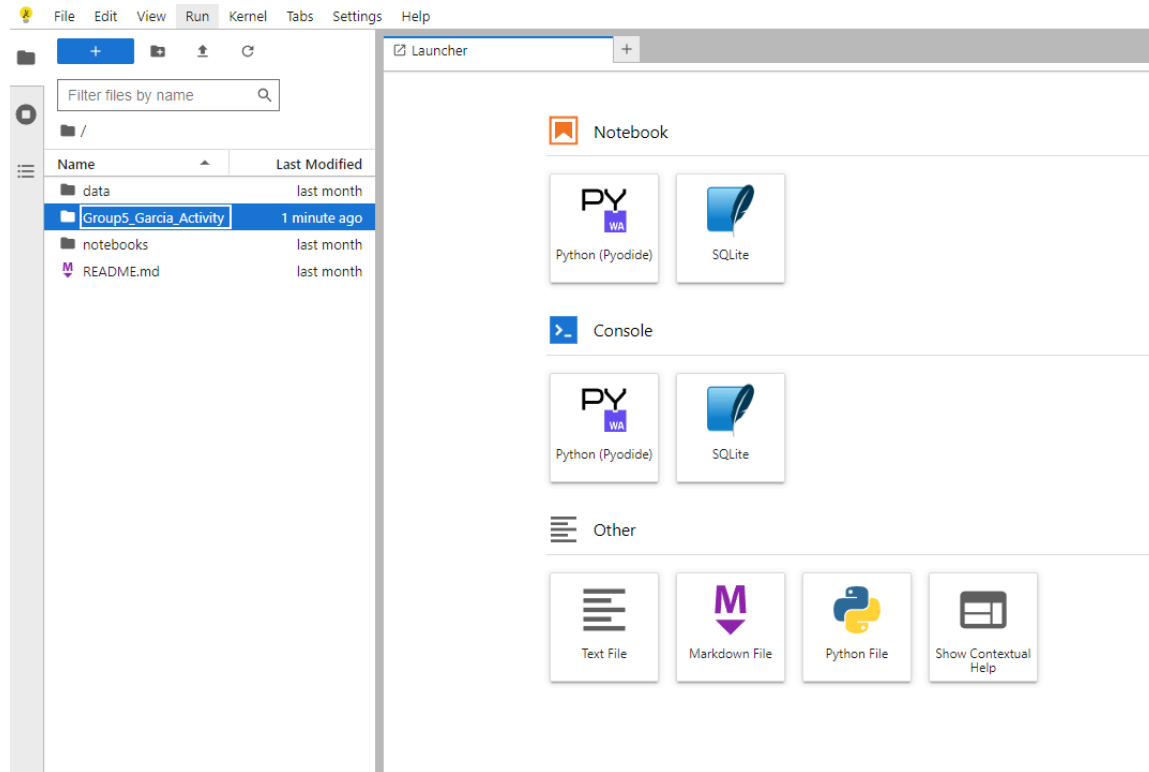
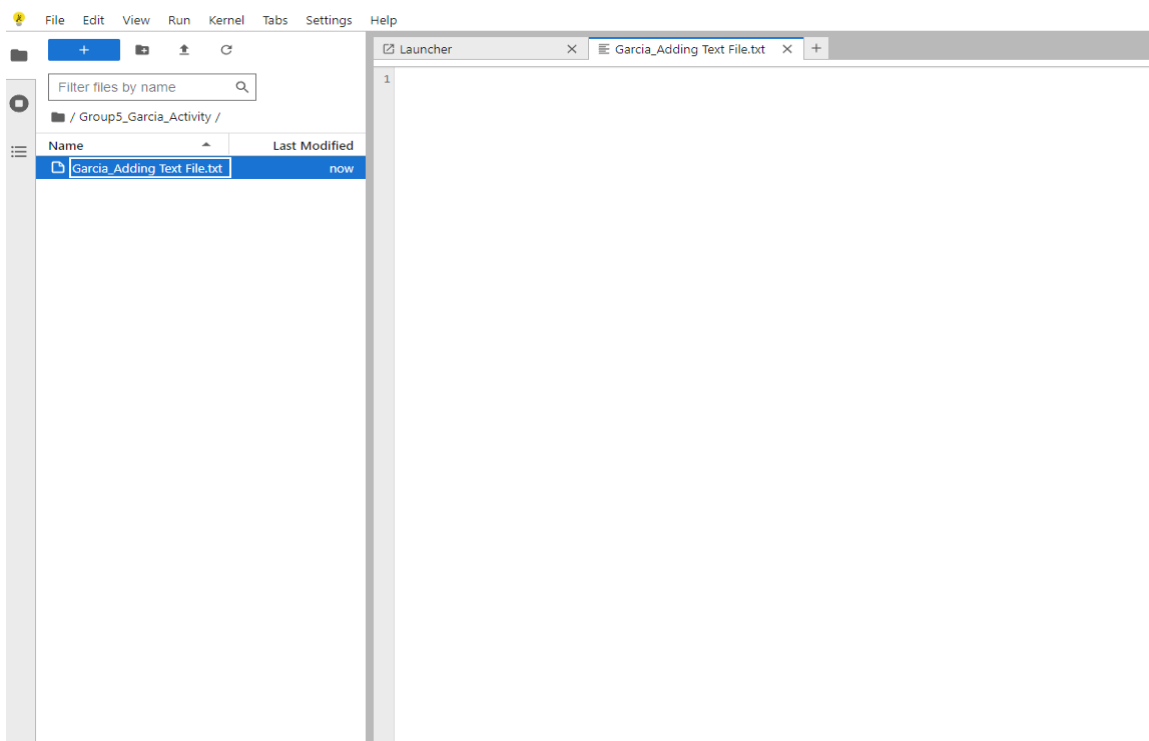


