Nama : Prawira Setia Ramdhani

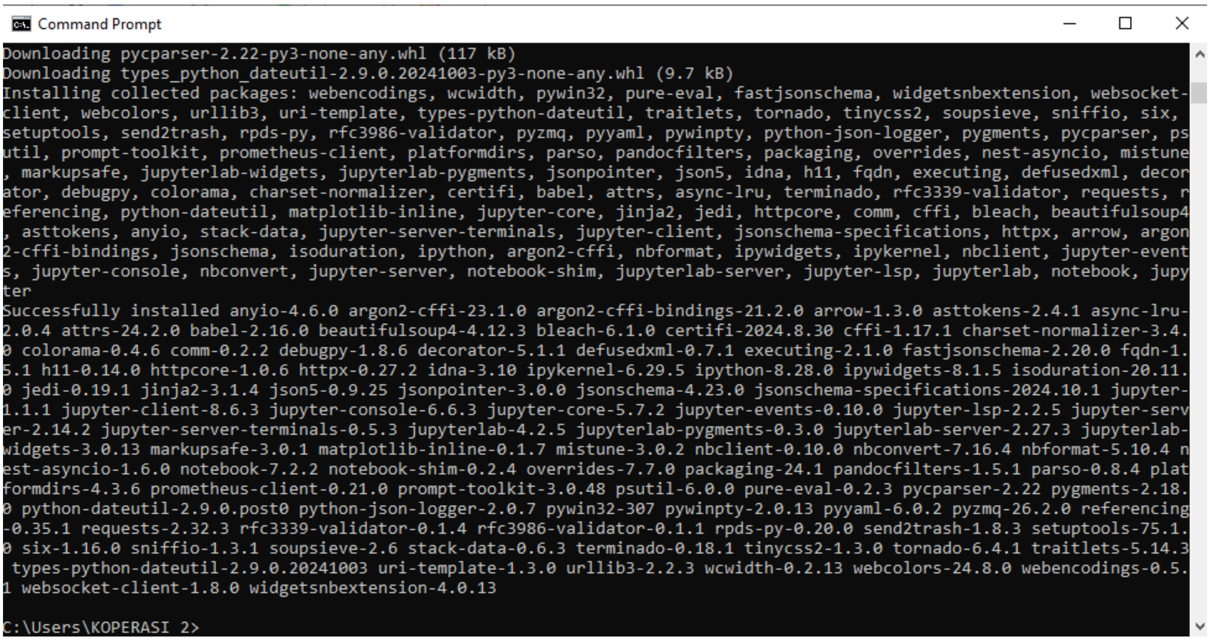
Kelas : INF – A2

NPM : 41155050210062

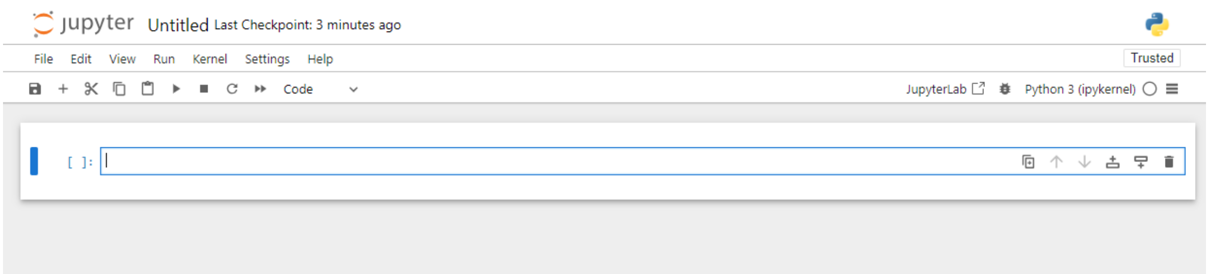
Tugas Pertemuan 1

1.Instalasi Jupyter Notebook

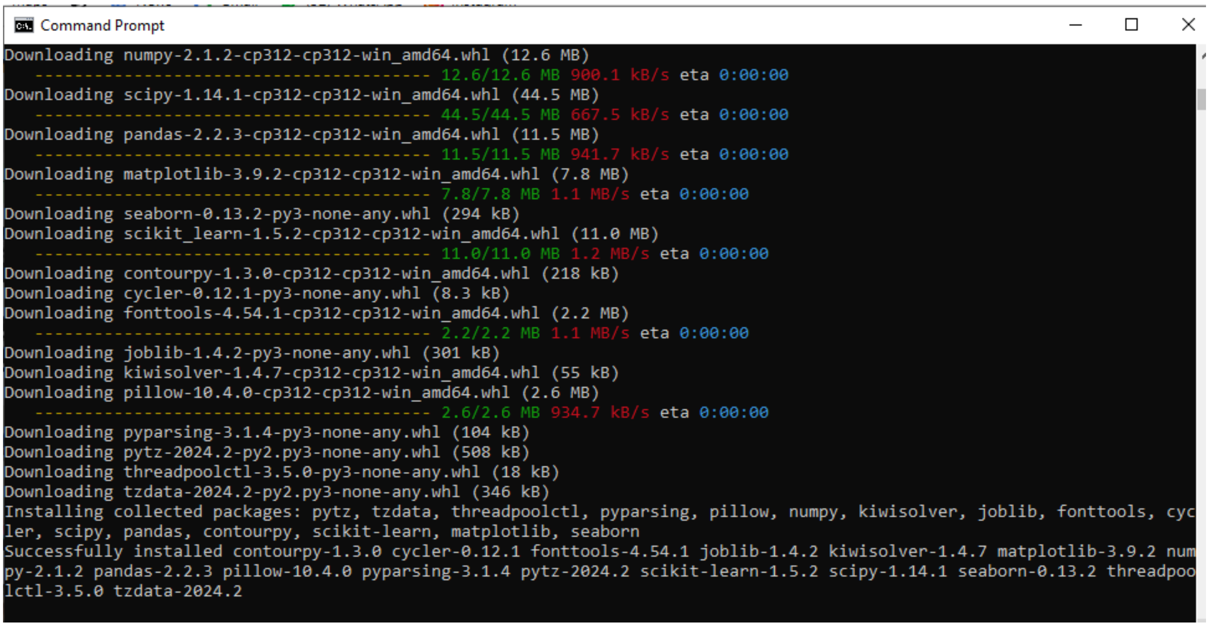
1. Jupyter Notebook



```
Downloading pycparser-2.22-py3-none-any.whl (117 kB)
Downloading types_python_dateutil-2.9.0.20241003-py3-none-any.whl (9.7 kB)
Installing collected packages: webencodings, wcwidth, pywin32, pure-eval, fastjsonschema, widgetsnbextension, websocket-
client, webcolors, urllib3, uri-template, types-python-dateutil, traitlets, tornado, tinycss2, soupsieve, sniffio, six,
setuptools, send2trash, rpds-py, rfc3986-validator, pyzmq, pyyaml, pywinpty, python-json-logger, pygments, pycparser, ps
util, prompt-toolkit, prometheus-client, platformdirs, parso, pandocfilters, packaging, overrides, nest-asyncio, mistune
, markupsafe, jupyterlab-widgets, jupyterlab-pygments, jsonpointer, json5, idna, h11, fqdn, executing, defusedxml, decor
ator, debugpy, colorama, charset-normalizer, certifi, babel, attrs, async-lru, terminado, rfc3339-validator, requests, r
eferencing, python-dateutil, matplotlib-inline, jupyter-core, jinja2, jedi, httpcore, comm, cffi, bleach, beautifulsoup4
, asttokens, anyio, stack-data, jupyter-server-terminals, jupyter-client, jsonschema-specifications, httpx, arrow, argon
2-cffi-bindings, jsonschema, isoduration, ipython, argon2-cffi, nbformat, ipywidgets, ipykernel, nbclient, jupyter-event
s, jupyter-console, nbconvert, jupyter-server, notebook-shim, jupyterlab-server, jupyter-lsp, jupyterlab, notebook, jupy
ter
Successfully installed anyio-4.6.0 argon2-cffi-23.1.0 argon2-cffi-bindings-21.2.0 arrow-1.3.0 asttokens-2.4.1 async-lru-
2.0.4 attrs-24.2.0 babel-2.16.0 beautifulsoup4-4.12.3 bleach-6.1.0 certifi-2024.8.30 cffi-1.17.1 charset-normalizer-3.4.
0 colorama-0.4.6 comm-0.2.2 debugpy-1.8.6 decorator-5.1.1 defusedxml-0.7.1 executing-2.1.0 fastjsonschema-2.20.0 fqdn-1.
5.1 h11-0.14.0 httpcore-1.0.6 httpx-0.27.2 idna-3.10 ipykernel-6.29.5 ipython-8.28.0 ipywidgets-8.1.5 isoduration-20.11.
0 jedi-0.19.1 jinja2-3.1.4 json5-0.9.25 jsonpointer-3.0.0 jsonschema-4.23.0 jsonschema-specifications-2024.10.1 jupyter-
1.1.1 jupyter-client-8.6.3 jupyter-console-6.6.3 jupyter-core-5.7.2 jupyter-events-0.10.0 jupyter-lsp-2.2.5 jupyter-serv
er-2.14.2 jupyter-server-terminals-0.5.3 jupyterlab-4.2.5 jupyterlab-pygments-0.3.0 jupyterlab-server-2.27.3 jupyterlab-
widgets-3.0.13 markupsafe-3.0.1 matplotlib-inline-0.1.7 mistune-3.0.2 nbclient-0.10.0 nbconvert-7.16.4 nbformat-5.10.4 n
est-asyncio-1.6.0 notebook-7.2.2 notebook-shim-0.2.4 overrides-7.7.0 packaging-24.1 pandocfilters-1.5.1 parso-0.8.4 plat
formdirs-4.3.6 prometheus-client-0.21.0 prompt-toolkit-3.0.48 psutil-6.0.0 pure-eval-0.2.3 pycparser-2.22 pygments-2.18.
0 python-dateutil-2.9.0.post0 python-json-logger-2.0.7 pywin32-307 pywinpty-2.0.13 pyyaml-6.0.2 pyzmq-26.2.0 referencing
-0.35.1 requests-2.32.3 rfc3339-validator-0.1.4 rfc3986-validator-0.1.1 rpds-py-0.20.0 send2trash-1.8.3 setuptools-75.1.
0 six-1.16.0 sniffio-1.3.1 soupsieve-2.6 stack-data-0.6.3 terminado-0.18.1 tinycss2-1.3.0 tornado-6.4.1 traitlets-5.14.3
 types-python-dateutil-2.9.0.20241003 uri-template-1.3.0 urllib3-2.2.3 wcwidth-0.2.13 webcolors-24.8.0 webencodings-0.5.
1 websocket-client-1.8.0 widgetsnbextension-4.0.13

C:\Users\KOPERASI 2>
```

