**Data Mining and Data Warehousing Lab**

CSEL-4108

|  |
| --- |
| Assignment on Data Analysis Techniques Using Pandas — 2 |

|  |  |
| --- | --- |
| Submitted By | |
|  |  |
| **NISHAT MAHMUD**  ID: B190305003 | **MD. WALIUL ISLAM RAYHAN**  ID: B190305034 |

Submitted To

**MD. MANOWARUL ISLAM, PHD**

Associate Professor

Dept. of CSE

Jagannath University, Dhaka - 1100

Department of Computer Science and Engineering, Jagannath University, Dhaka

This assignment explores various data analysis techniques using the Pandas library in Python. Below are detailed explanations, code examples, and expected outputs for different operations.

1. **Finding Maximum Values**

**Method: df.max()**

This method finds the maximum value for each column in a DataFrame.

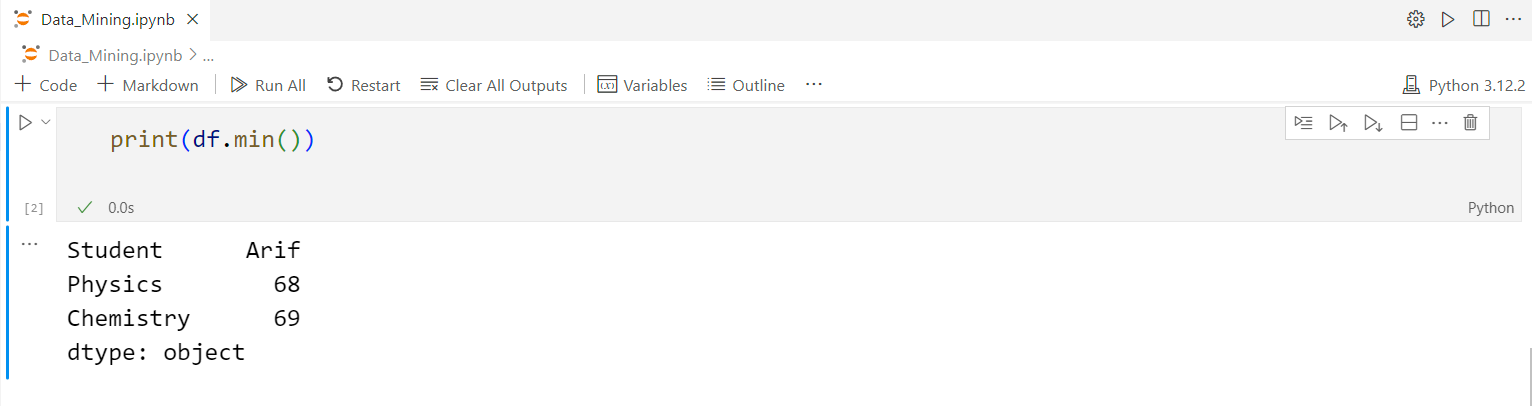
A screenshot of a computer

Description automatically generated

**2. Finding Minimum Values**

**Method: df.min()**

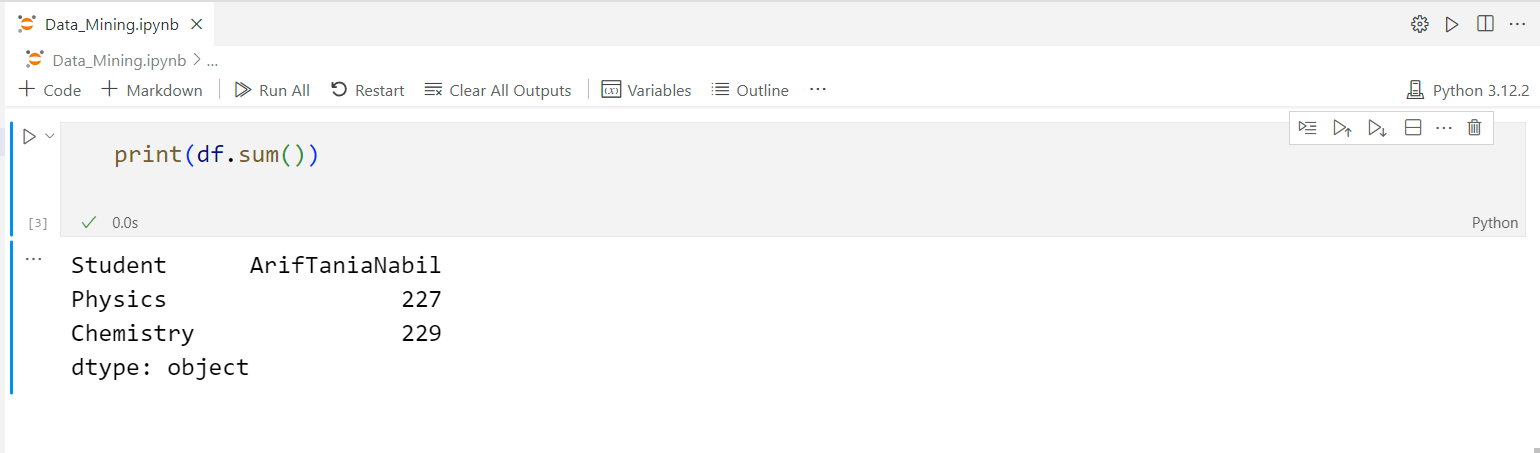
This method finds the minimum value for each column in a DataFrame.



**3. Summing Values**

**Method: df.sum()**

This method adds up the values for each column in a DataFrame. For non-numeric columns, it concatenates the values.



**4. Counting Values**

**Method: df.count()**

This method counts the number of non-null values in each column.