CS3C

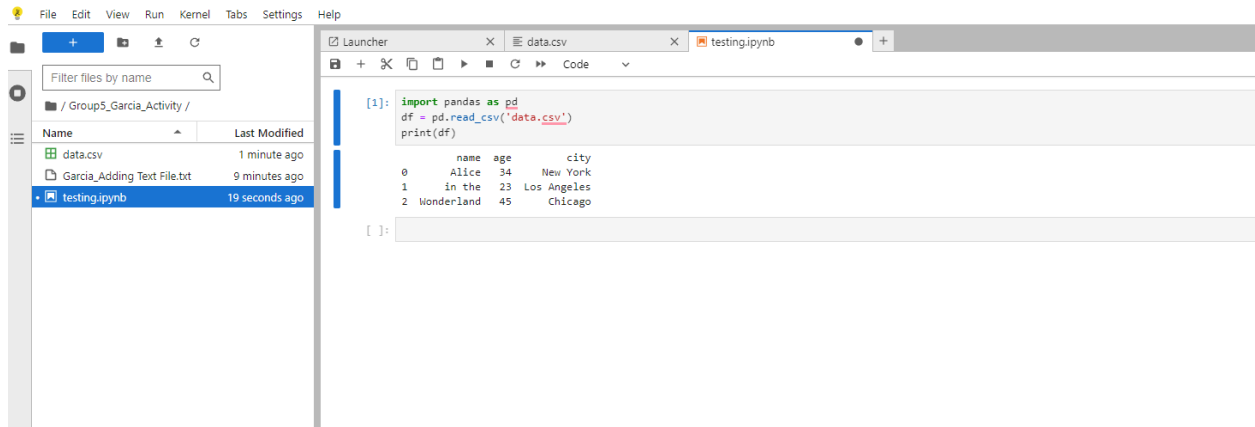
- **ADDING FOLDER**



- **ADDING TEXT FILE**



- **CSV FILE FOR DATA ANALYSIS AND VISULIZATION**



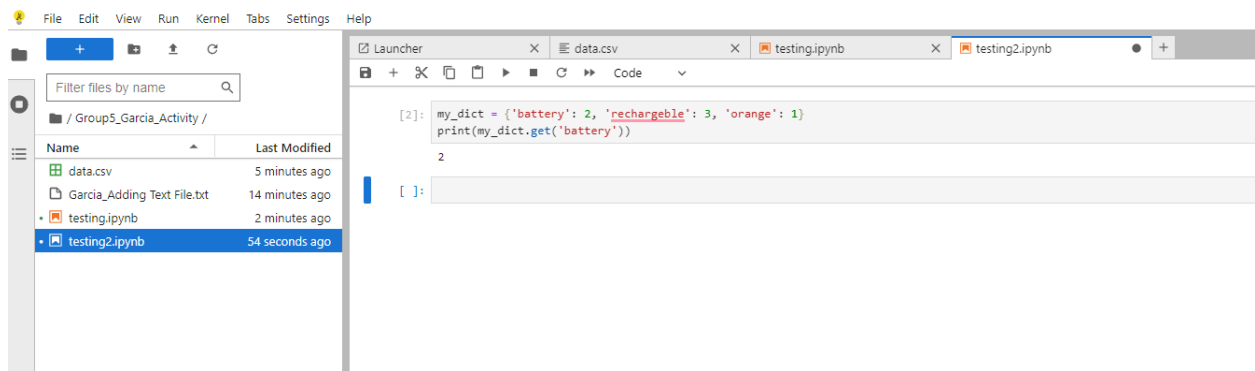
The screenshot shows a Jupyter Notebook interface with a file browser on the left and a code editor on the right. The file browser shows a directory named 'Group5_Garcia_Activity' containing 'data.csv', 'Garcia_Adding Text File.txt', and 'testing.ipynb'. The code editor has two tabs: 'data.csv' and 'testing.ipynb'. The 'testing.ipynb' tab is active, showing a code cell with the following code:

```
[1]: import pandas as pd
df = pd.read_csv('data.csv')
print(df)
```

The output of the code cell is a DataFrame with 3 rows and 4 columns: 'name', 'age', and 'city'. The data is as follows:

	name	age	city
0	Alice	34	New York
1	in the	23	Los Angeles
2	Wonderland	45	Chicago

- **TO WRITE AND CALL DICTIONARY METHODS.**

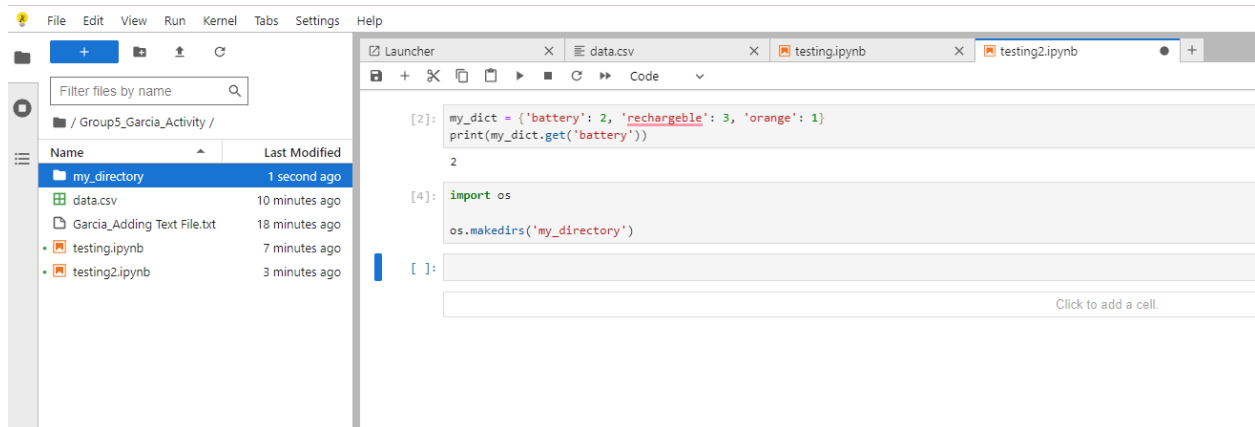


The screenshot shows a Jupyter Notebook interface with a file browser on the left and a code editor on the right. The file browser shows a directory named 'Group5_Garcia_Activity' containing 'data.csv', 'Garcia_Adding Text File.txt', 'testing.ipynb', and 'testing2.ipynb'. The code editor has three tabs: 'data.csv', 'testing.ipynb', and 'testing2.ipynb'. The 'testing2.ipynb' tab is active, showing a code cell with the following code:

```
[2]: my_dict = {'battery': 2, 'rechargeable': 3, 'orange': 1}
print(my_dict.get('battery'))
```

The output of the code cell is the value 2.