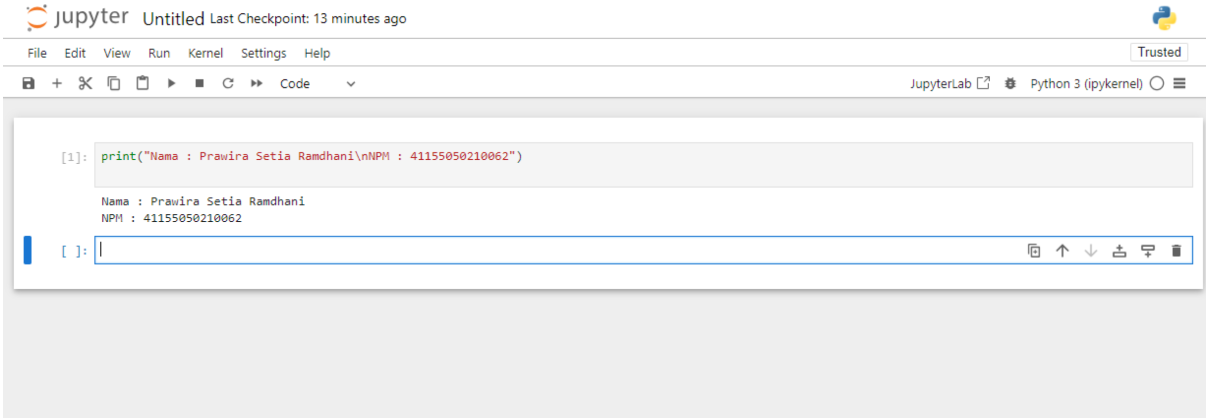


2. Library Python



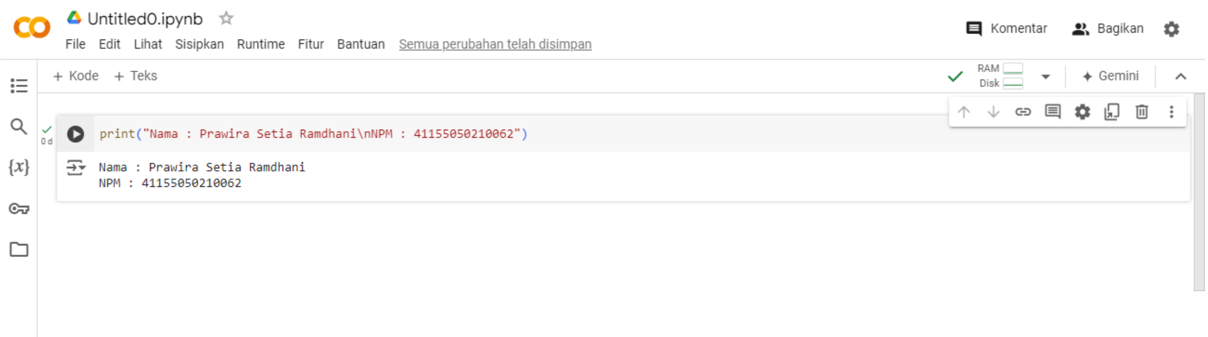
```
Downloading numpy-2.1.2-cp312-cp312-win_amd64.whl (12.6 MB)
   ---------------------------------------- 12.6/12.6 MB 900.1 kB/s eta 0:00:00
Downloading scipy-1.14.1-cp312-cp312-win_amd64.whl (44.5 MB)
   ---------------------------------------- 44.5/44.5 MB 667.5 kB/s eta 0:00:00
Downloading pandas-2.2.3-cp312-cp312-win_amd64.whl (11.5 MB)
   ---------------------------------------- 11.5/11.5 MB 941.7 kB/s eta 0:00:00
Downloading matplotlib-3.9.2-cp312-cp312-win_amd64.whl (7.8 MB)
   ---------------------------------------- 7.8/7.8 MB 1.1 MB/s eta 0:00:00
Downloading seaborn-0.13.2-py3-none-any.whl (294 kB)
Downloading scikit_learn-1.5.2-cp312-cp312-win_amd64.whl (11.0 MB)
   ---------------------------------------- 11.0/11.0 MB 1.2 MB/s eta 0:00:00
Downloading contourpy-1.3.0-cp312-cp312-win_amd64.whl (218 kB)
Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Downloading fonttools-4.54.1-cp312-cp312-win_amd64.whl (2.2 MB)
   ---------------------------------------- 2.2/2.2 MB 1.1 MB/s eta 0:00:00
Downloading joblib-1.4.2-py3-none-any.whl (301 kB)
Downloading kiwisolver-1.4.7-cp312-cp312-win_amd64.whl (55 kB)
Downloading pillow-10.4.0-cp312-cp312-win_amd64.whl (2.6 MB)
   ---------------------------------------- 2.6/2.6 MB 934.7 kB/s eta 0:00:00
Downloading pyparsing-3.1.4-py3-none-any.whl (104 kB)
Downloading pytz-2024.2-py2.py3-none-any.whl (508 kB)
Downloading threadpoolctl-3.5.0-py3-none-any.whl (18 kB)
Downloading tzdata-2024.2-py2.py3-none-any.whl (346 kB)
Installing collected packages: pytz, tzdata, threadpoolctl, pyparsing, pillow, numpy, kiwisolver, joblib, fonttools, cyc
ler, scipy, pandas, contourpy, scikit-learn, matplotlib, seaborn
Successfully installed contourpy-1.3.0 cycler-0.12.1 fonttools-4.54.1 joblib-1.4.2 kiwisolver-1.4.7 matplotlib-3.9.2 num
py-2.1.2 pandas-2.2.3 pillow-10.4.0 pyparsing-3.1.4 pytz-2024.2 scikit-learn-1.5.2 scipy-1.14.1 seaborn-0.13.2 threadpoo
lctl-3.5.0 tzdata-2024.2
```

