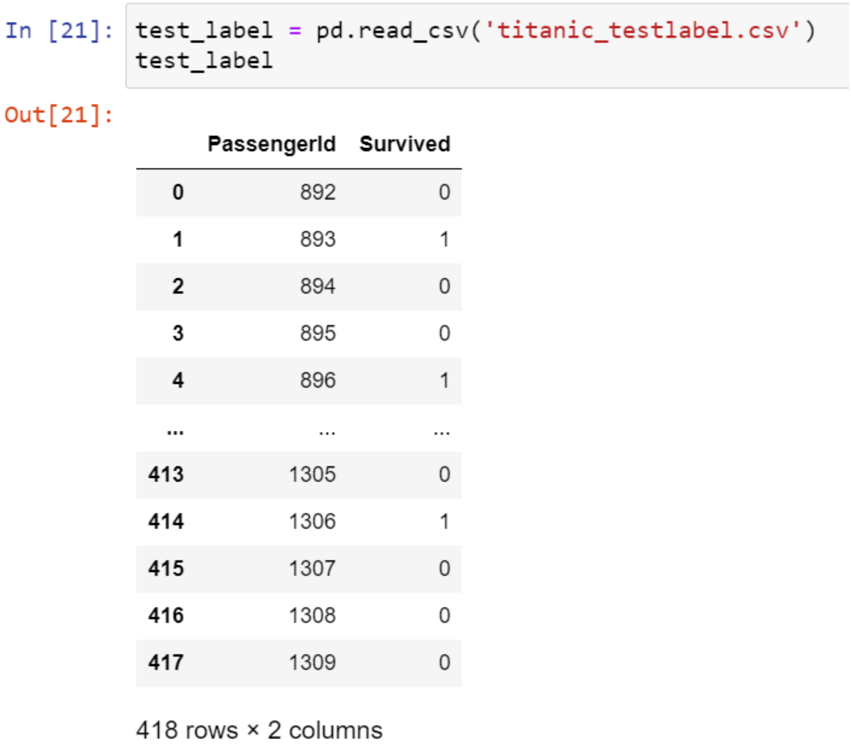
Out[20]:

	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 titanic_test.csv

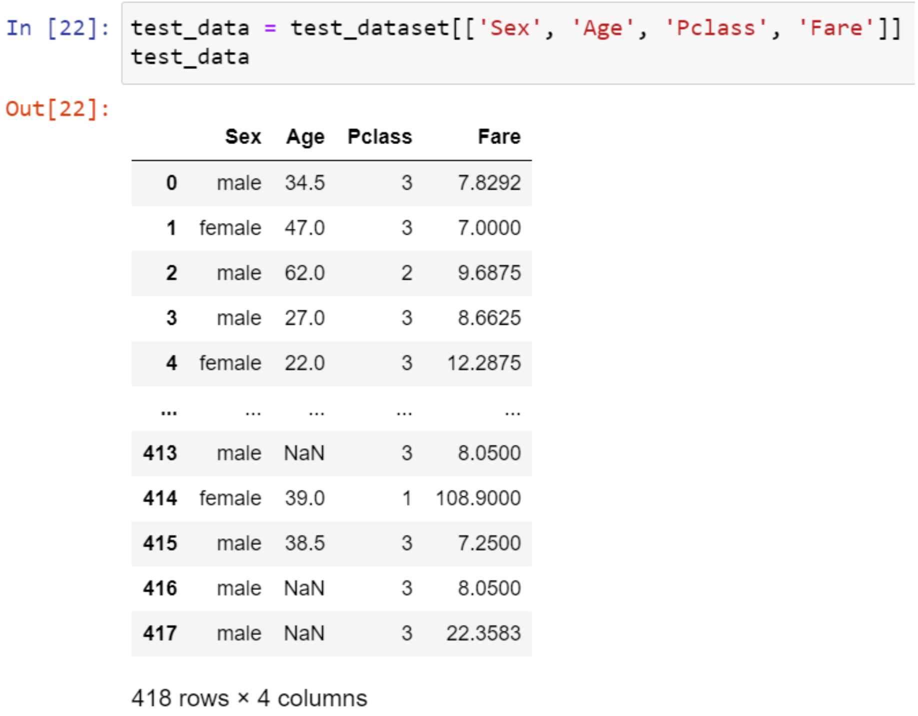
Code 17



```
In [21]: test_label = pd.read_csv('titanic_testlabel.csv')
         test_label

Out[21]:
              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

         418 rows × 2 columns
```

Membaca titanic_testlabel.csv

Code 18



```
In [22]: test_data = test_dataset[['Sex', 'Age', 'Pclass', 'Fare']]
         test_data
```

