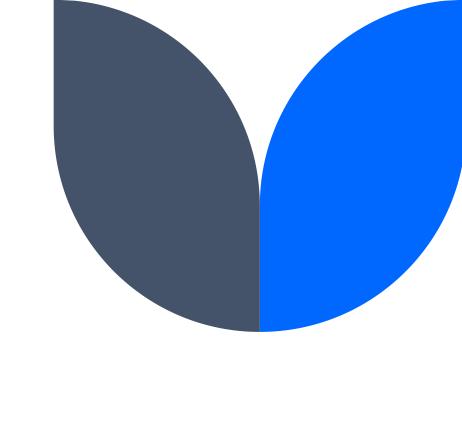
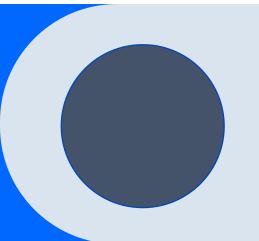
MidTest Minggu Ke-7





Faris Saifullah 3124640034

```
[17] #no.1
     import pandas as pd
     dataset = pd.read_csv('titanic.csv')
     (dataset.head())
                                                                                                                                                        PassengerId Survived Pclass
                                                                                    Sex Age SibSp Parch
                                                                                                                     Ticket
                                                                                                                                Fare Cabin Embarked
                                                                           Name
                            0
                                                           Braund, Mr. Owen Harris
                                                                                                                   A/5 21171
     0
                                                                                   male 22.0
                                                                                                         0
                                                                                                                             7.2500
                                                                                                                                       NaN
                                                                                                                                                        1 Cumings, Mrs. John Bradley (Florence Briggs Th... female 38.0
                                                                                                                   PC 17599 71.2833
                                                                                                                                       C85
                                                                                                                                                   С
                                                             Heikkinen, Miss. Laina female 26.0
                                                                                                         0 STON/O2. 3101282 7.9250
                                                                                                                                       NaN
                                                                                                                                                   S
                                            Futrelle, Mrs. Jacques Heath (Lily May Peel) female 35.0
                                                                                                                      113803 53.1000
                                                                                                                                      C123
                                                                                                                                                   S
                                                            Allen, Mr. William Henry
                                                                                   male 35.0
                                                                                                                     373450
                                                                                                                              8.0500
                                                                                                                                       NaN
```

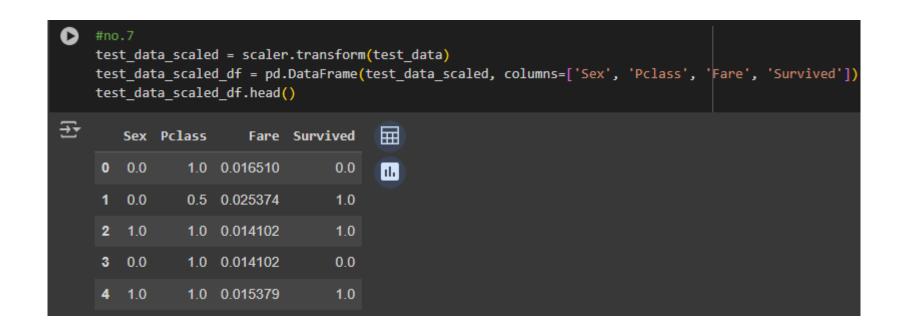








```
[26] #no.6
     from sklearn.preprocessing import MinMaxScaler
     scaler = MinMaxScaler()
     train_data_normalized = scaler.fit_transform(train_data)
     print("Normalized train_data:\n", train_data_normalized)
     print("Min values of train_data:\n", scaler.data_min_)
     print("Max values of train_data:\n", scaler.data_max_)
→ Normalized train_data:
      [[0.
                  1.
                             0.01415106 0.
      [1.
                 0.
                            0.13913574 1.
      [1.
                            0.01546857 1.
      [1.
                 0.
                            0.0585561 1.
      [0.
                 0.
                            0.0585561 1.
      [0.
                 1.
                            0.01512699 0.
     Min values of train_data:
     [0. 1. 0. 0.]
     Max values of train_data:
      1.
                        512.3292 1.
```



```
from sklearn.neighbors import KNeighborsRegressor
knn_regressor = KNeighborsRegressor(n_neighbors=3)
knn_regressor.fit(train_data[['Sex', 'Pclass', 'Fare', 'Survived']], train_label)
class_result = knn_regressor.predict(test_data[['Sex', 'Pclass', 'Fare', 'Survived']])
print("Hasil prediksi usia untuk data yang hilang:", class_result)
Hasil prediksi usia untuk data yang hilang: [31.66666667 35.66666667 16.66666667 33.5
                                                                                           21.33333333 29.
26.97333333 20.
                                    29.
                        22.
                                                36.83333333 36.
            18.66666667 43.
                                    38.5
                                               10.66666667 29.
20.
 36.83333333 20.
                        36.83333333 36.83333333 29.
                                                           24.33333333
 22.66666667 36.83333333 55.83333333 18.66666667 29.33333333 21.5
 25.3333333 39.3333333 28.66666667 57.66666667 41.
                                                           39.33333333
 41.33333333 42.
                                    55.83333333 20.
                        21.
                                                            39.33333333
                        33.3333333 32.16666667 25.83333333 21.
 55.83333333 29.
 21.5
            27.66666667 55.83333333 32.16666667 50.33333333 20.
 35.66666667 57.66666667 38.5
                                    37.
                                               20.
                                                            29.33333333
                                    39.3333333 18.66666667 43.66666667
            36.83333333 35.
 29.
            19.33333333 54.33333333 33.5
                                               21.33333333 21.33333333
 36.
            16.66666667 20.
                                    39.66666667 29.
                                                           26.
 33.3333333 29.
                                    35.66666667 25.33333333 29.
                        31.
 21.5
            55.8333333 19.33333333 17.66666667 22.33333333 36.83333333
 45.
            55.83333333 36.83333333 35.66666667 26.
                                                           21.5
                                                           38.83333333
             35.66666667 33.33333333 22.33333333 24.
 32.16666667 42.66666667 36.83333333 33.
                                               33.5
                                                           20.83333333
 32,66666667 20,83333333 18,66666667 28,83333333 21,89
 32.66666667 55.83333333 36.83333333 25.33333333 20.83333333 20.
             25.3333333 36.83333333 32.16666667 22.66666667 33.5
            16.66666667 26.33333333 21.
 29.
                                               55.83333333 26.
            32.16666667 30.
                                    29.33333333 29.
                                                           21.333333333
 29.
            24.66666667 45.
                                    35.66666667 25.33333333 30.
            10.66666667 54.33333333 36.
                                               18.
                                                            35.66666667
                                    28.83333333 47.66666667 33.
 29.
            29.
                        46.
 33.5
            55.83333333 26.
                                    33.
                                               55.83333333 39.33333333
 46.66666667 39.
                        32,66666667 28,66666667 25.
                                                           20.83333333
 36.83333333 46.
                        39.33333333 39.
                                               20.83333333 39.33333333
            29.
                        25.333333333
```