3. Hasil



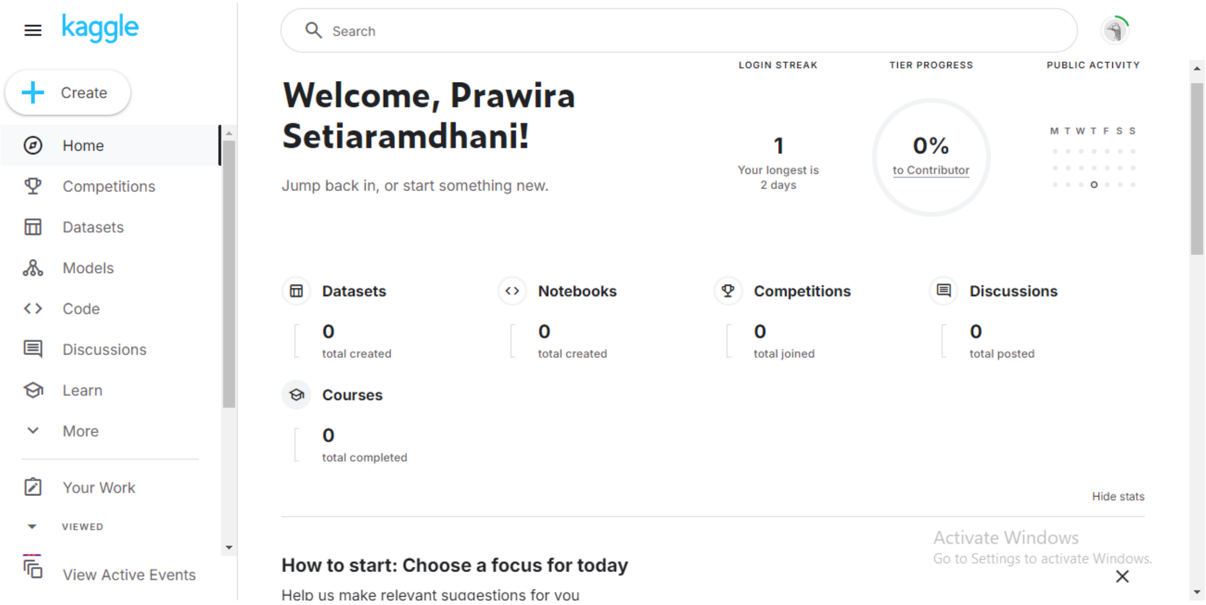
```
[1]: print("Nama : Prawira Setia Ramdhani\nNPM : 41155050210062")

     Nama : Prawira Setia Ramdhani
     NPM : 41155050210062
[ ]:
```

2. Menggunakan Google Collab



```
print("Nama : Prawira Setia Ramdhani\nNPM : 41155050210062")

Nama : Prawira Setia Ramdhani
NPM : 41155050210062
```

3. Buatlah akun di https://www.kaggle.com/



Welcome, Prawira Setiaramdhani!

Jump back in, or start something new.

LOGIN STREAK: 1 — Your longest is 2 days

TIER PROGRESS: 0% to Contributor

PUBLIC ACTIVITY: M T W T F S S

Datasets	Notebooks	Competitions	Discussions
0 total created	0 total created	0 total joined	0 total posted

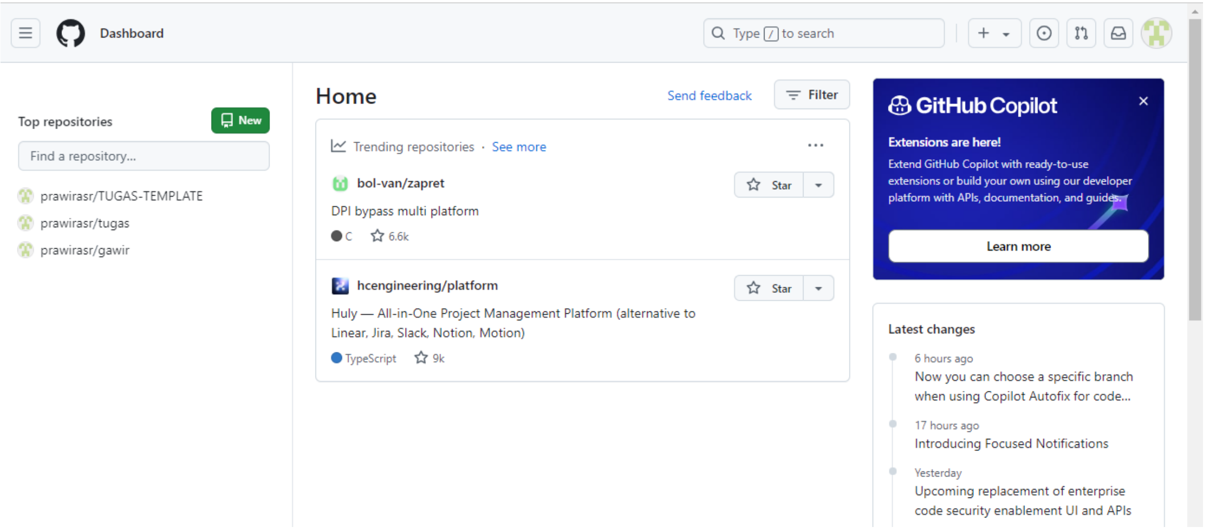
Courses: 0 total completed

How to start: Choose a focus for today

Help us make relevant suggestions for you

Activate Windows
Go to Settings to activate Windows.

4. Buatlah akun di https://github.com/



Top repositories

- prawirasr/TUGAS-TEMPLATE
- prawirasr/tugas
- prawirasr/gawir

Home

Trending repositories · See more

bol-van/zapret
DPI bypass multi platform
C ☆ 6.6k

hcengineering/platform
Huly — All-in-One Project Management Platform (alternative to Linear, Jira, Slack, Notion, Motion)
TypeScript ☆ 9k

GitHub Copilot

Extensions are here!

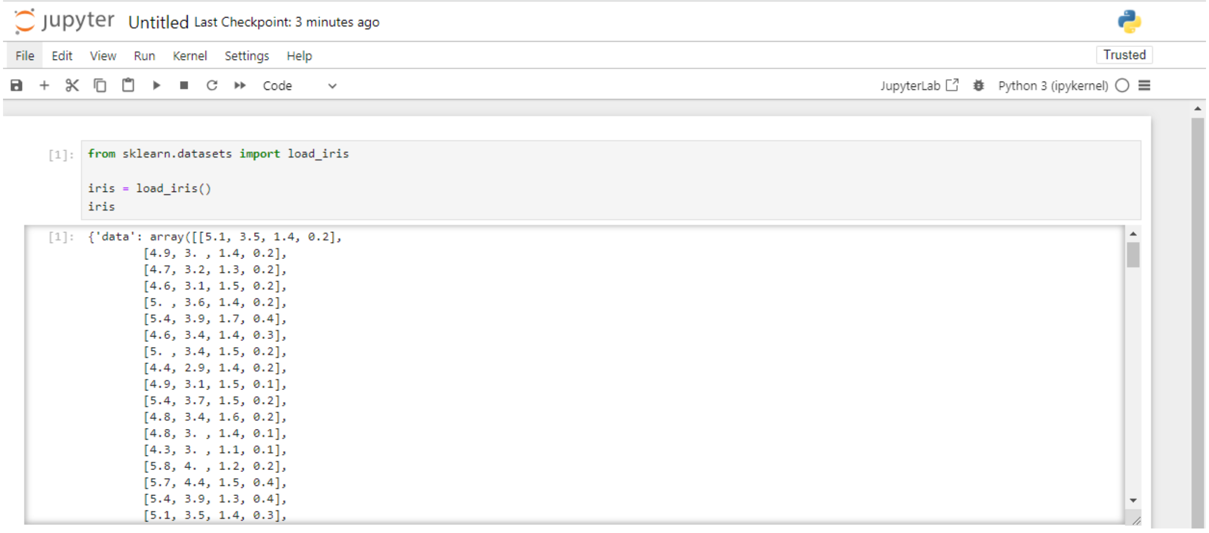
Extend GitHub Copilot with ready-to-use extensions or build your own using our developer platform with APIs, documentation, and guides.

