

LAMPIRAN

```
#import library yang dibutuhkan
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
#import excel ke dataset
data = pd.read_excel('dataset.xlsx')
data
```

	KI	LI	SI	TT	CB
0	39.6	-3.66	-2.65	46.6	1
1	34.6	-1.46	0.64	42.0	1
2	39.0	-3.74	-0.14	43.7	1
3	34.6	-0.68	0.43	41.0	0
4	36.6	-4.73	-1.88	46.2	1
...	...	...	...	...	...
1747	37.2	-4.71	-0.50	42.2	1
1748	39.8	-4.58	-1.59	44.2	1
1749	37.2	-4.42	-1.82	45.0	1
1750	36.2	-3.23	-1.38	43.4	1
1751	37.6	-4.23	-1.28	43.9	1

1752 rows x 5 columns

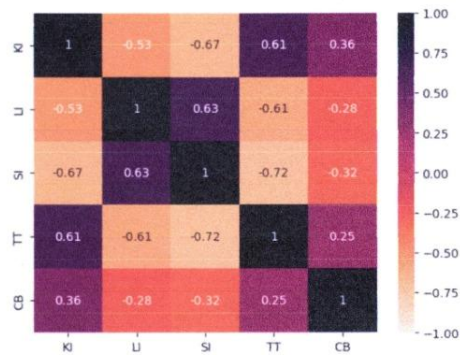
Langkah berikutnya: [Buat kode dengan data](#) [New interactive sheet](#)

```
data.corr()
```

	KI	LI	SI	TT	CB
KI	1.000000	-0.525642	-0.673139	0.611822	0.355262
LI	-0.525642	1.000000	0.627728	-0.608243	-0.277486
SI	-0.673139	0.627728	1.000000	-0.717266	-0.318089
TT	0.611822	-0.608243	-0.717266	1.000000	0.245986
CB	0.355262	-0.277486	-0.318089	0.245986	1.000000

```
import seaborn as sns
```

```
sns.heatmap(data.corr(), vmin=-1, vmax=1,
annot=True, cmap="rocket_r")
plt.show()
```



```
#memanggil data 5 teratas
data.head()
```

	KI	LI	SI	TT	CB
0	39.6	-3.66	-2.65	46.6	1
1	34.6	-1.46	0.64	42.0	1
2	39.0	-3.74	-0.14	43.7	1
3	34.6	-0.68	0.43	41.0	0
4	36.6	-4.73	-1.88	46.2	1

Langkah berikutnya: [Buat kode dengan data](#) [New interactive sheet](#)

```
# menampilkan informasi data
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1752 entries, 0 to 1751
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  --
 0    KI      1746 non-null      float64
 1    LI      1752 non-null      float64
 2    SI      1751 non-null      float64
 3    TT      1750 non-null      float64
 4    CB      1752 non-null      int64
dtypes: float64(4), int64(1)
memory usage: 68.6 KB
```

```
#melihat missing value)
data_miss=data[data.isna().any(axis=1)]
data_miss
```

	KI	LI	SI	TT	CB
89	NaN	23.70	21.08	NaN	1
635	NaN	-1.80	0.60	43.1	0
860	NaN	3.06	7.06	31.9	0
862	NaN	2.89	8.02	32.6	0
864	NaN	2.70	7.01	34.8	0
1100	NaN	-2.13	NaN	NaN	1

Langkah berikutnya: [Buat kode dengan data_miss](#) [New interactive sheet](#)

```
# melihat jumlah data missing
print(data.isnull().sum())
```

```
KI    6
LI    0
SI    1
TT    2
CB    0
dtype: int64
```

```
#menghapus baris yang kosong
data=data.dropna()
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1746 entries, 0 to 1751
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ---
 0   KI      1746 non-null    float64
 1   LI      1746 non-null    float64
 2   SI      1746 non-null    float64
 3   TT      1746 non-null    float64
 4   CB      1746 non-null    int64
dtypes: float64(4), int64(1)
memory usage: 81.8 KB
```

```
# melihat jumlah data missing
print(data.isnull().sum())
```

```
KI    0
LI    0
SI    0
TT    0
CB    0
dtype: int64
```

```
#memanggil data 5 teratas
data.head()
```

	KI	LI	SI	TT	CB
0	39.6	-3.66	-2.65	46.6	1
1	34.6	-1.46	0.64	42.0	1
2	39.0	-3.74	-0.14	43.7	1
3	34.6	-0.68	0.43	41.0	0
4	36.6	-4.73	-1.88	46.2	1

Langkah berikutnya: [Buat kode dengan data](#) [New interactive sheet](#)

```
# Variabel independen
x = data[['KI', 'LI', 'SI', 'TT']]
x.head()
```

	KI	LI	SI	TT
0	39.6	-3.66	-2.65	46.6
1	34.6	-1.46	0.64	42.0
2	39.0	-3.74	-0.14	43.7
3	34.6	-0.68	0.43	41.0
4	36.6	-4.73	-1.88	46.2

Langkah berikutnya: [Buat kode dengan x](#) [New interactive sheet](#)

```
# Variabel dependen
y = data['CB']
y.head()
```

skripsi_Gaib.ipynb - Colab

Metode KNN

<https://colab.research.google.com/drive/1gGteqveFtGLEixBLIxA7ijYr4CQ2YTJ#scrollTo=Jo6u8Anhlufy&printMode=true>

macro avg	0.72	0.69	0.70	350
weighted avg	0.75	0.75	0.75	350

```
accuracy_score: 0.7542857142857143
Precision: 0.6555555555555556
Recall: 0.5175438596491229
F1-Score: 0.5784313725490197
```

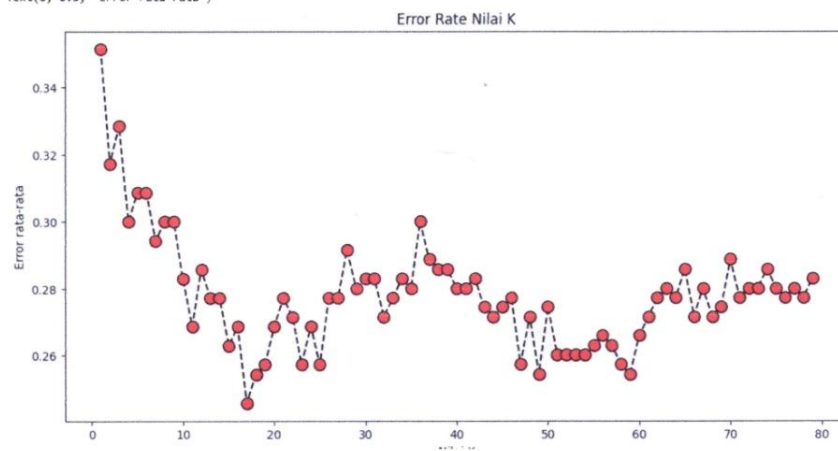
```
#menentukan nilai k terbaik
import matplotlib.pyplot as plt

error = []

# Calculating error for K values between 1 and 80
for i in range(1, 80):
    knn = KNeighborsClassifier(n_neighbors=i)
    knn.fit(x_train, y_train)
    pred_i = knn.predict(x_test)
    error.append(np.mean(pred_i != y_test))

plt.figure(figsize=(12, 6))
plt.plot(range(1, 80), error, color='black', linestyle='dashed', marker='o',
         markerfacecolor='red', markersize=10)
plt.title('Error Rate Nilai K')
plt.xlabel('Nilai K')
plt.ylabel('Error rata-rata')
```

```
Text(0, 0.5, 'Error rata-rata')
```



[illegible]