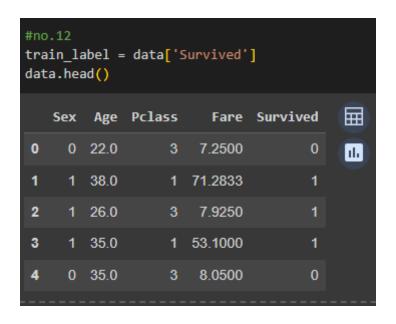
```
#no.9
if len(class_result) == data['Age'].isna().sum():
   data.loc[data['Age'].isna(), 'Age'] = class_result
   print("\nData dengan nilai Age yang sudah diisi:")
   print(data.head())
else:
   print(f"Error: Jumlah prediksi ({len(class_result)}) tidak cocok dengan jumlah nilai Age yang hilang ({data['Age'].isna().sum()})")
Data dengan nilai Age yang sudah diisi:
                       Fare Survived
  Sex Age Pclass
    0 22.0
                 3 7.2500
    1 38.0
                 1 71.2833
    1 26.0
                 3 7.9250
    1 35.0
                 1 53.1000
    0 35.0
                 3 8.0500
                                   0
```

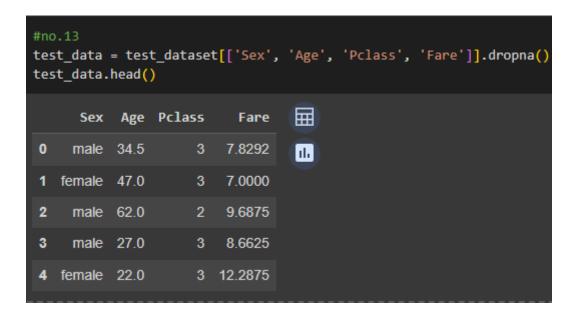






```
train_data = pd.read_csv('titanic_test.csv')
train_data = train_data[['Sex', 'Age', 'Pclass', 'Fare']]
train_data.head()
                              Sex Age Pclass
                        Fare
0 male 34.5
                   3 7.8292
                               Ш
                   3 7.0000
 1 female 47.0
2 male 62.0
                   2 9.6875
    male 27.0
                   3 8.6625
 4 female 22.0
                   3 12.2875
```







```
#no.15
train_data = pd.read_csv('titanic.csv')
train_data = train_data[['Sex', 'Age', 'Pclass', 'Fare']]
le = LabelEncoder()
train_data['Sex'] = le.fit_transform(train_data['Sex'])
train_data['Age'].fillna(train_data['Age'].mean(), inplace=True)
scaler = MinMaxScaler()
train_data = scaler.fit_transform(train_data)
min_values = scaler.data_min_
max_values = scaler.data_max_
train_data = pd.DataFrame(train_data, columns=['Sex', 'Age', 'Pclass', 'Fare'])
train_data.head()
print("Normalized train_data:\n", train_data)
print("Min values of train_data:\n", min_values)
print("Max values of train_data:\n", max_values)
Normalized train data:
      Sex
               Age Pclass
                               Fare
   1.0 0.271174
                     1.0 0.014151
    0.0 0.472229
                     0.0 0.139136
    0.0 0.321438 1.0 0.015469
    0.0 0.434531
                     0.0 0.103644
    1.0 0.434531
                     1.0 0.015713
886 1.0 0.334004
                     0.5 0.025374
887 0.0 0.233476
                     0.0 0.058556
888 0.0 0.367921
                     1.0 0.045771
889 1.0 0.321438
                     0.0 0.058556
890 1.0 0.396833
                     1.0 0.015127
```