Latest changes

- 6 hours ago — Now you can choose a specific branch when using Copilot Autofix for code...
- 17 hours ago — Introducing Focused Notifications
- Yesterday — Upcoming replacement of enterprise code security enablement UI and APIs

5. Lakukan praktek dari https://youtu.be/mSO2hJln0OY?feature=shared

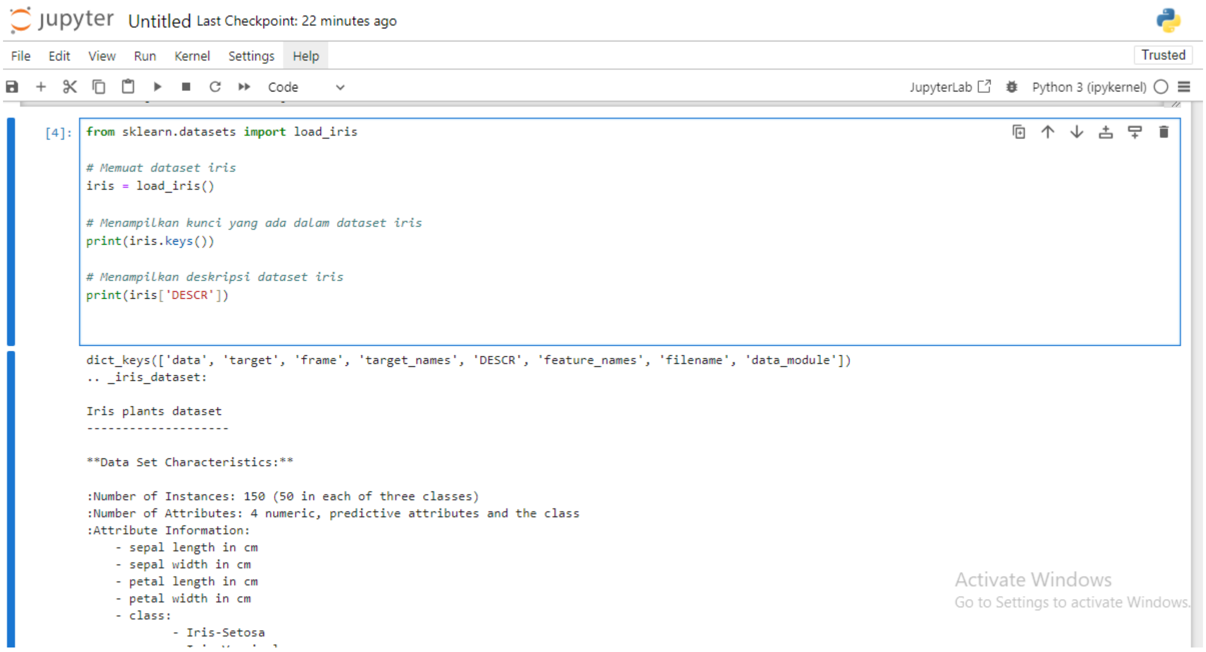
1. Load sample dataset


jupyter Untitled Last Checkpoint: 3 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)

[1]:
```
from sklearn.datasets import load_iris

iris = load_iris()
iris
```

[1]:
```
{'data': array([[5.1, 3.5, 1.4, 0.2],
        [4.9, 3. , 1.4, 0.2],
        [4.7, 3.2, 1.3, 0.2],
        [4.6, 3.1, 1.5, 0.2],
        [5. , 3.6, 1.4, 0.2],
        [5.4, 3.9, 1.7, 0.4],
        [4.6, 3.4, 1.4, 0.3],
        [5. , 3.4, 1.5, 0.2],
        [4.4, 2.9, 1.4, 0.2],
        [4.9, 3.1, 1.5, 0.1],
        [5.4, 3.7, 1.5, 0.2],
        [4.8, 3.4, 1.6, 0.2],
        [4.8, 3. , 1.4, 0.1],
        [4.3, 3. , 1.1, 0.1],
        [5.8, 4. , 1.2, 0.2],
        [5.7, 4.4, 1.5, 0.4],
        [5.4, 3.9, 1.3, 0.4],
        [5.1, 3.5, 1.4, 0.3],
```

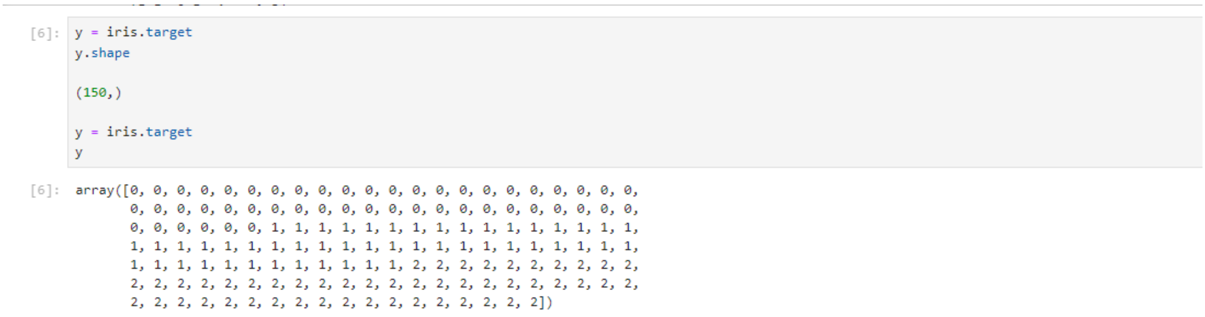
2. Metadata | Deskripsi dari sample dataset


jupyter Untitled Last Checkpoint: 22 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
Activate Windows
Go to Settings to activate Windows.

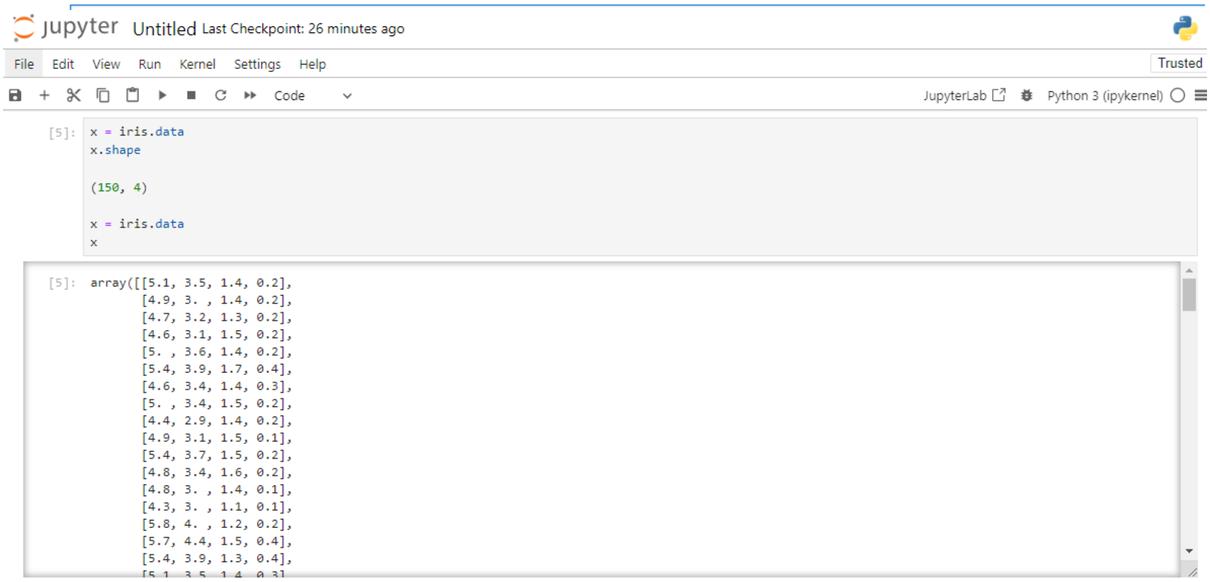
[4]:
```
from sklearn.datasets import load_iris

# Memuat dataset iris
iris = load_iris()

# Menampilkan kunci yang ada dalam dataset iris
print(iris.keys())

# Menampilkan deskripsi dataset iris
print(iris['DESCR'])
```

```
dict_keys(['data', 'target', 'frame', 'target_names', 'DESCR', 'feature_names', 'filename', 'data_module'])
.. _iris_dataset:

Iris plants dataset
--------------------

**Data Set Characteristics:**

:Number of Instances: 150 (50 in each of three classes)
:Number of Attributes: 4 numeric, predictive attributes and the class
:Attribute Information:
    - sepal length in cm
    - sepal width in cm
    - petal length in cm
    - petal width in cm
    - class:
            - Iris-Setosa
```

