🔹 1. Importing Required Libraries

```
import pandas as pd
import urllib.request
```

- `pandas`: For data manipulation and analysis.
- `urllib.request`: For accessing data via a URL (though not directly used here; `pandas.read_csv(url)` handles it).

🔹 2. Loading the Dataset

```
url = "http://archive.ics.uci.edu/ml/machine-learning-
databases/iris/iris.data"
column_names =["Sepal_Length", "Sepal_width", "Petal_length", "Petal_Width",
"Class"]
iris = pd.read_csv(url, names=column_names)
iris.head()
```

- Downloads the **Iris dataset**.
- Assigns column names manually.
- Displays the first 5 rows using `head()`.

🔹 3. Basic Data Exploration

```
iris.head()
iris.tail()
iris.index
iris.columns
iris.shape
iris.dtypes
```

- Views first and last rows.
- Inspects the index, column names, shape (rows × columns), and data types.

🔹 4. Data Selection Techniques

```
iris["Sepal_Length"]
iris.iloc[5]
iris[0:3]
iris.loc[:,["Sepal_Length","Sepal_width"]]
iris.iloc[:5,:]
iris.iloc[:,:5]
iris.iloc[:5,:5]
```

These lines show different **indexing and selection techniques**:

- `iloc`: Integer-location-based indexing.
- `loc`: Label-based indexing.

🔹 5. Missing Data Checks

```
iris.isnull().any()
iris.isnull().sum()
iris.isnull().sum().sum()
iris.isnull().sum(axis=1)
```

- Checks for missing values:
  - `any()`: If any column has nulls.
  - `sum()`: Count of nulls in each column and overall.

Output shows there are no missing values.

🔹 6. Data Type Inspection

```
iris.dtypes
```

- Shows each column's data type (float or object).

🔹 7. Type Conversion

```
iris["Petal_length"] = iris["Petal_length"].astype('int')
iris.head()
```

- Converts the `Petal_length` from `float` to `int`.
- This may be done for specific ML algorithms or visual clarity.