



# **Praktikum Data Mining**

## **Minggu Ke-6**



**Faris Saifullah**  
**3124640034**

# Input

```
#no.1
import pandas as pd
dataset = pd.read_csv('titanic.csv')
print(dataset.head())
```

Analisa:  
Menampilkan dataset dari file data titanic.csv

# Output

```
PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                     Name    Sex  Age  SibSp  \
0                               Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female  38.0      1
2                               Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                               Allen, Mr. William Henry    male  35.0      0

   Parch    Ticket   Fare Cabin Embarked
0      0   A/5 21171   7.2500   NaN        S
1      0    PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
3      0    113803  53.1000  C123        S
4      0    373450   8.0500   NaN        S
```

# Input

```
#no.2
test_dataset = pd.read_csv('titanic_test.csv')
print(test_dataset.head())
```

# Output

	PassengerId	Pclass	Name	Sex
0	892	3	Kelly, Mr. James	male
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female
2	894	2	Myles, Mr. Thomas Francis	male
3	895	3	Wirz, Mr. Albert	male
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female

	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	34.5	0	0	330911	7.8292	NaN	Q
1	47.0	1	0	363272	7.0000	NaN	S
2	62.0	0	0	240276	9.6875	NaN	Q
3	27.0	0	0	315154	8.6625	NaN	S
4	22.0	1	1	3101298	12.2875	NaN	S

Analisa:  
Menampilkan dataset dari file data titanic\_test.csv



# Input

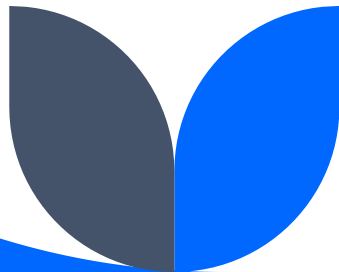
```
#no.3
train_data = pd.read_csv('titanic.csv', usecols=['Sex', 'Age', 'Pclass', 'Fare'])
mean_age_by_class = train_data.groupby('Pclass')['Age'].mean()
def fill_missing_age(row):
    if pd.isnull(row['Age']):
        return mean_age_by_class[row['Pclass']]
    else:
        return row['Age']
train_data['Age'] = train_data.apply(fill_missing_age, axis=1)
print(train_data.head())
```

# Output

	Pclass	Sex	Age	Fare
0	3	male	22.0	7.2500
1	1	female	38.0	71.2833
2	3	female	26.0	7.9250
3	1	female	35.0	53.1000
4	3	male	35.0	8.0500

Analisa:

Melakukan pengisian data missing value pada fitur Age dengan nilai mean dari masing-masing class



# Input

```
#no.4
test_data = pd.read_csv('titanic_test.csv', usecols=['Sex', 'Age', 'Pclass', 'Fare'])
print(test_data.head())
```

# Output

	Pclass	Sex	Age	Fare
0	3	male	34.5	7.8292
1	3	female	47.0	7.0000
2	2	male	62.0	9.6875
3	3	male	27.0	8.6625
4	3	female	22.0	12.2875

Analisa:

Menampilkan data test\_data pada kolom Sex, Age, Pclass, Fare



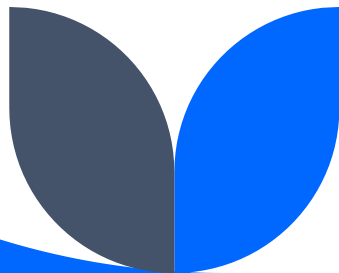
# Input

```
#no.5  
train_label = pd.read_csv('titanic.csv', usecols=['Survived'])  
print(train_label.head())
```

Analisa:  
Menampilkan data train\_label terutama kolom survived

# Output

	Survived
0	0
1	1
2	1
3	1
4	0



# Input

```
#no.6  
test_label = pd.read_csv('titanic_testlabel.csv')  
print(test_label.head())
```

Analisa:  
Menampilkan data pada file titanic\_testlabel.csv

# Output

	PassengerId	Survived
0	892	0
1	893	1
2	894	0
3	895	0
4	896	1



# Input

```
#no.7
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder

label_encoder = LabelEncoder()
X_train['Sex'] = label_encoder.fit_transform(X_train['Sex'])
X_test['Sex'] = label_encoder.transform(X_test['Sex'])
clf = DecisionTreeClassifier()
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
error_rate = 1 - accuracy_score(y_test, y_pred)
print("Error rate:", error_rate)
```

# Output

```
Error rate: 0.19617224880382778
```

Analisa:

Menampilkan ratio error pada klasifikasi data yang menggunakan metode tree decision



# Input

```
#no.8
from sklearn.tree import export_text

tree_text = export_text(clf, feature_names=['Sex', 'Age', 'Pclass', 'Fare'])
print(tree_text)
```

Analisa:  
Menampilkan hirarki dari Decision Tree

# Output

```
--- Sex <= 0.50
|--- Pclass <= 2.50
|   |--- Age <= 2.50
|   |   |--- Pclass <= 1.50
|   |   |   |--- class: 0
|   |   |   |--- Pclass > 1.50
|   |   |   |--- class: 1
|   |   |--- Age > 2.50
|   |   |--- Fare <= 28.86
|   |   |   |--- Fare <= 28.23
|   |   |   |   |--- Age <= 56.00
|   |   |   |   |   |--- Age <= 23.50
|   |   |   |   |   |--- class: 1
|   |   |   |   |   |--- Age > 23.50
|   |   |   |   |   |   |--- Age <= 27.50
|   |   |   |   |   |   |   |--- Age <= 25.50
|   |   |   |   |   |   |   |   |--- Fare <= 13.75
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Fare > 13.75
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Age > 25.50
|   |   |   |   |   |   |   |   |   |--- Fare <= 17.43
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Fare > 17.43
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Age > 27.50
|   |   |   |   |   |   |   |   |--- Age <= 37.00
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Age > 37.00
|   |   |   |   |   |   |   |   |   |--- Age <= 39.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Age > 39.00
|   |   |   |   |   |   |   |   |   |   |--- Fare <= 25.96
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Fare > 25.96
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |--- Age > 56.00
|   |   |   |   |   |   |   |   |--- Age <= 57.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Age > 57.50
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |--- Fare > 28.23
|   |   |   |--- class: 0
|   |--- Fare > 28.86
```