3. Explanatory & Response variables | Fatures & Target

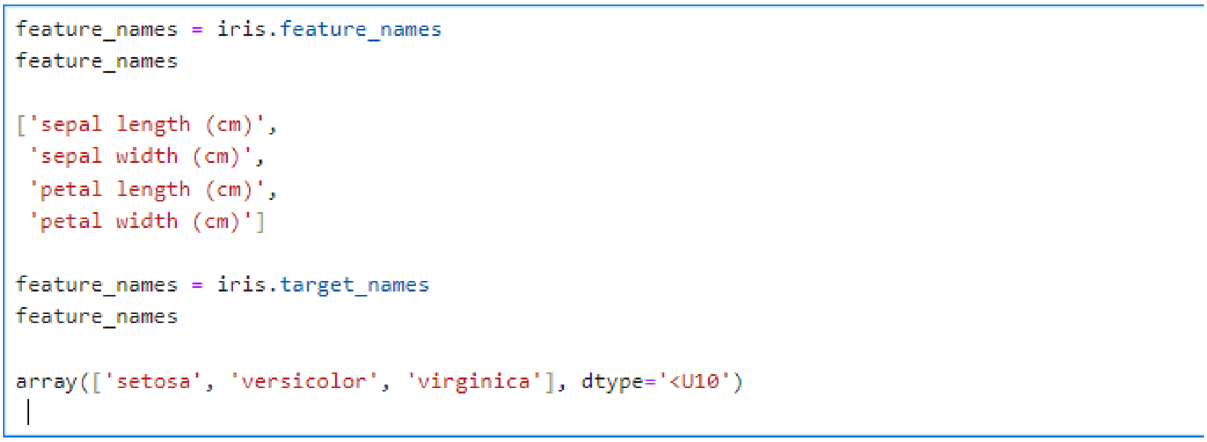


```
[6]: y = iris.target
     y.shape

     (150,)

     y = iris.target
     y

[6]: array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
            1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
            1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
            2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
            2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```



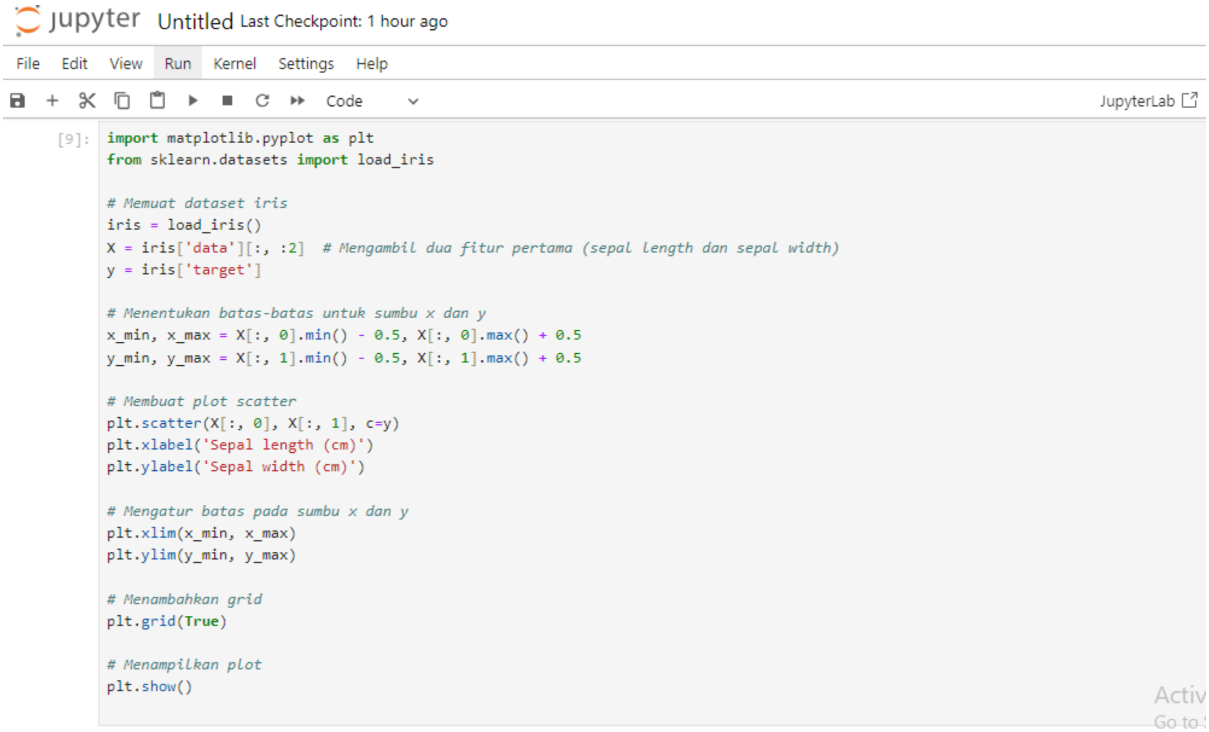
```
[5]: x = iris.data
     x.shape

     (150, 4)

     x = iris.data
     x

[5]: array([[5.1, 3.5, 1.4, 0.2],
            [4.9, 3. , 1.4, 0.2],
            [4.7, 3.2, 1.3, 0.2],
            [4.6, 3.1, 1.5, 0.2],
            [5. , 3.6, 1.4, 0.2],
            [5.4, 3.9, 1.7, 0.4],
            [4.6, 3.4, 1.4, 0.3],
            [5. , 3.4, 1.5, 0.2],
            [4.4, 2.9, 1.4, 0.2],
            [4.9, 3.1, 1.5, 0.1],
            [5.4, 3.7, 1.5, 0.2],
            [4.8, 3.4, 1.6, 0.2],
            [4.8, 3. , 1.4, 0.1],
            [4.3, 3. , 1.1, 0.1],
            [5.8, 4. , 1.2, 0.2],
            [5.7, 4.4, 1.5, 0.4],
            [5.4, 3.9, 1.3, 0.4],
            [5.1, 3.5, 1.4, 0.3],
```

4. Feature & Target Names

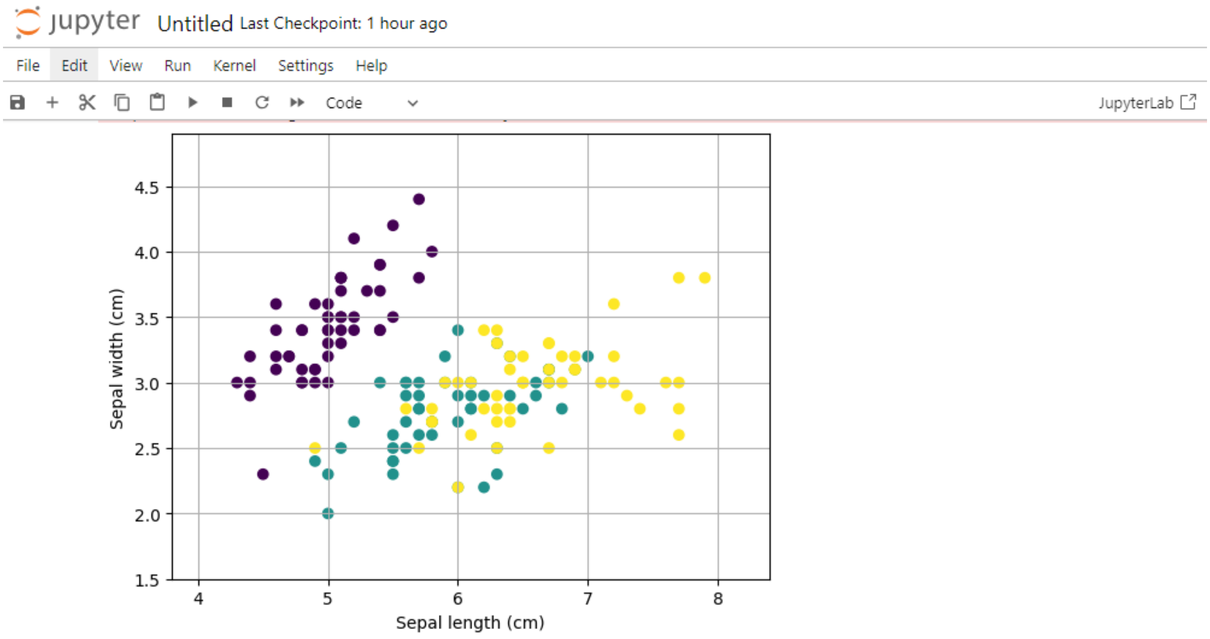