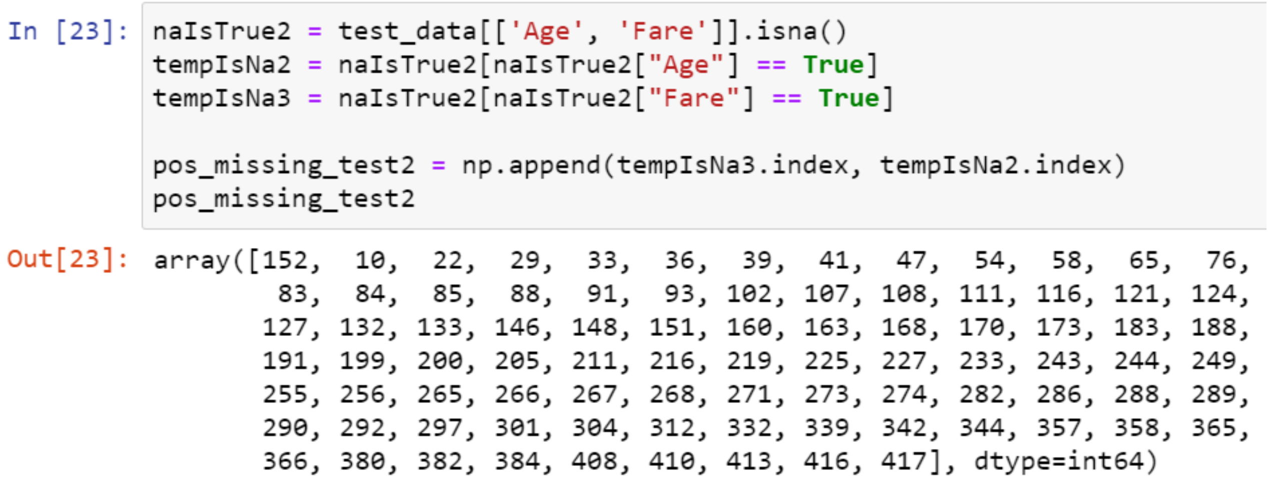
Out[22]:

	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

418 rows × 4 columns

Mengambil test_dataset fitur sex, age, pclass, fare

Code 19

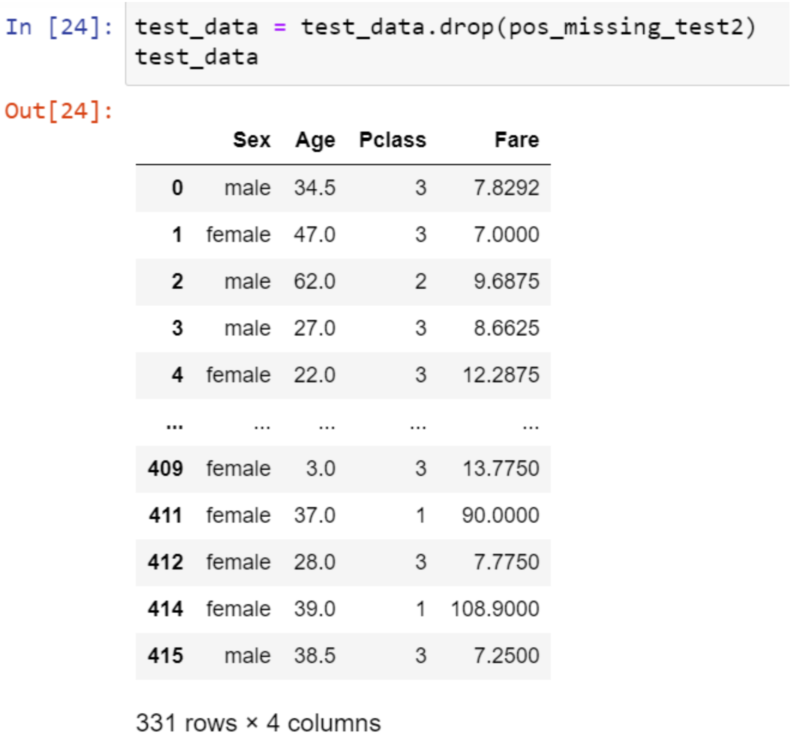


```
In [23]: naIsTrue2 = test_data[['Age', 'Fare']].isna()
         tempIsNa2 = naIsTrue2[naIsTrue2["Age"] == True]
         tempIsNa3 = naIsTrue2[naIsTrue2["Fare"] == True]

         pos_missing_test2 = np.append(tempIsNa3.index, tempIsNa2.index)
         pos_missing_test2
```

```
Out[23]: 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 nilai missing dari fitur age dan fare
Pada test data

Code 20