- **TO CREATE A DIRECTORY USING JUPYTER NOTEBOOK.**



The screenshot shows a Jupyter Notebook interface with a file browser on the left and a code editor on the right. The file browser shows a directory named 'Group5_Garcia_Activity' containing 'my_directory', 'data.csv', 'Garcia_Adding Text File.txt', 'testing.ipynb', and 'testing2.ipynb'. The code editor has three tabs: 'data.csv', 'testing.ipynb', and 'testing2.ipynb'. The 'testing2.ipynb' tab is active, showing two code cells. The first code cell contains the following code:

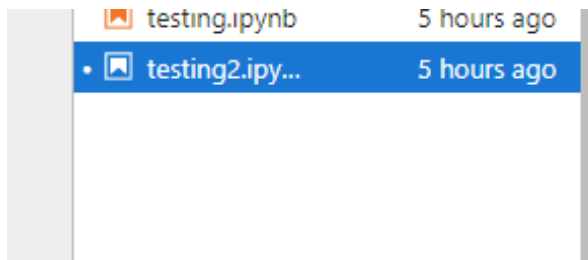
```
[2]: my_dict = {'battery': 2, 'rechargeable': 3, 'orange': 1}
print(my_dict.get('battery'))
```

The output of the first code cell is the value 2. The second code cell contains the following code:

```
[4]: import os
os.makedirs('my_directory')
```

The output of the second code cell is an empty box with the text 'Click to add a cell.'

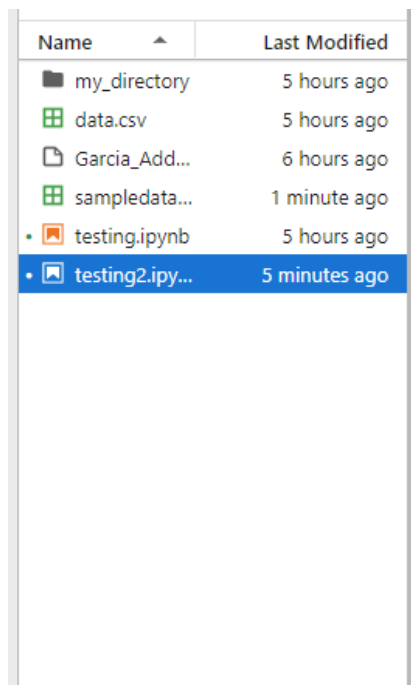
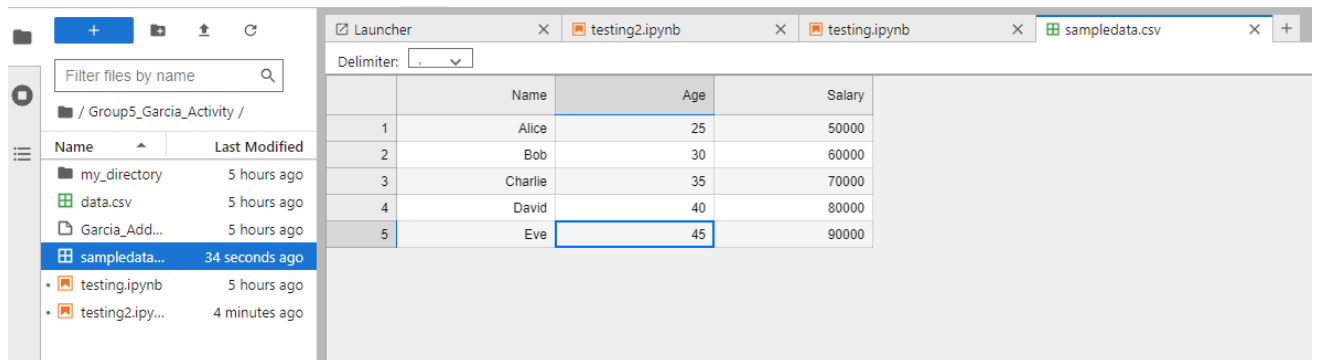
- TO IMPORT LIBRARIES



```
[2]: import pandas as pd
import numpy as np
my_array = np.array([1,2,3,4,5])
print(my_array)

[1 2 3 4 5]
```