```
feature_names = iris.feature_names
feature_names

['sepal length (cm)',
 'sepal width (cm)',
 'petal length (cm)',
 'petal width (cm)']

feature_names = iris.target_names
feature_names

array(['setosa', 'versicolor', 'virginica'], dtype='<U10')
```

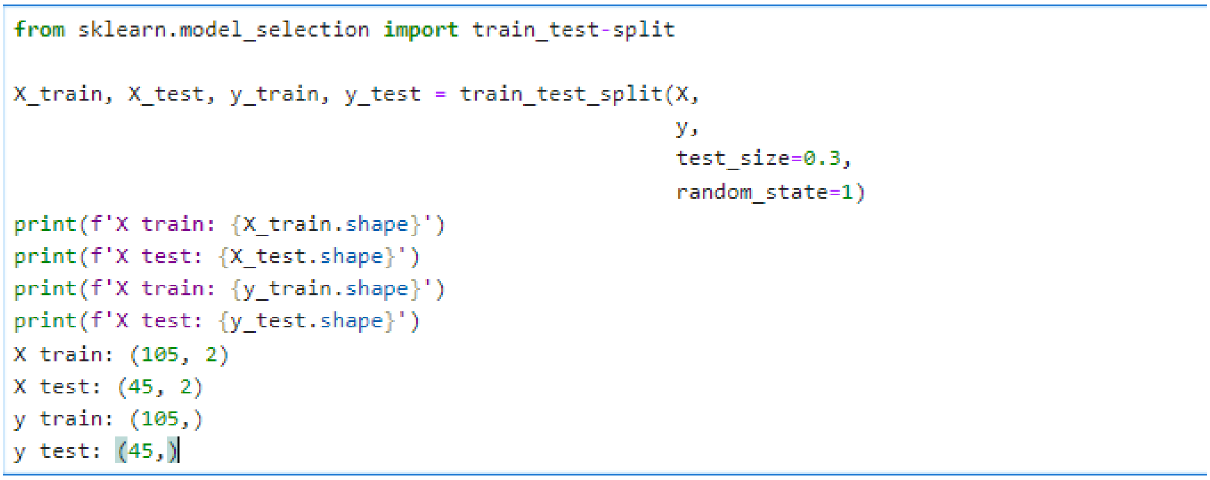
5. Visualisasi Data


jupyter Untitled Last Checkpoint: 1 hour ago
File Edit View Run Kernel Settings Help
Code
JupyterLab
[9]:

```python
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris

# Memuat dataset iris
iris = load_iris()
X = iris['data'][:, :2]  # Mengambil dua fitur pertama (sepal length dan sepal width)
y = iris['target']

# Menentukan batas-batas untuk sumbu x dan y
x_min, x_max = X[:, 0].min() - 0.5, X[:, 0].max() + 0.5
y_min, y_max = X[:, 1].min() - 0.5, X[:, 1].max() + 0.5

# Membuat plot scatter
plt.scatter(X[:, 0], X[:, 1], c=y)
plt.xlabel('Sepal length (cm)')
plt.ylabel('Sepal width (cm)')

# Mengatur batas pada sumbu x dan y
plt.xlim(x_min, x_max)
plt.ylim(y_min, y_max)

# Menambahkan grid
plt.grid(True)

# Menampilkan plot
plt.show()
```

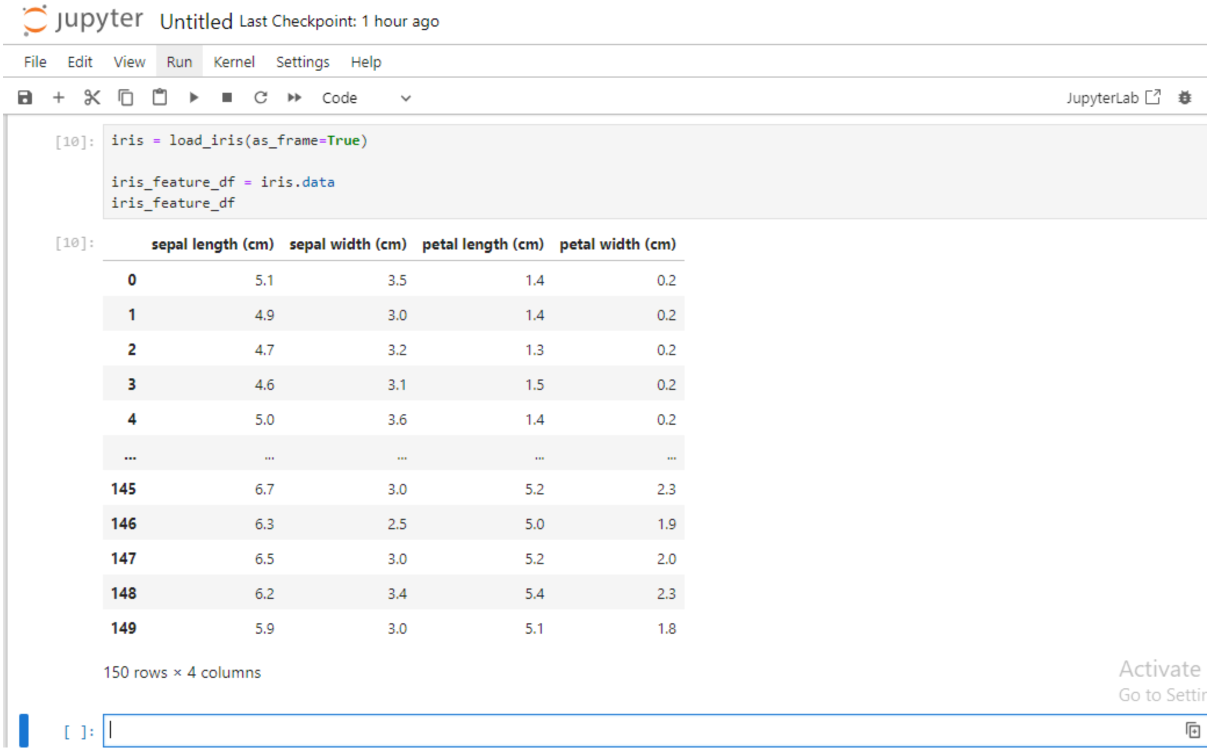


6. Training Set & Testing Set



```
from sklearn.model_selection import train_test-split

X_train, X_test, y_train, y_test = train_test_split(X,
                                                    y,
                                                    test_size=0.3,
                                                    random_state=1)
print(f'X train: {X_train.shape}')
print(f'X test: {X_test.shape}')
print(f'X train: {y_train.shape}')
print(f'X test: {y_test.shape}')
X train: (105, 2)
X test: (45, 2)
y train: (105,)
y test: (45,)
```

7. Load sample dataset sebagai Pandas Data Frame



```
[10]: iris = load_iris(as_frame=True)

      iris_feature_df = iris.data
      iris_feature_df
```

[10]:

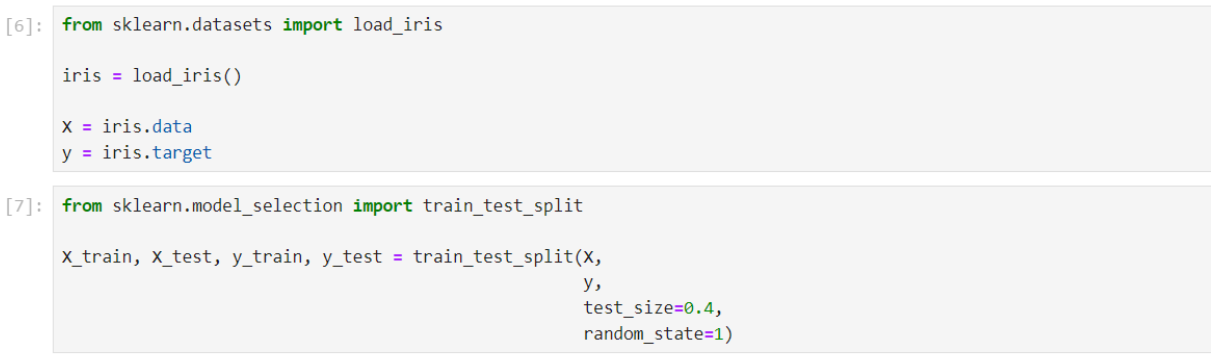
	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
...	...	...	...	...
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

150 rows × 4 columns

[ ]:

6. Lakukan praktek dari https://youtu.be/iREcHrtDLo?feature=shared

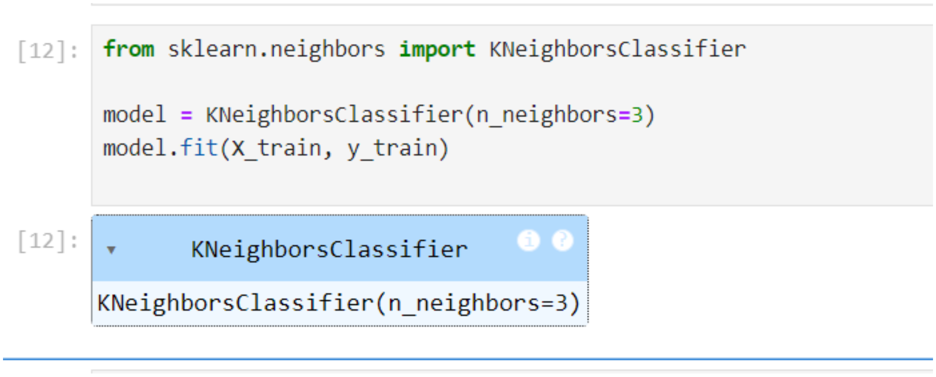
1. Persiapan dataset | Loading & splitting dataset