```
In [24]: test_data = test_data.drop(pos_missing_test2)
         test_data
```

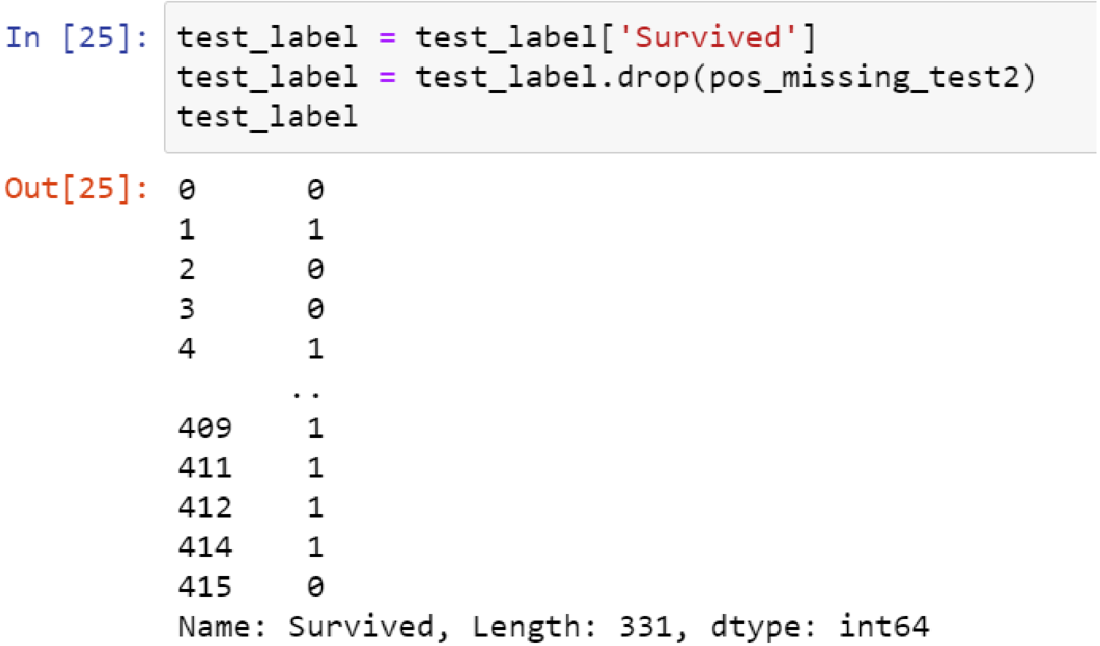
Out[24]:

	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
...	...	...	...	...
409	female	3.0	3	13.7750
411	female	37.0	1	90.0000
412	female	28.0	3	7.7750
414	female	39.0	1	108.9000
415	male	38.5	3	7.2500

331 rows × 4 columns

Menghapus baris yang memiliki missing value pada test data

Code 21



In [25]:

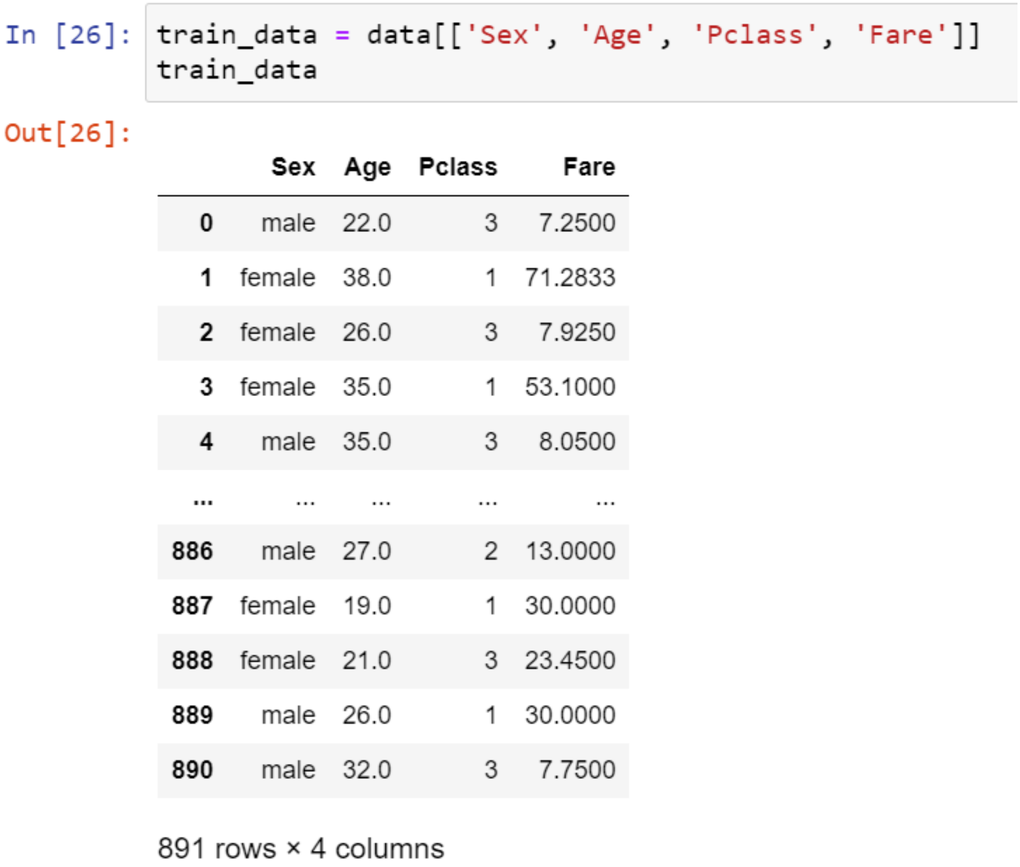
```
test_label = test_label['Survived']
test_label = test_label.drop(pos_missing_test2)
test_label
```

Out[25]:

```
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
```

Mengambil test label kolom survived

Code 22

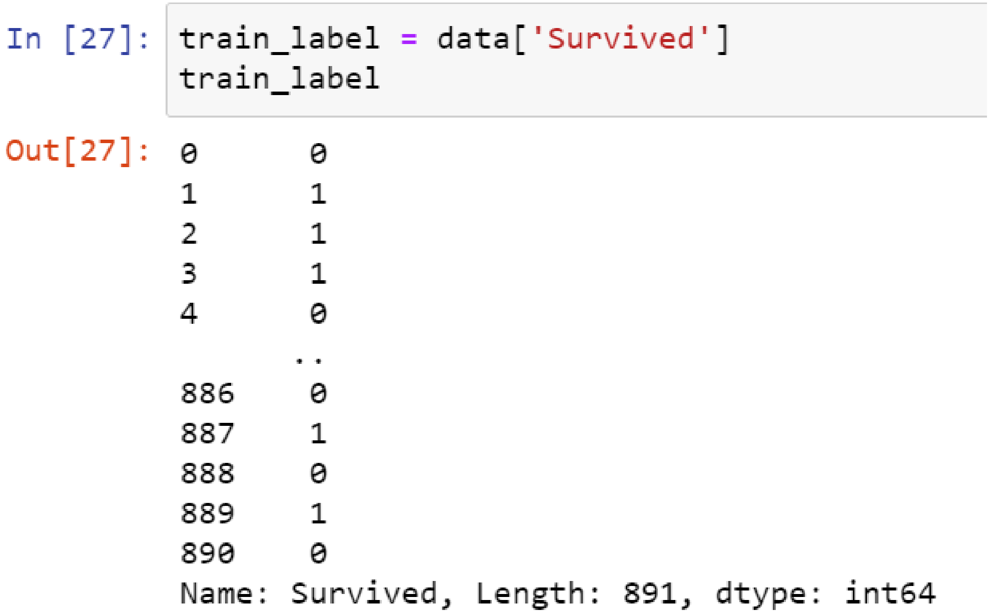


```
In [26]: train_data = data[['Sex', 'Age', 'Pclass', 'Fare']]
         train_data
```

```
Out[26]:
            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  21.0       3  23.4500
    889    male  26.0       1  30.0000
    890    male  32.0       3   7.7500

