

POLITECNICO
MILANO 1863

**DEPARTMENT OF MECHANICAL
ENGINEERING**

Introduction to Visual Studio Code

MATTEO BUGATTI



DIPARTIMENTO DI ECCELLENZA
MIUR 2018-2022



Milano

INTRODUCTION TO VISUAL STUDIO CODE



Visual Studio Code is a source-code editor developed by Microsoft. It can be used with a variety of programming languages, including C++, JavaScript and Python.

Visual Studio Code has many features such as built-in debugging, syntax highlighting, auto-completion, code formatting, and Git integration. It is available for Windows, macOS, and Linux.

TUTORIAL LINK:

<https://code.visualstudio.com/docs/introvideos/basics>

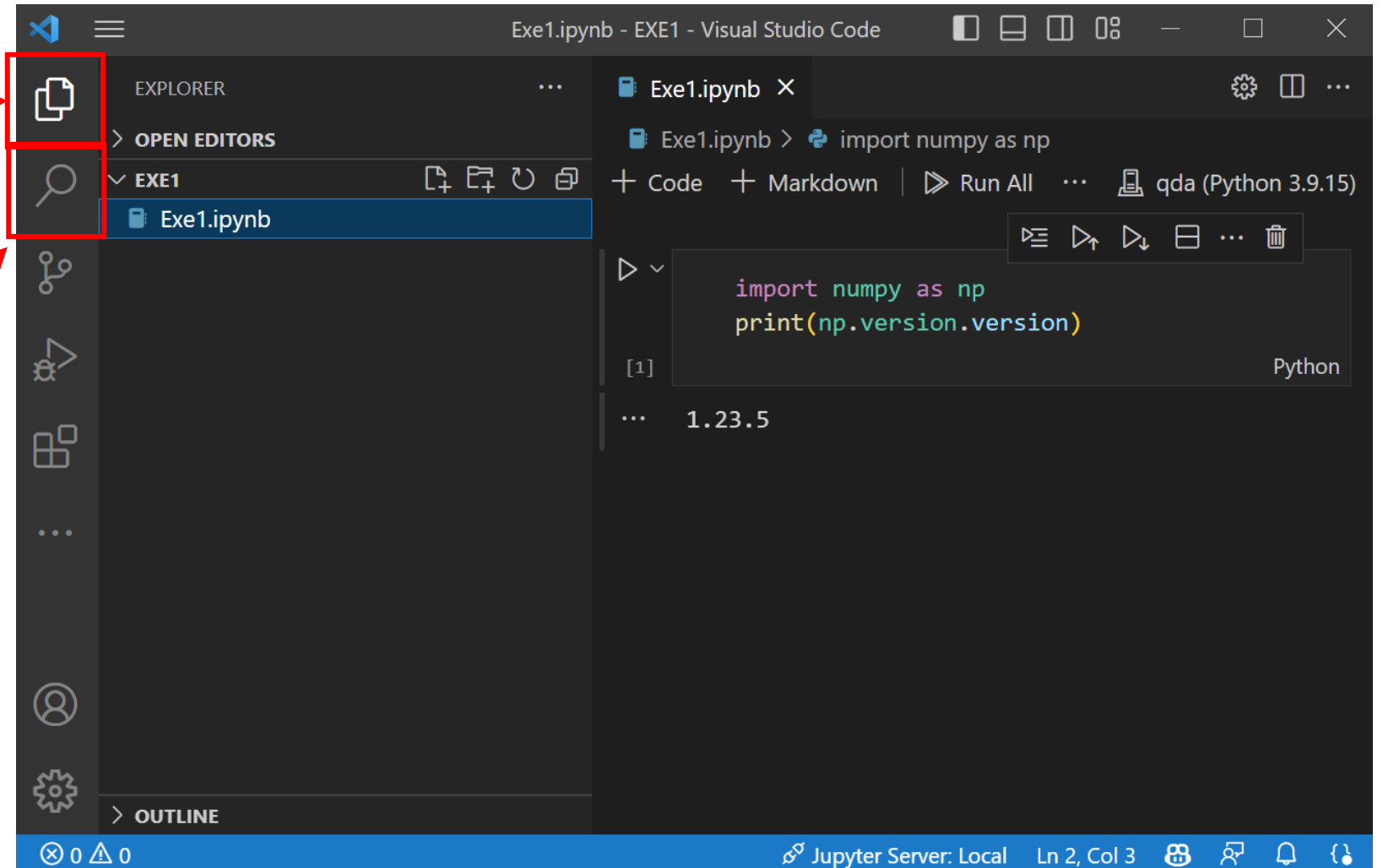
USER INTERFACE TOUR

Explorer

browse the workspace folder,
and all the file contained in it

Search

search, find and substitute
words or files across the
workspace



USER INTERFACE TOUR

Source Control

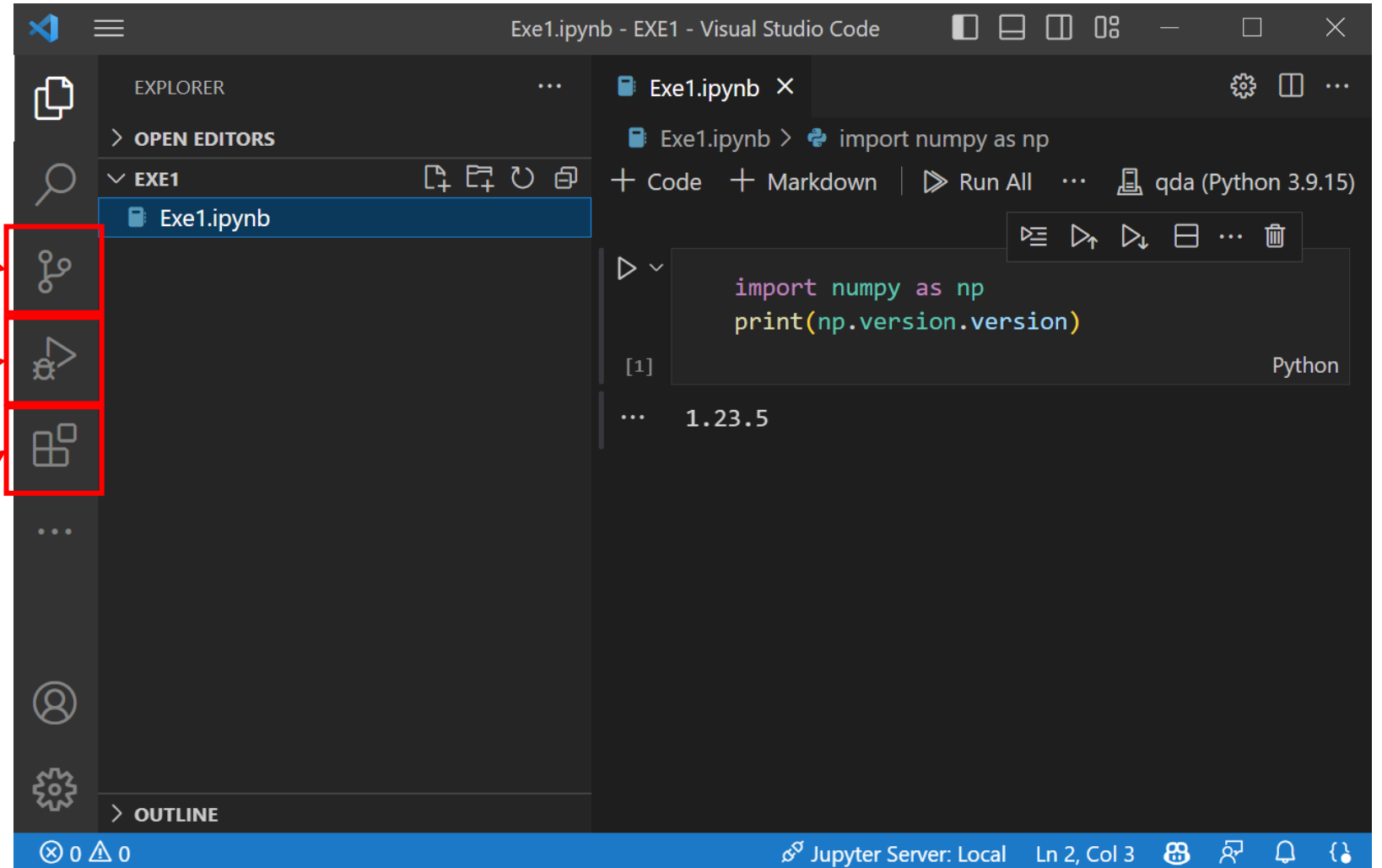
To track changes in code with
Git Hub

Debug

Run and Debug codes
using breakpoints

Extensions

The marketplace for additional
features



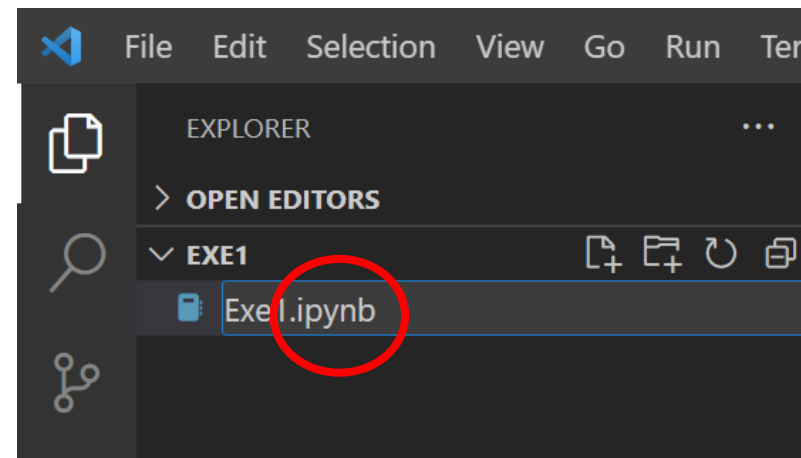
JUPYTER NOTEBOOK

Jupyter Notebook files (extension “.ipynb”) are documents that contain a combination of code, text, and visualizations, all in a single interactive environment.