```
[6]: from sklearn.datasets import load_iris

     iris = load_iris()

     X = iris.data
     y = iris.target
```

```
[7]: from sklearn.model_selection import train_test_split

     X_train, X_test, y_train, y_test = train_test_split(X,
                                                         y,
                                                         test_size=0.4,
                                                         random_state=1)
```

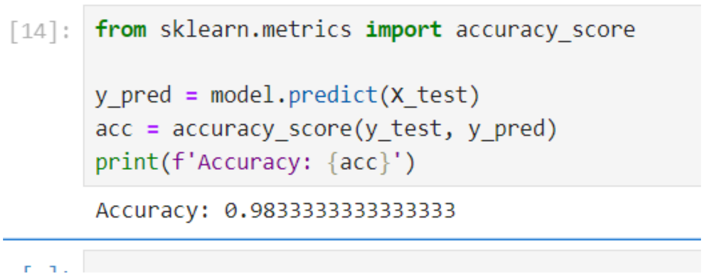
2. Training model Machine Learning



```
[12]: from sklearn.neighbors import KNeighborsClassifier

      model = KNeighborsClassifier(n_neighbors=3)
      model.fit(X_train, y_train)
```

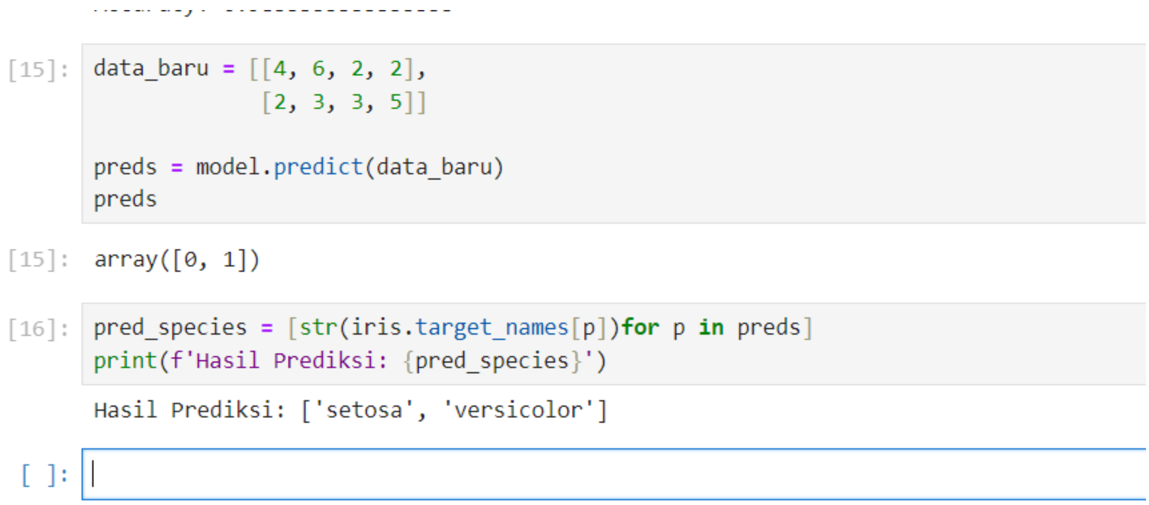
```
[12]: ▾      KNeighborsClassifier
      KNeighborsClassifier(n_neighbors=3)
```

3. Evaluasi model Machine Learning



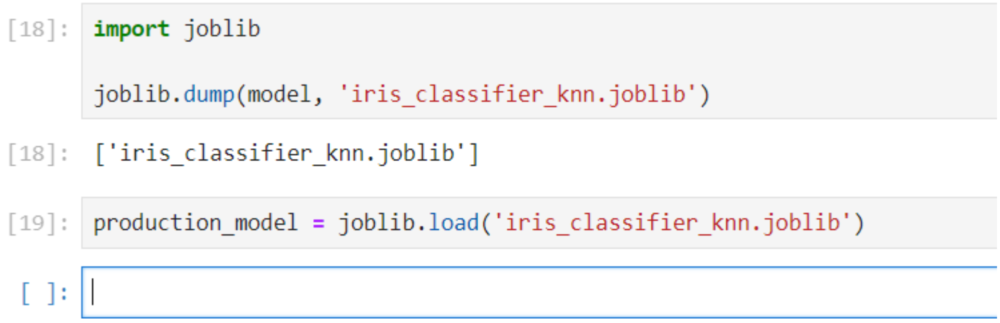
```
[14]: from sklearn.metrics import accuracy_score

      y_pred = model.predict(X_test)
      acc = accuracy_score(y_test, y_pred)
      print(f'Accuracy: {acc}')

      Accuracy: 0.9833333333333333
```

4. Pemanfaatan trained model machine learning



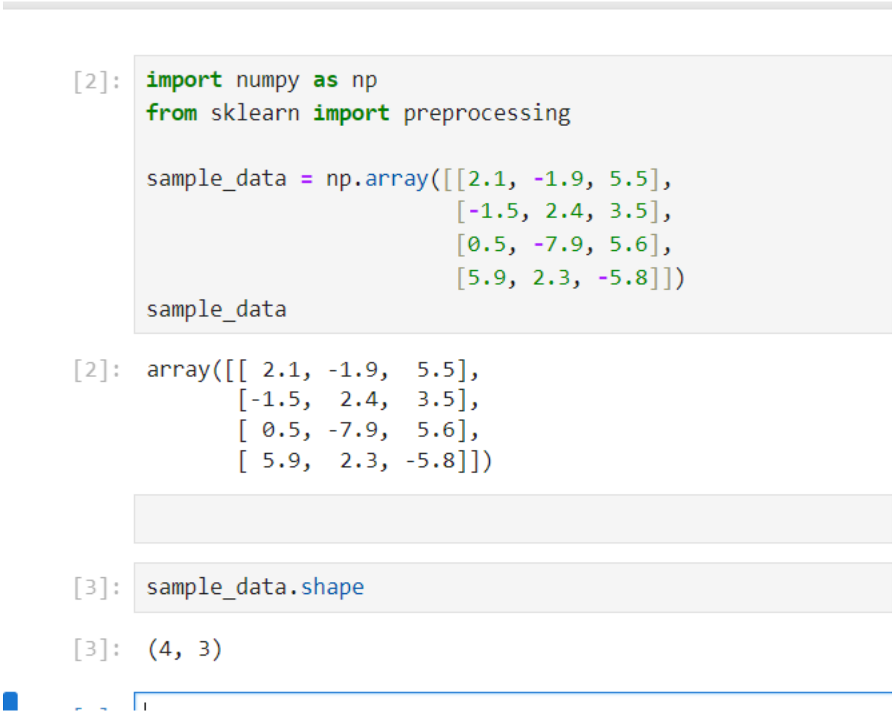
```
[15]: data_baru = [[4, 6, 2, 2],
                   [2, 3, 3, 5]]

      preds = model.predict(data_baru)
      preds

[15]: array([0, 1])

[16]: pred_species = [str(iris.target_names[p])for p in preds]
      print(f'Hasil Prediksi: {pred_species}')

      Hasil Prediksi: ['setosa', 'versicolor']

[ ]: |
```

5. Deploy model Machine learning | dumping dan loading model machine



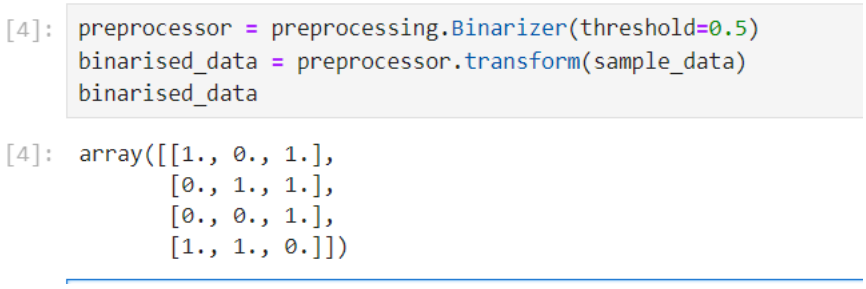
```
[18]: import joblib

      joblib.dump(model, 'iris_classifier_knn.joblib')