    891 rows × 4 columns
```

Mengambil data terbaru dengan fitur sex, age, pclass, fare sebagai train data

Code 23

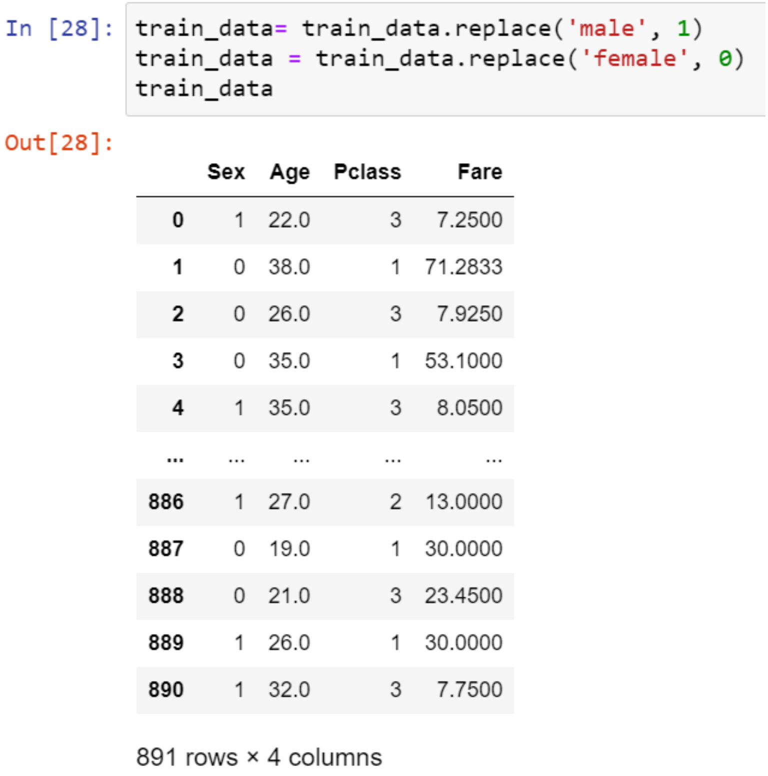


```
In [27]: train_label = data['Survived']
         train_label
```

```
Out[27]: 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 data fitur survived sebagai train label

Code 24

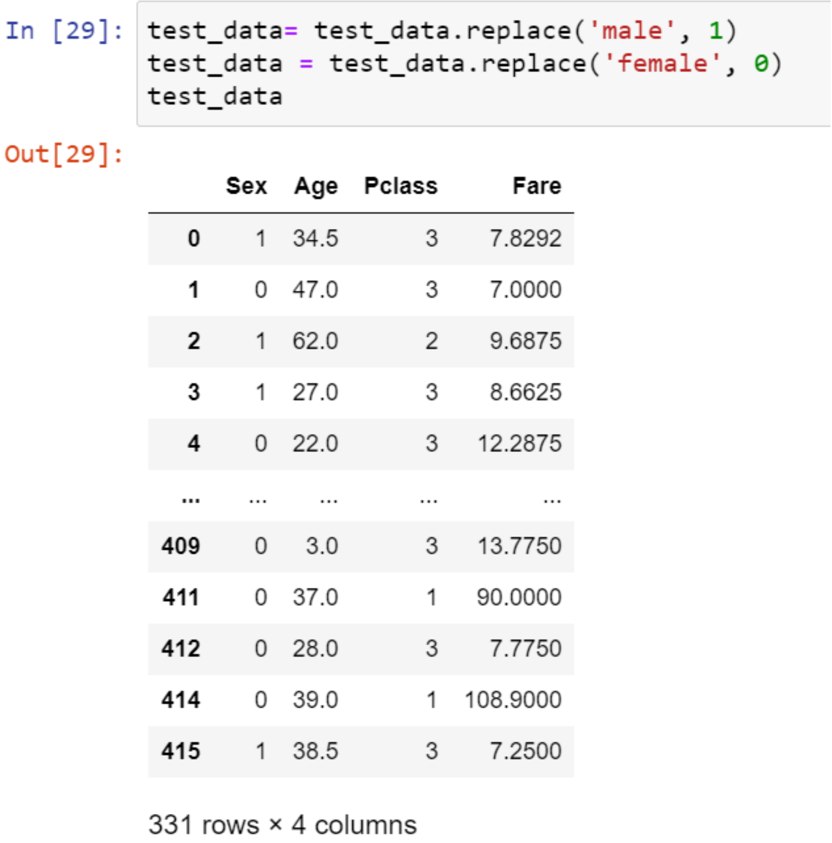


```
In [28]: train_data= train_data.replace('male', 1)
         train_data = train_data.replace('female', 0)
         train_data
```

```
Out[28]:
        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  21.0       3  23.4500
 889      1  26.0       1  30.0000
 890      1  32.0       3   7.7500

891 rows × 4 columns
```

Mengganti male dan female menjadi 1 dan 0 pada train data

Code 25



```
In [29]: test_data= test_data.replace('male', 1)
         test_data = test_data.replace('female', 0)
         test_data
```