These files are divided in “**cells**” of code, that the user can run separately rather than running the entire code simultaneously. It allows users to interact with and explore data in real-time.

IMPORTANT

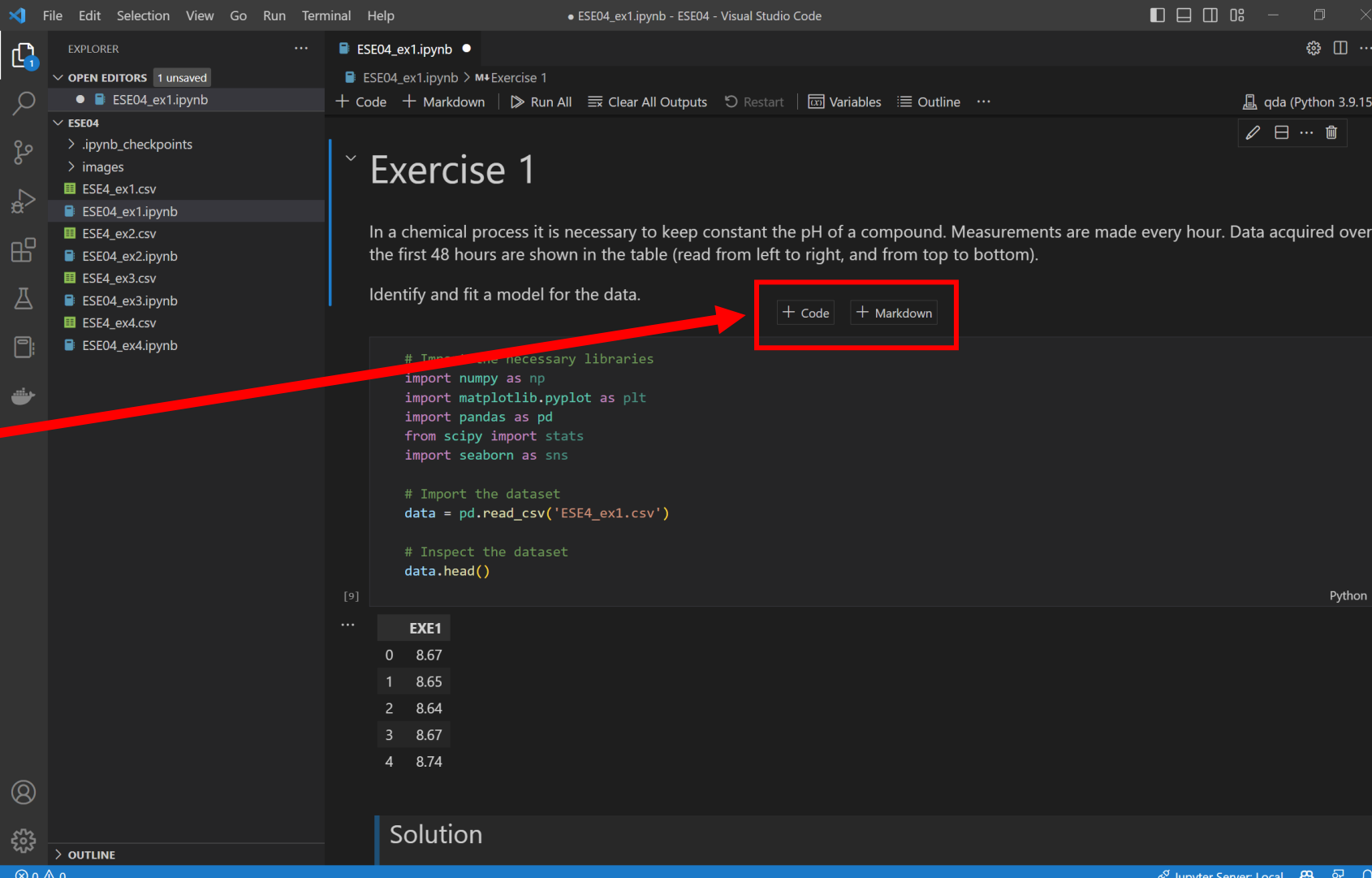
When creating a new file, we must specify the extension .ipynb!



JUPYTER NOTEBOOK

It is possible to add a new cell of code or a new markdown between any cells.

Markdowns can be used to import images, formulas or write simple texts.



The screenshot shows a Jupyter Notebook interface in Visual Studio Code. The Explorer panel on the left shows a file named `ESE04_ex1.ipynb`. The main editor area displays the notebook content, which includes a markdown cell titled "Exercise 1" and a code cell. The markdown cell contains the following text:

In a chemical process it is necessary to keep constant the pH of a compound. Measurements are made every hour. Data acquired over the first 48 hours are shown in the table (read from left to right, and from top to bottom).

Identify and fit a model for the data.