- TO USE CSV FILE FOR DATA



```
print(my_dict.get('battery'))

2

[4]: import os

os.makedirs('my_directory')

[2]: import pandas as pd
import numpy as np
my_array = np.array([1,2,3,4,5])
print(my_array)

[1 2 3 4 5]

[3]: import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('sampledata.csv')
print(df)

   Name  Age  Salary
0  Alice   25   50000
1   Bob   30   60000
2 Charlie   35   70000
3  David   40   80000
4   Eve   45   90000
```

- ANALYSIS AND VISUALIZATION

```
[4]: print("First 5 rows of the DataFrame:")  
      print(df.head())
```

First 5 rows of the DataFrame:

	Name	Age	Salary
0	Alice	25	50000
1	Bob	30	60000
2	Charlie	35	70000
3	David	40	80000
4	Eve	45	90000

```
[5]: print("\nMissing Values:")  
      print(df.isnull().sum())
```

Missing Values:

Name	0
Age	0
Salary	0

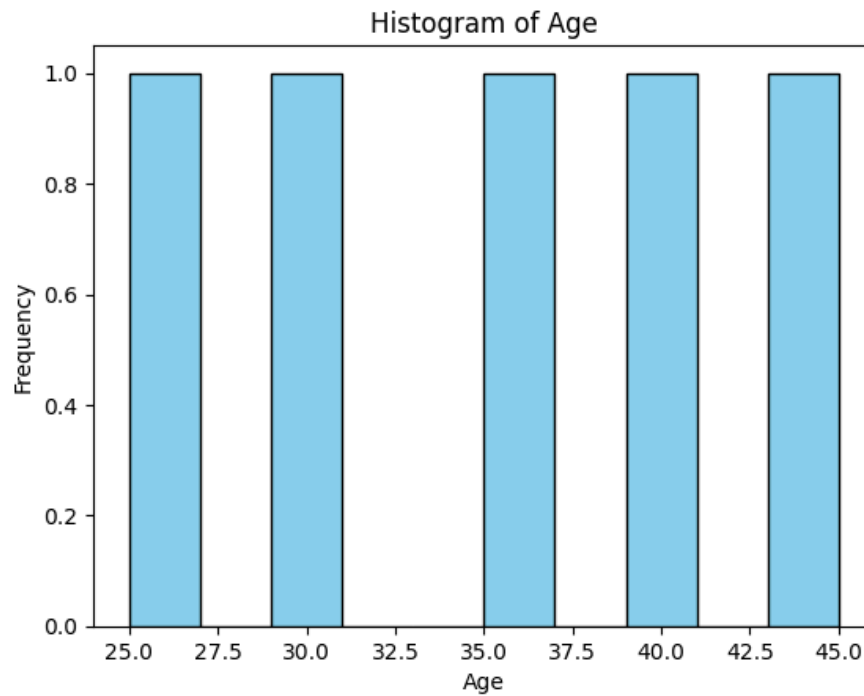
dtype: int64

```
[6]: print("\nSummary Statistics:")  
      print(df.describe())
```

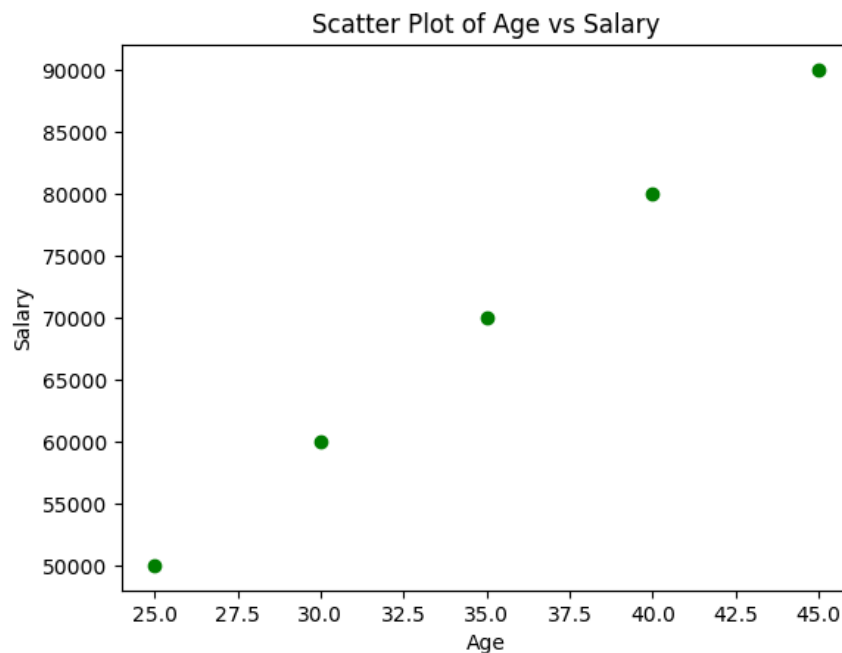
Summary Statistics:

	Age	Salary
count	5.000000	5.000000
mean	35.000000	70000.000000
std	7.905694	15811.388301
min	25.000000	50000.000000
25%	30.000000	60000.000000
50%	35.000000	70000.000000
75%	40.000000	80000.000000
max	45.000000	90000.000000

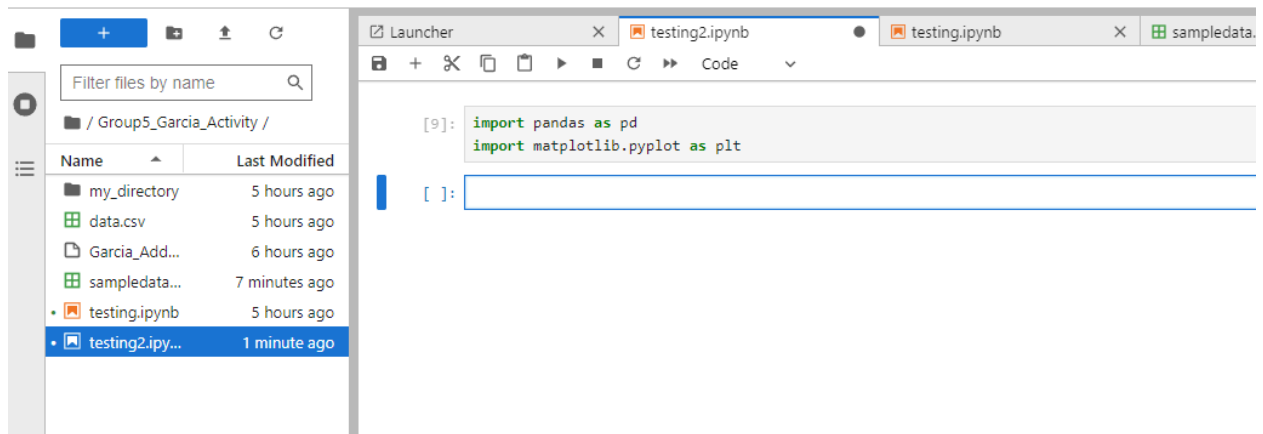
```
[7]: plt.hist(df['Age'], bins=10, color='skyblue', edgecolor='black')
plt.title('Histogram of Age')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```



```
[8]: plt.scatter(df['Age'], df['Salary'], color='green')
plt.title('Scatter Plot of Age vs Salary')
plt.xlabel('Age')
plt.ylabel('Salary')
plt.show()
```



• IMPORT LIBRARIES



The screenshot shows the JupyterLab interface. On the left is the file explorer for the directory `/ Group5_Garcia_Activity /`. It contains a table with the following files and their last modified times:

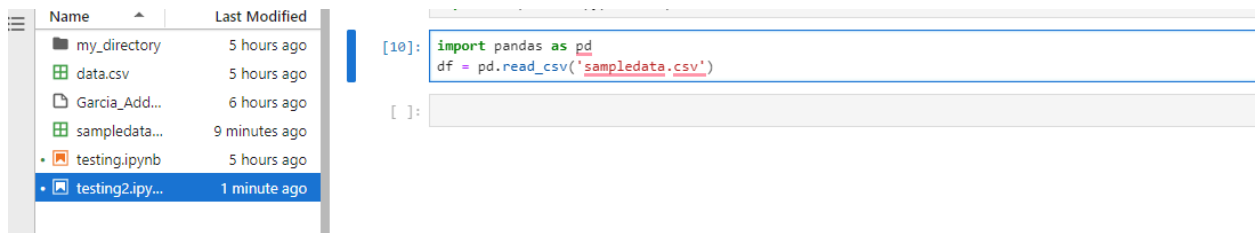
Name	Last Modified
my_directory	5 hours ago
data.csv	5 hours ago
Garcia_Add...	6 hours ago
sampledata...	7 minutes ago
testing.ipynb	5 hours ago
testing2.ipynb	1 minute ago

The main area shows the code editor for `testing2.ipynb`. The code in cell [9] is:

```
[9]: import pandas as pd
import matplotlib.pyplot as plt
```

Below the code is an empty input box for the next cell.

• FINDING DATA



The screenshot shows the JupyterLab interface. On the left is the file explorer for the directory `/ Group5_Garcia_Activity /`. It contains a table with the following files and their last modified times:

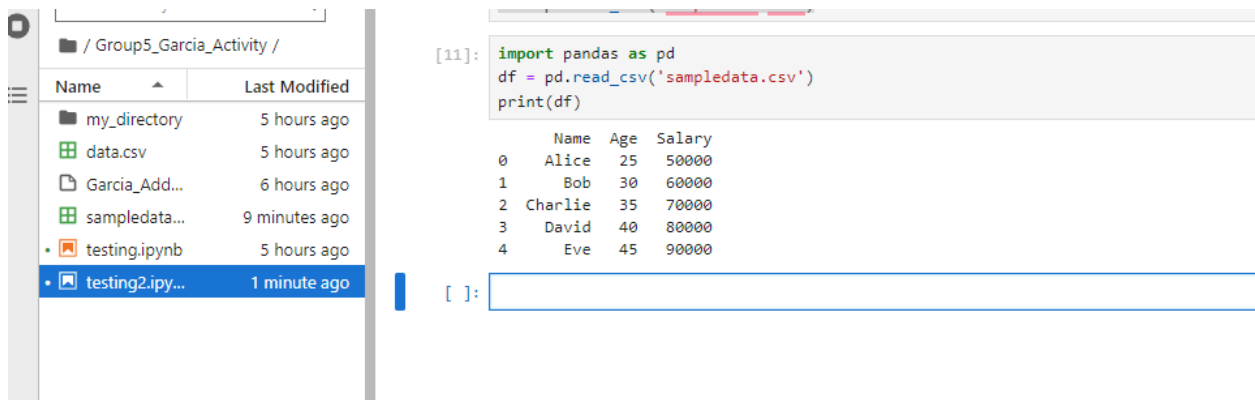
Name	Last Modified
my_directory	5 hours ago
data.csv	5 hours ago
Garcia_Add...	6 hours ago
sampledata...	9 minutes ago
testing.ipynb	5 hours ago
testing2.ipynb	1 minute ago

The main area shows the code editor for `testing2.ipynb`. The code in cell [10] is:

```
[10]: import pandas as pd
df = pd.read_csv('sampledata.csv')
```

Below the code is an empty input box for the next cell.

• IMPORTING DATA



The screenshot shows the JupyterLab interface. On the left is the file explorer for the directory `/ Group5_Garcia_Activity /`. It contains a table with the following files and their last modified times:

Name	Last Modified
my_directory	5 hours ago
data.csv	5 hours ago
Garcia_Add...	6 hours ago
sampledata...	9 minutes ago
testing.ipynb	5 hours ago
testing2.ipynb	1 minute ago

The main area shows the code editor for `testing2.ipynb`. The code in cell [11] is:

```
[11]: import pandas as pd
df = pd.read_csv('sampledata.csv')
print(df)
```

The output of the code is a table with 5 rows and 4 columns:

	Name	Age	Salary
0	Alice	25	50000
1	Bob	30	60000
2	Charlie	35	70000
3	David	40	80000
4	Eve	45	90000

Below the output is an empty input box for the next cell.

- DATA ATTRIBUTES

```
[12]: print("Data Types:")  
      print(df.dtypes)
```

```
Data Types:  
Name      object  
Age       int64  
Salary    int64  
dtype: object
```

```
[13]: print("\nColumn Names:")  
      print(df.columns)
```

```
Column Names:  
Index(['Name', 'Age', 'Salary'], dtype='object')
```

```
[14]: print("\nMissing Values:")  
      print(df.isnull().sum())
```

```
Missing Values:  
Name      0  
Age       0  
Salary    0  
dtype: int64
```

```
[ ]:
```