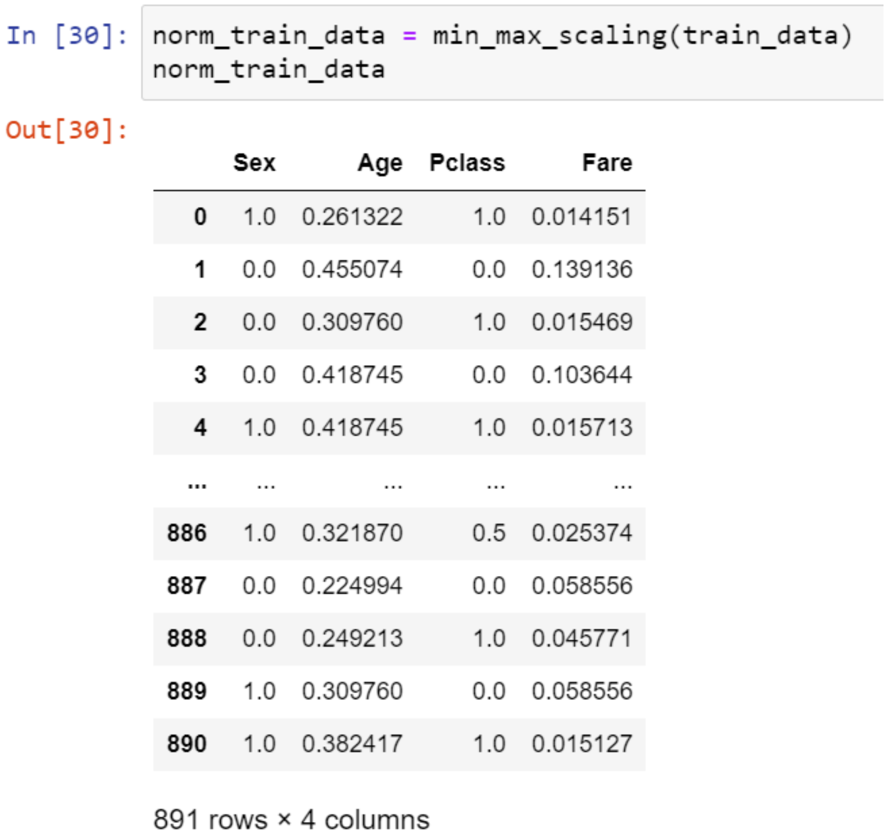
Out[29]:

	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

Mengganti male dan female menjadi 1 dan 0 pada test data

Code 26

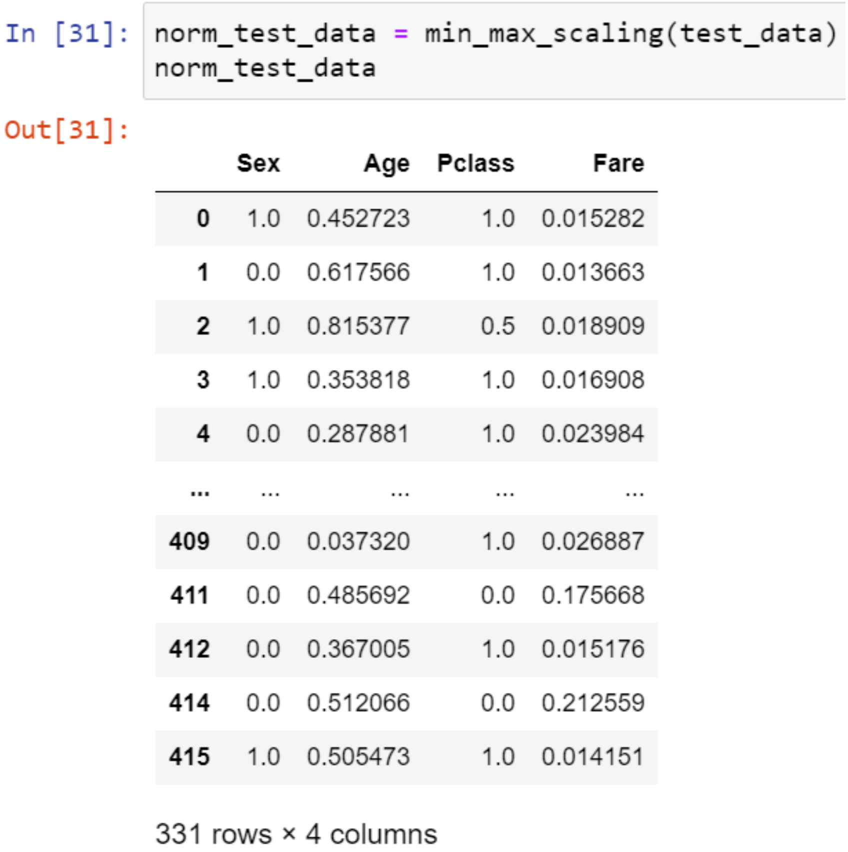


```
In [30]: norm_train_data = min_max_scaling(train_data)
         norm_train_data
```

```
Out[30]:
         Sex       Age  Pclass      Fare
    0    1.0  0.261322     1.0  0.014151
    1    0.0  0.455074     0.0  0.139136
    2    0.0  0.309760     1.0  0.015469
    3    0.0  0.418745     0.0  0.103644
    4    1.0  0.418745     1.0  0.015713
  ...    ...       ...     ...       ...
  886    1.0  0.321870     0.5  0.025374
  887    0.0  0.224994     0.0  0.058556
  888    0.0  0.249213     1.0  0.045771
  889    1.0  0.309760     0.0  0.058556
  890    1.0  0.382417     1.0  0.015127

891 rows × 4 columns
```