Below the markdown cell, there are two buttons: "+ Code" and "+ Markdown". A red arrow points from the text "It is possible to add a new cell of code or a new markdown between any cells." to the "+ Code" button. The code cell contains the following Python code:

```
# Import the necessary libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from scipy import stats
import seaborn as sns

# Import the dataset
data = pd.read_csv('ESE4_ex1.csv')

# Inspect the dataset
data.head()
```

Below the code cell, there is a table of data:

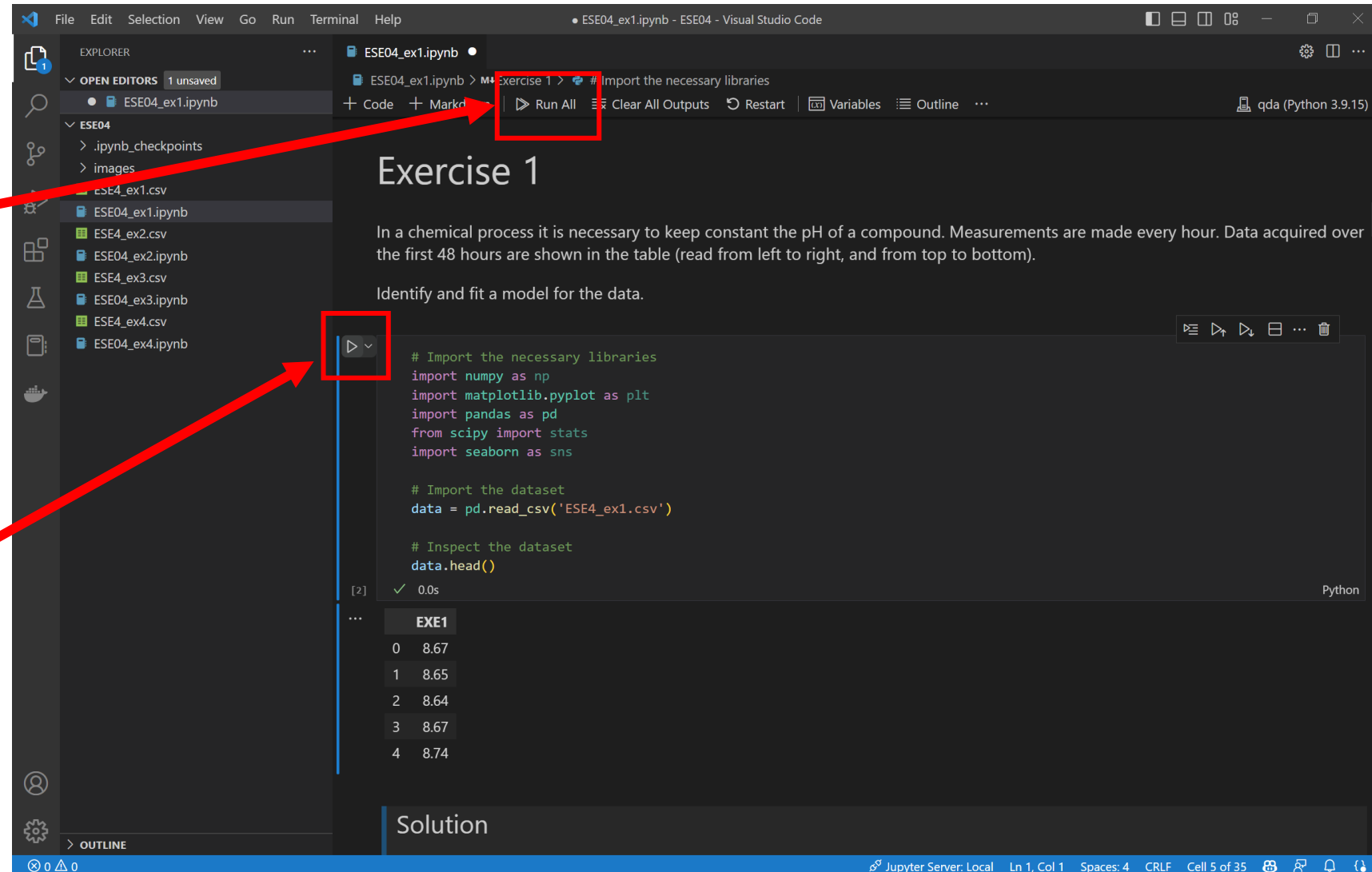
	EXE1
0	8.67
1	8.65
2	8.64
3	8.67
4	8.74

The bottom of the interface shows the "Solution" section and the status bar indicating "Jupyter Server: Local".