[18]: ['iris_classifier_knn.joblib']

[19]: production_model = joblib.load('iris_classifier_knn.joblib')

[ ]: |
```

7. . Lakukan praktek dari https://youtu.be/smNnhEd26Ek?feature=shared .
Praktek tersebut yaitu:
1. Persiapan sample dataset



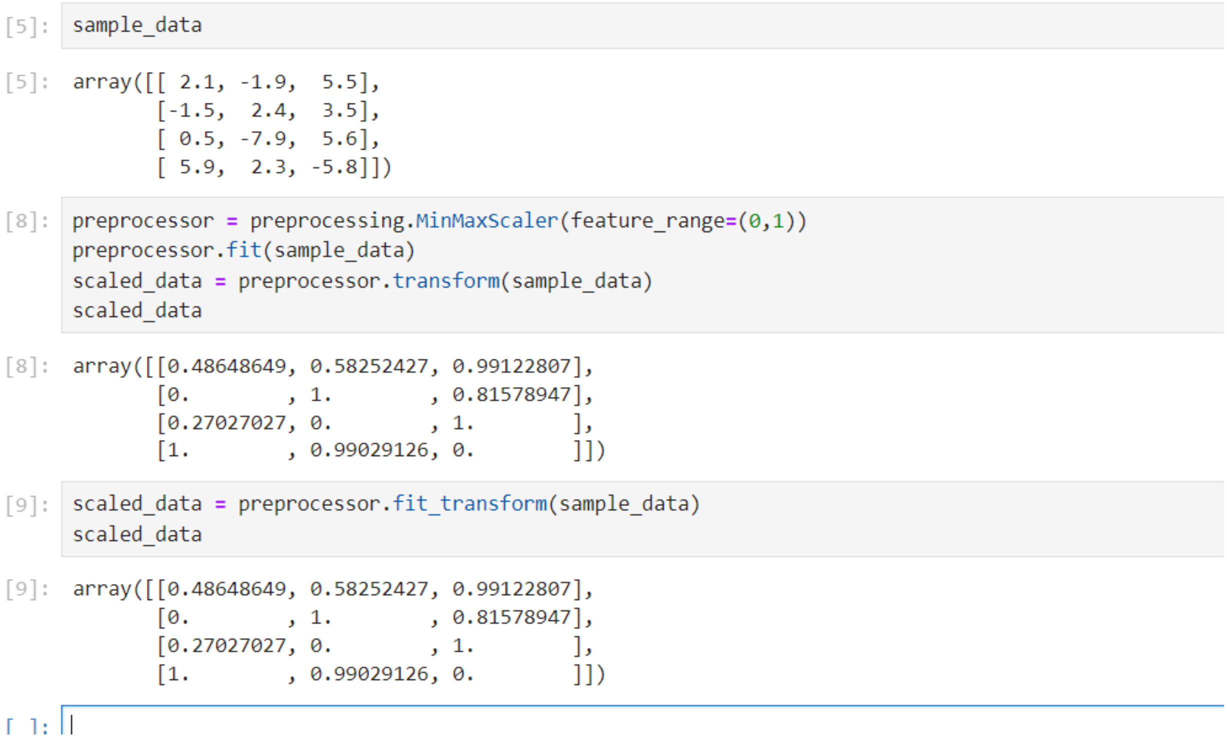
```
[2]: import numpy as np
     from sklearn import preprocessing

     sample_data = np.array([[2.1, -1.9, 5.5],
                             [-1.5, 2.4, 3.5],
                             [0.5, -7.9, 5.6],
                             [5.9, 2.3, -5.8]])
     sample_data

[2]: array([[ 2.1, -1.9,  5.5],
            [-1.5,  2.4,  3.5],
            [ 0.5, -7.9,  5.6],
            [ 5.9,  2.3, -5.8]])

[3]: sample_data.shape

[3]: (4, 3)
```

2. **Teknik data preprocessing 1: binarisation**



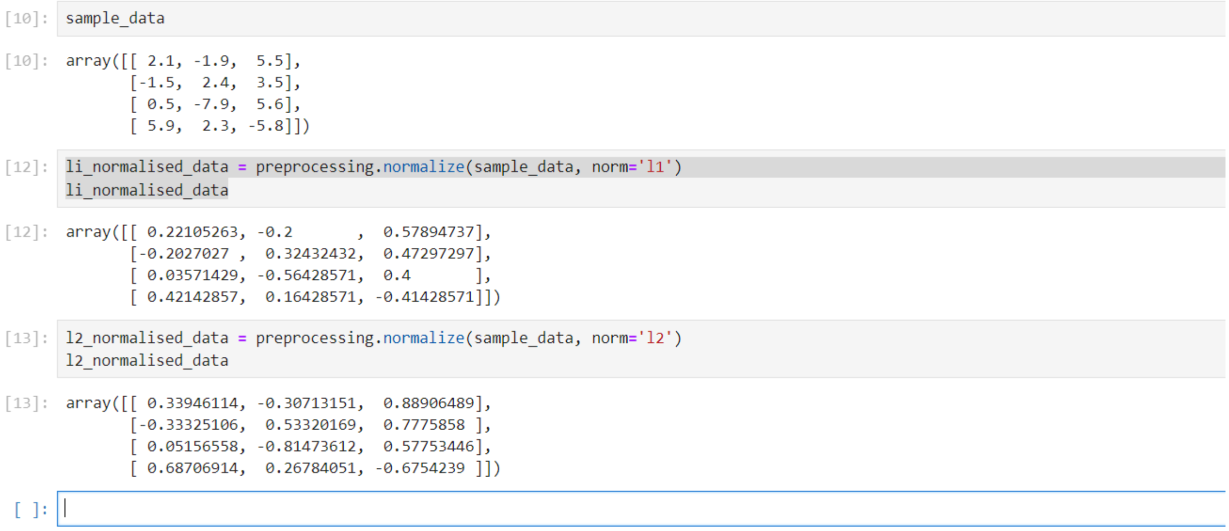
```
[4]: preprocessor = preprocessing.Binarizer(threshold=0.5)
     binarised_data = preprocessor.transform(sample_data)
     binarised_data

[4]: array([[1., 0., 1.],
            [0., 1., 1.],
            [0., 0., 1.],
            [1., 1., 0.]])
```

3. Teknik data preprocessing 2: scaling



```
[5]: sample_data

[5]: array([[ 2.1, -1.9,  5.5],
            [-1.5,  2.4,  3.5],
            [ 0.5, -7.9,  5.6],
            [ 5.9,  2.3, -5.8]])

[8]: preprocessor = preprocessing.MinMaxScaler(feature_range=(0,1))
     preprocessor.fit(sample_data)
     scaled_data = preprocessor.transform(sample_data)
     scaled_data

[8]: array([[0.48648649, 0.58252427, 0.99122807],
            [0.        , 1.        , 0.81578947],
            [0.27027027, 0.        , 1.        ],
            [1.        , 0.99029126, 0.        ]])

[9]: scaled_data = preprocessor.fit_transform(sample_data)
     scaled_data

[9]: array([[0.48648649, 0.58252427, 0.99122807],
            [0.        , 1.        , 0.81578947],
            [0.27027027, 0.        , 1.        ],
            [1.        , 0.99029126, 0.        ]])

[ ]:
```

4. Teknik data preprocessing 3: normalisation



```
[10]: sample_data

[10]: array([[ 2.1, -1.9,  5.5],
             [-1.5,  2.4,  3.5],
             [ 0.5, -7.9,  5.6],
             [ 5.9,  2.3, -5.8]])

[12]: li_normalised_data = preprocessing.normalize(sample_data, norm='l1')
      li_normalised_data

[12]: array([[ 0.22105263, -0.2       ,  0.57894737],
             [-0.2027027 ,  0.32432432,  0.47297297],
             [ 0.03571429, -0.56428571,  0.4       ],
             [ 0.42142857,  0.16428571, -0.41428571]])

[13]: l2_normalised_data = preprocessing.normalize(sample_data, norm='l2')
      l2_normalised_data

[13]: array([[ 0.33946114, -0.30713151,  0.88906489],
             [-0.33325106,  0.53320169,  0.7775858 ],
             [ 0.05156558, -0.81473612,  0.57753446],
             [ 0.68706914,  0.26784051, -0.6754239 ]])

 [ ]: |
```