Mengubah train data baru menjadi normalisasi

Code 27



```
In [31]: norm_test_data = min_max_scaling(test_data)
         norm_test_data
```

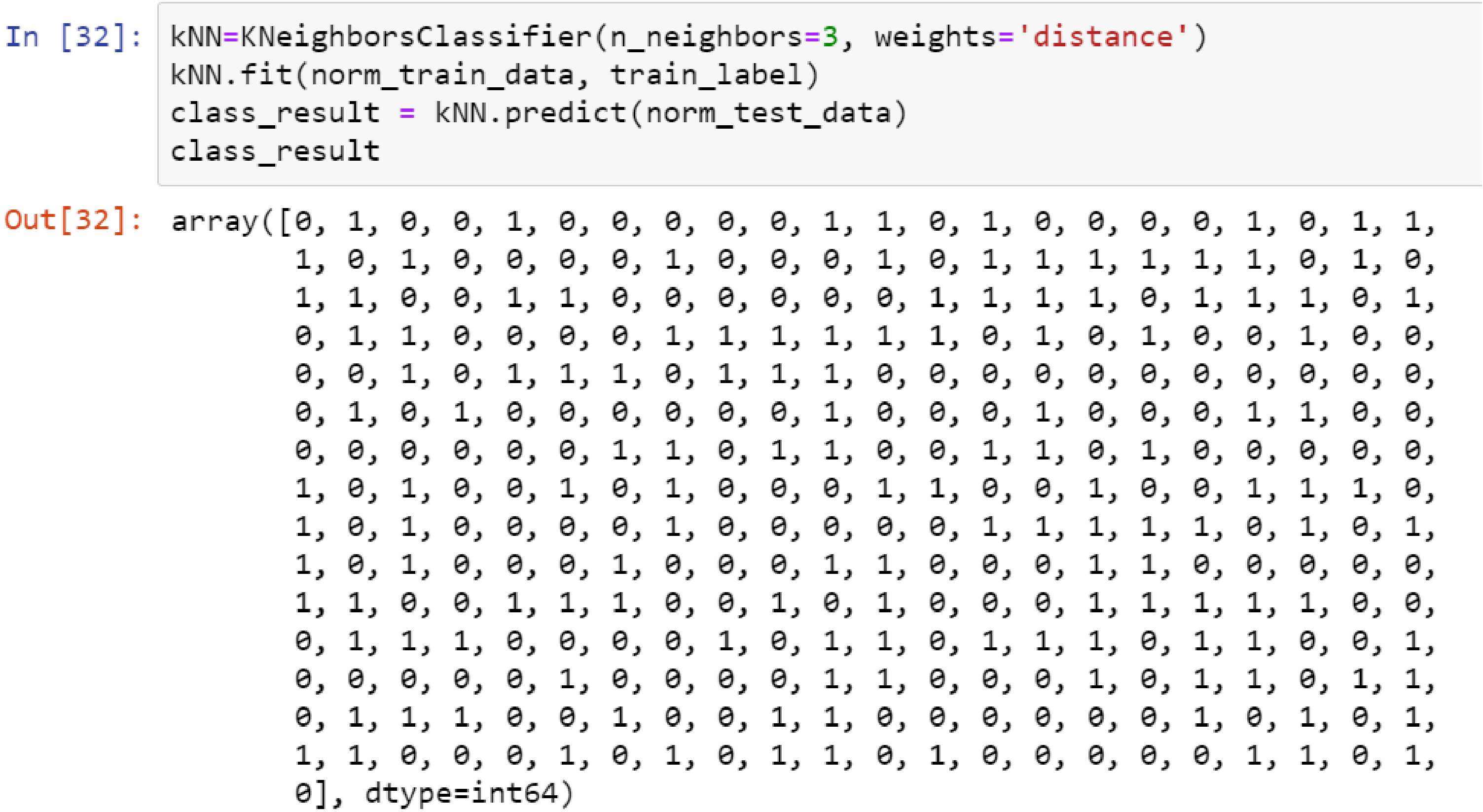
Out[31]:

	Sex	Age	Pclass	Fare
0	1.0	0.452723	1.0	0.015282
1	0.0	0.617566	1.0	0.013663
2	1.0	0.815377	0.5	0.018909
3	1.0	0.353818	1.0	0.016908
4	0.0	0.287881	1.0	0.023984
...	...	...	...	...
409	0.0	0.037320	1.0	0.026887
411	0.0	0.485692	0.0	0.175668
412	0.0	0.367005	1.0	0.015176
414	0.0	0.512066	0.0	0.212559
415	1.0	0.505473	1.0	0.014151

331 rows × 4 columns

Mengubah test data baru menjadi normalisasi

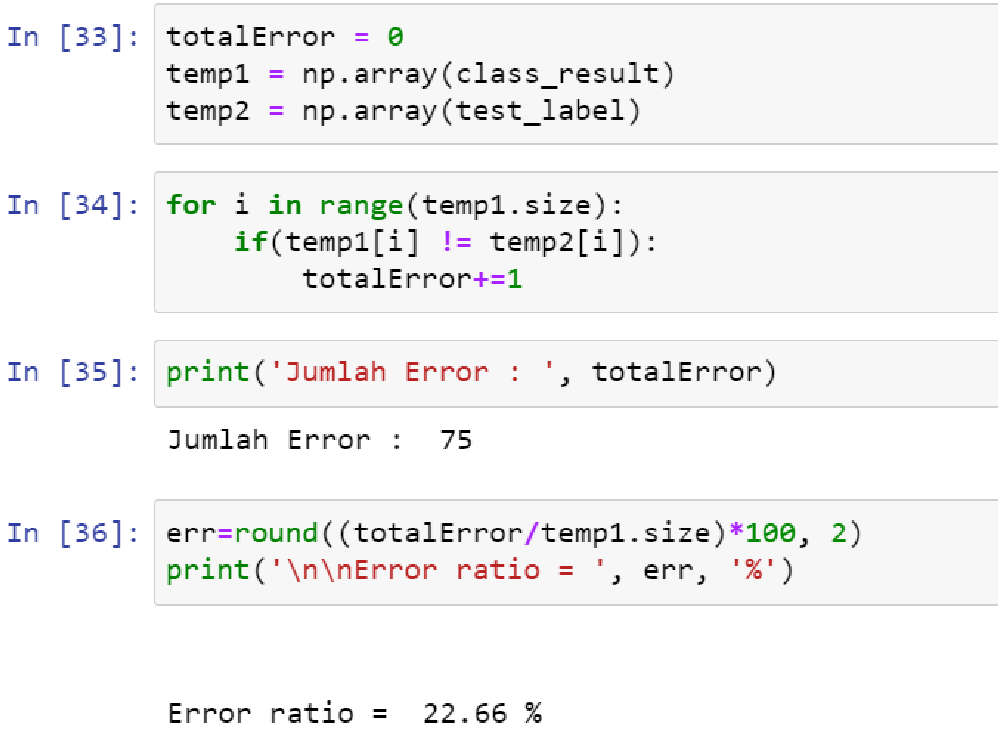
Code 28



```
In [32]: kNN=KNeighborsClassifier(n_neighbors=3, weights='distance')
         kNN.fit(norm_train_data, train_label)
         class_result = kNN.predict(norm_test_data)
         class_result

Out[32]: array([0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1,
                1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0,
                1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1,
                0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0,
                0, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0,
                0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0,
                1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0,
                1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 0, 1,
                1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0,
                1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0,
                0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1,
                0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1,
                0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1,
                1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1,
                0], dtype=int64)
```

Membuat model knn dengan k = 3 untuk memprediksi hasil survived

Code 29



```
In [33]: totalError = 0
         temp1 = np.array(class_result)
         temp2 = np.array(test_label)
```

```
In [34]: for i in range(temp1.size):
             if(temp1[i] != temp2[i]):
                 totalError+=1
```

```
In [35]: print('Jumlah Error : ', totalError)

         Jumlah Error :  75
```

```
In [36]: err=round((totalError/temp1.size)*100, 2)
         print('\n\nError ratio = ', err, '%')


         Error ratio =  22.66 %
```

Dapat kita ketahui dengan membandingkan class result dan test label
Kita peroleh jumlah error = 75 dan ratio error = 22.66%