JUPYTER NOTEBOOK

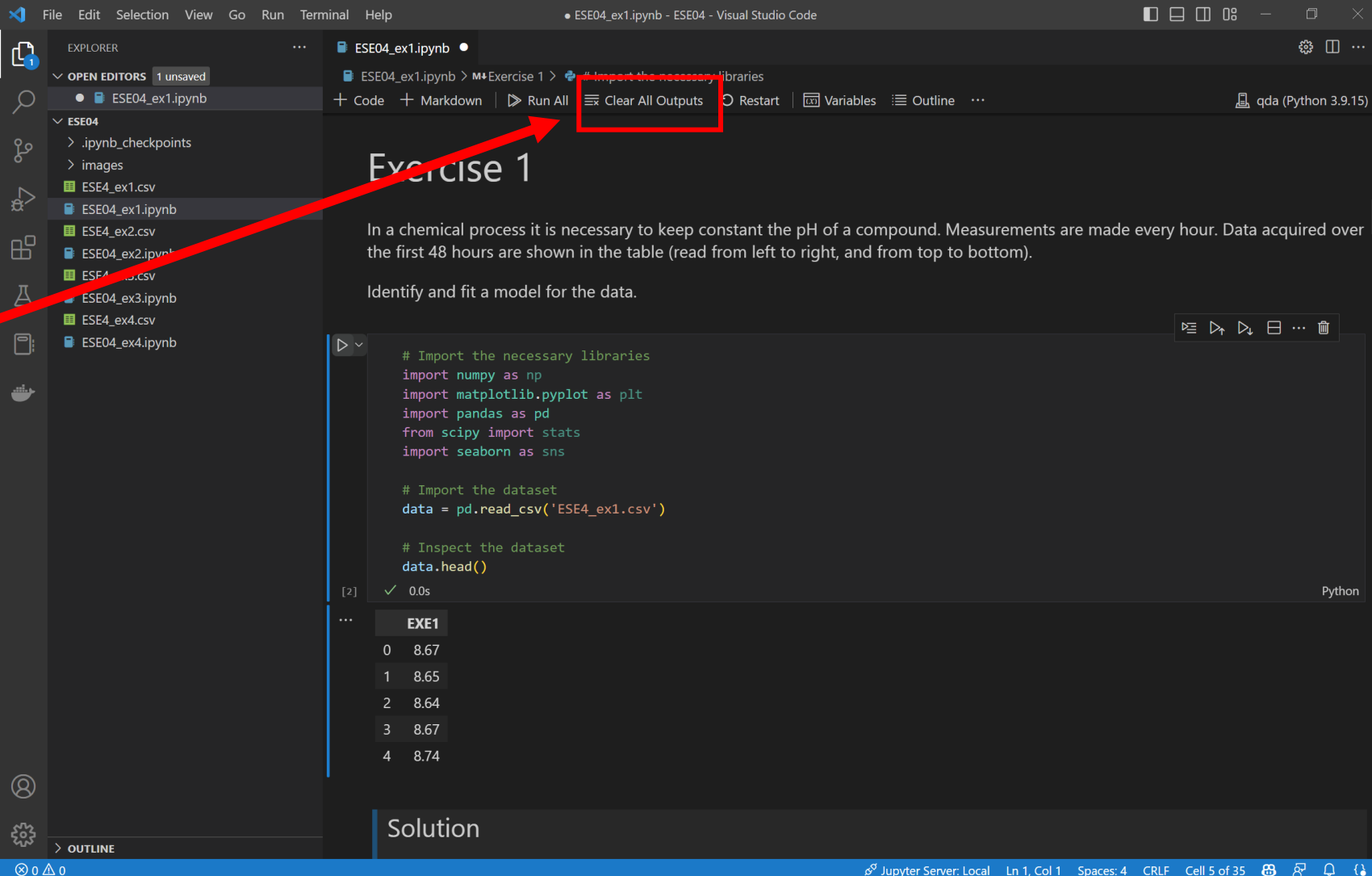
It is possible to run the entire code with the command *Run all*

Or either the individual cell of code can be run independently with the *Run cell* command.



JUPYTER NOTEBOOK

The command *Clear All Outputs* is useful to delete all plots and prints



The screenshot shows a Jupyter Notebook titled "Exercise 1" in a Visual Studio Code environment. The notebook contains a code cell with the following Python code:

```
# Import the necessary libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from scipy import stats
import seaborn as sns

# Import the dataset
data = pd.read_csv('ESE4_ex1.csv')

# Inspect the dataset
data.head()
```

The code cell has been executed, and the output is displayed below it. The output shows the first five rows of the dataset:

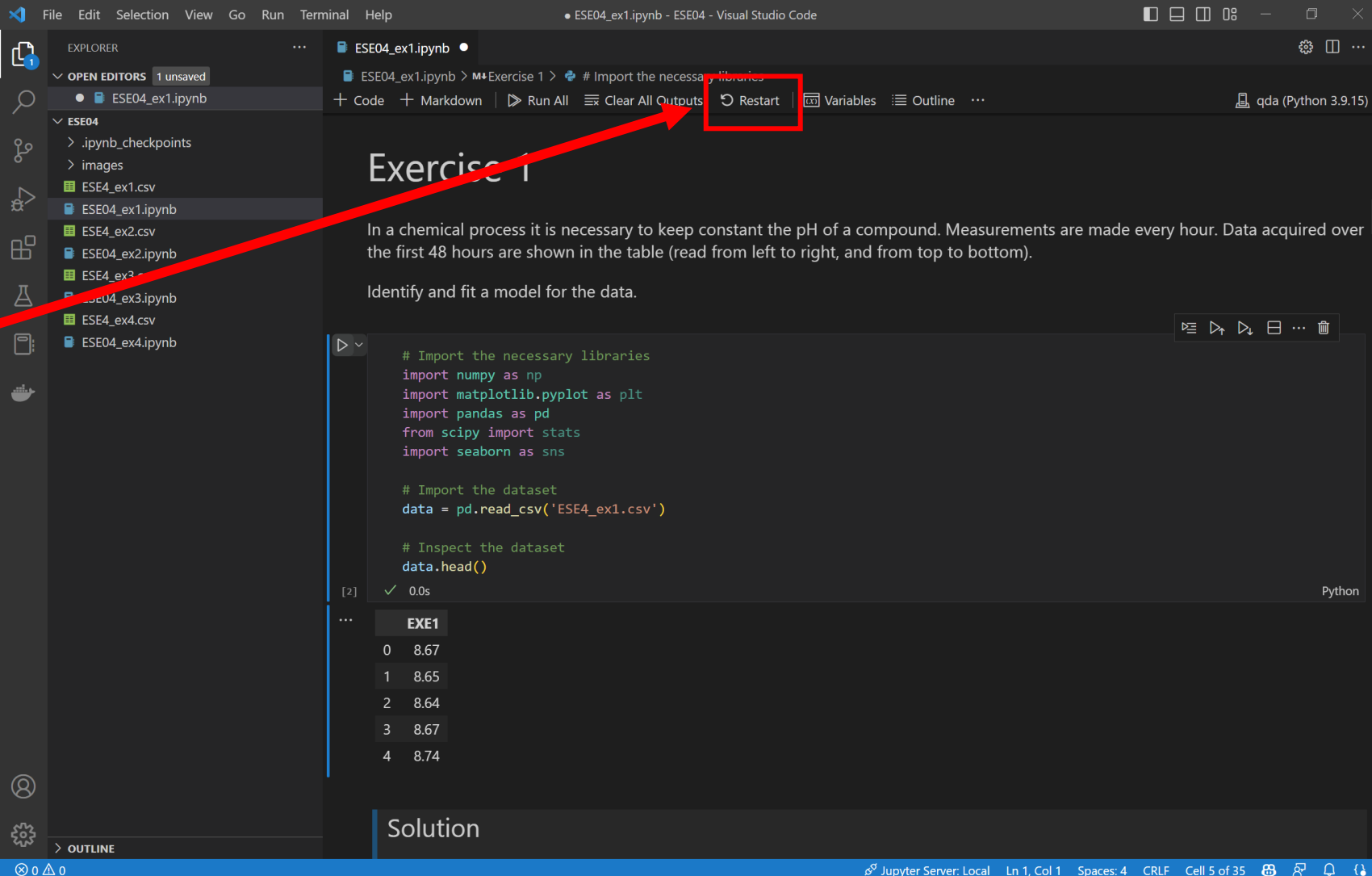
0	8.67
1	8.65
2	8.64
3	8.67
4	8.74

The "Clear All Outputs" button is highlighted with a red box, and a red arrow points to it from the text on the left. The button is located in the top right corner of the code cell's output area.

JUPYTER NOTEBOOK

The *Restart* command will restart the kernel, and clear all the variables.

This might be useful when incurring in errors caused by variables.



The screenshot shows a Jupyter Notebook titled "Exercise 1" in Visual Studio Code. The notebook content includes a title "Exercise 1", a description of a chemical process, and a task to identify and fit a model. The code cell contains the following Python code:

```
# Import the necessary libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from scipy import stats
import seaborn as sns

# Import the dataset
data = pd.read_csv('ESE4_ex1.csv')

# Inspect the dataset
data.head()
```

The output of the code cell shows a table of data:

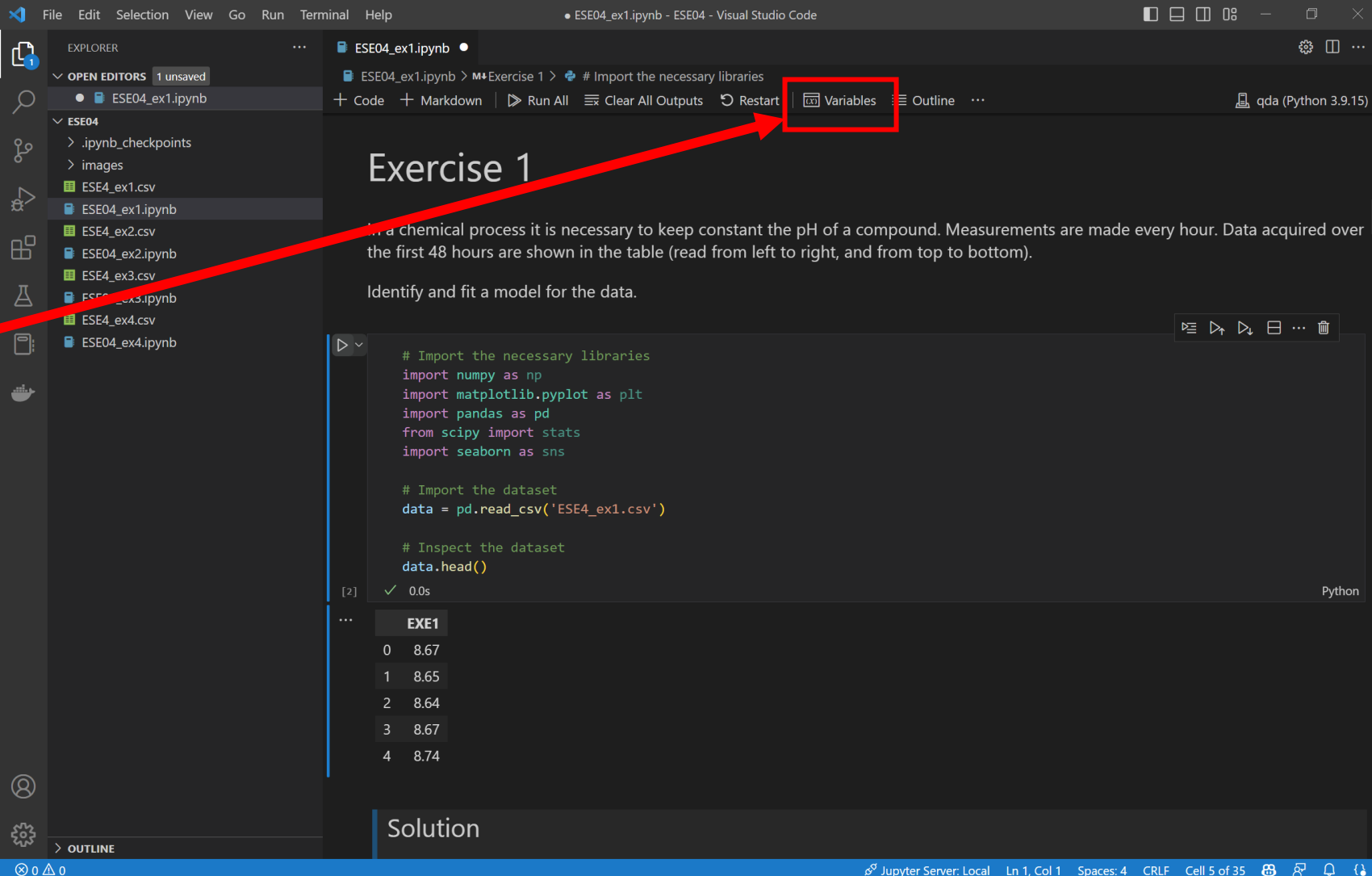
0	8.67
1	8.65
2	8.64
3	8.67
4	8.74

The "Restart" button is highlighted with a red box, and a red arrow points to it from the text on the left. The status bar at the bottom indicates "Jupyter Server: Local" and "Ln 1, Col 1".

JUPYTER NOTEBOOK

Variables can be displayed through the *Variables* command.

It is very useful for browsing and inspecting data sets or variables.



The screenshot shows the Jupyter Notebook interface in Visual Studio Code. The file explorer on the left lists files under the 'ESE04' directory, including 'ESE04_ex1.ipynb'. The main editor displays a notebook titled 'Exercise 1' with a code cell containing the following Python code:

```
# Import the necessary libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from scipy import stats
import seaborn as sns

# Import the dataset
data = pd.read_csv('ESE4_ex1.csv')

# Inspect the dataset
data.head()
```

The output of the code is displayed below the cell, showing the first five rows of the dataset:

0	8.67
1	8.65
2	8.64
3	8.67
4	8.74

A red arrow points from the text 'Variables can be displayed through the *Variables* command' to the 'Variables' button in the top right of the notebook interface. The 'Variables' button is highlighted with a red box.

JUPYTER NOTEBOOK

Pressing the *Variables* command, the list of variables will be displayed in a window.

The screenshot shows the Visual Studio Code interface with a Jupyter Notebook open. The Explorer sidebar on the left shows a file tree for 'ESE04' containing various CSV files and Jupyter notebooks. The main editor area displays the 'Exercise 1' notebook, which contains Python code for importing libraries and reading a dataset. A red box highlights the 'JUPYTER: VARIABLES' panel at the bottom, which lists the variables defined in the notebook.

Exercise 1

In a chemical process it is necessary to keep constant the pH of a compound. Measurements are made every hour. Data acquired over the first 48 hours are shown in the table (read from left to right, and from top to bottom).

Identify and fit a model for the data.

```
# Import the necessary libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from scipy import stats
import seaborn as sns

# Import the dataset
data = pd.read_csv('ESE4_ex1.csv')

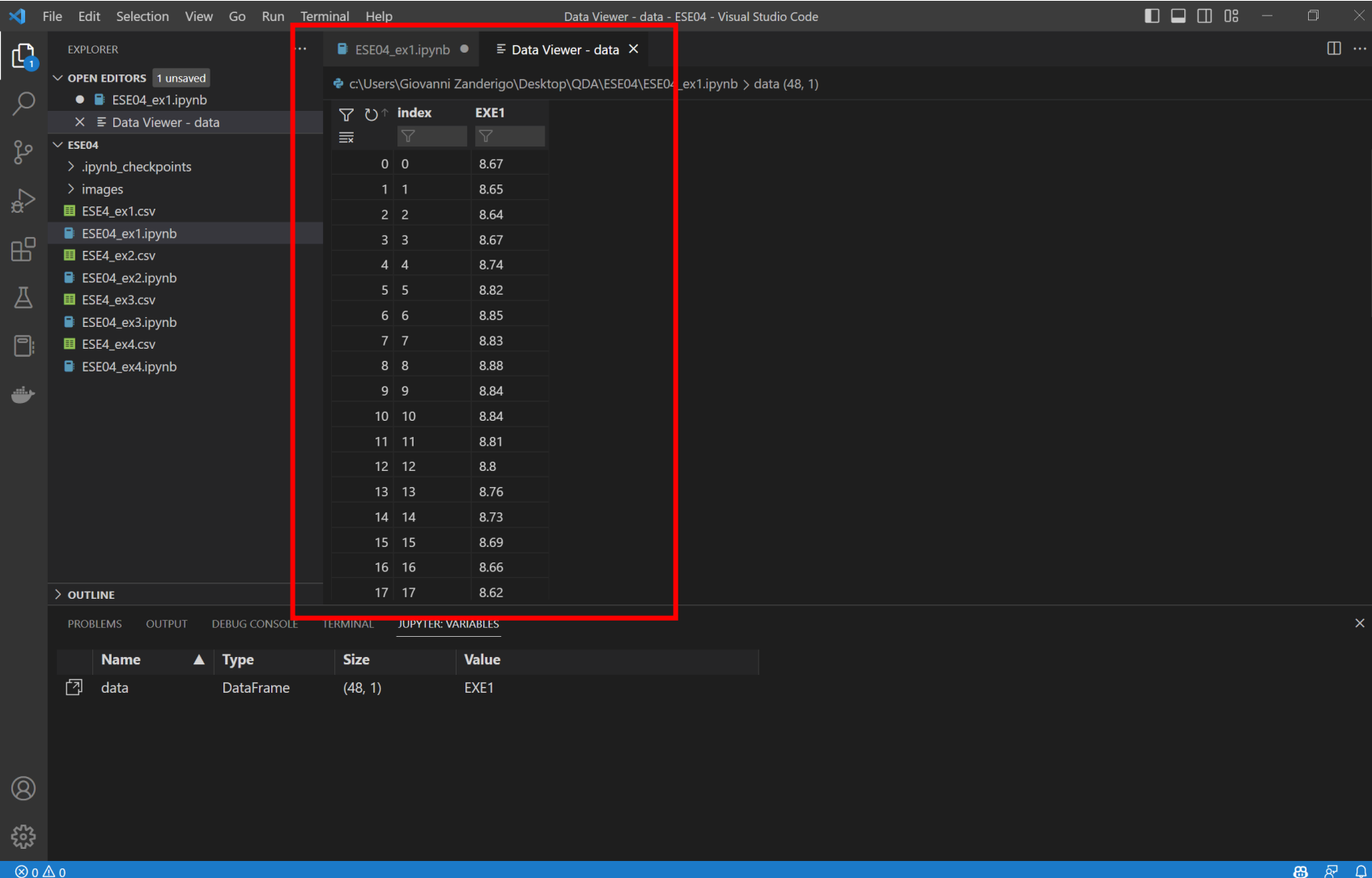
# Inspect the dataset
data.head()
```

JUPYTER: VARIABLES

Name	Type	Size	Value
data	DataFrame	(48, 1)	EXE1

JUPYTER NOTEBOOK

If we double-click on the variable's name, the *Data viewer* tab will open, displaying the content of the variable.



The screenshot shows the Jupyter Notebook interface in Visual Studio Code. The 'Data Viewer' tab is open, displaying a table with columns 'index' and 'EXE1'. The table contains 18 rows of data. A red box highlights the 'Data Viewer' tab and the table content. The 'JUPYTER VARIABLES' panel at the bottom shows the variable 'data' as a DataFrame with shape (48, 1) and value EXE1.

index	EXE1
0	8.67
1	8.65
2	8.64
3	8.67
4	8.74
5	8.82
6	8.85
7	8.83
8	8.88
9	8.84
10	8.84
11	8.81
12	8.8
13	8.76
14	8.73
15	8.69
16	8.66
17	8.62

Name	Type	Size	Value
data	DataFrame	(48, 1)	EXE1

TUTORIALS



Any additional information regarding Visual Studio Code can be found in the following link:

VS CODE TUTORIAL LINK:

<https://code.visualstudio.com/docs/introvideos/basics>

POLITECNICO DI MILANO

B 23



DIPARTIMENTO DI ECCELLENZA
MIUR 2018-2022

CONTACTS

Prof.

Matteo Bugatti

matteo.bugatti@polimi.it

www.mecc.polimi.it



@meccpolimi

POLITECNICO
MILANO 1863