A screenshot of a computer

Description automatically generated

**5. Calculating Mean**

**Method: df.mean()**

This method calculates the mean (average) of numeric values for each column.

A screenshot of a computer

Description automatically generated

**6. Calculating Median**

**Method: df.median()**

This method finds the median value for each column in a DataFrame.

A screenshot of a computer

Description automatically generated

**7. Calculating Mode**

**Method: df.mode()**

This method finds the mode (most frequent value) for each column.

A screenshot of a computer

Description automatically generated

**8. Calculating Quartiles**

**Method: df.quantile([0.25, 0.5, 0.75])**

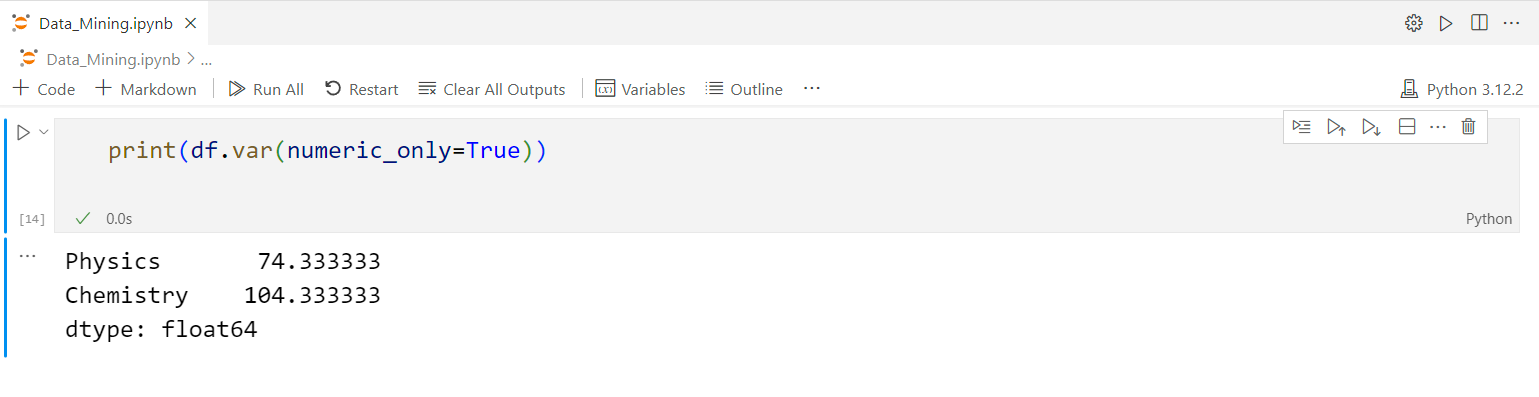
This method calculates the quartiles of numeric values in each column.



**9. Calculating Variance**

**Method: df.var()**

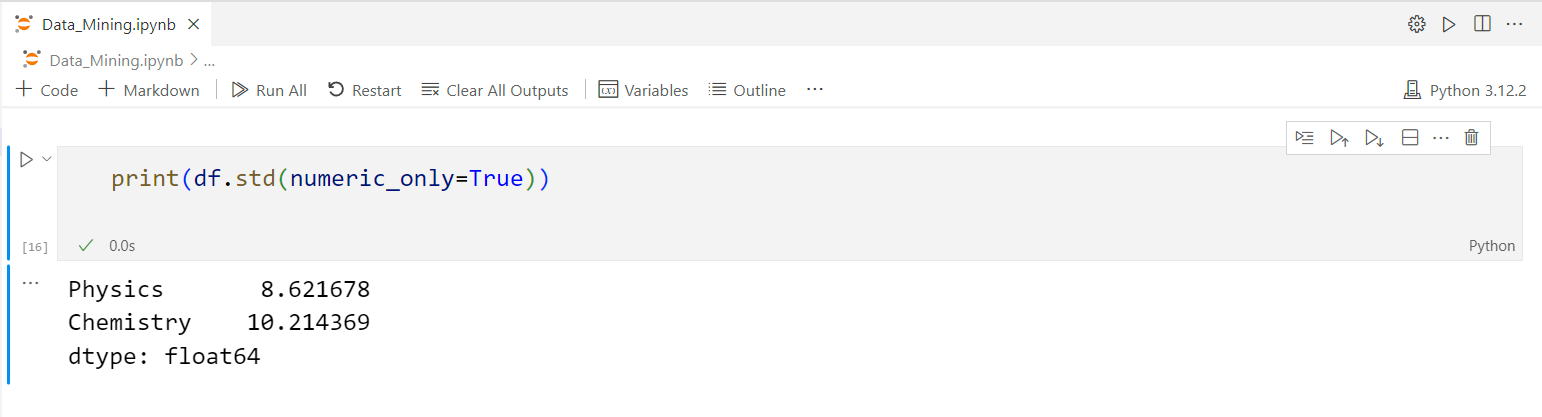
This method calculates the variance of numeric values in each column.



**10. Calculating Standard Deviation**

**Method: df.std()**

This method calculates the standard deviation of numeric values in each column.



**11. Performing Aggregation**

**Method: df.aggregate(['max', 'min', 'sum'])**

This method applies multiple aggregation functions to the columns.

A screenshot of a computer

Description automatically generated

**12. Grouping and Aggregation**

**Method: df.groupby('Student').aggregate(['mean', 'sum'])**

This method groups the DataFrame by a column and applies aggregation functions.

A screenshot of a computer code

Description automatically generated

**13. Sorting Values**

**Method: df.sort\_values(by='Score', ascending=False)**

This method sorts the DataFrame by a specified column.

A screenshot of a computer

Description automatically generated

**14. Handling Missing Values**

**Method: df.fillna(0)**

This method replaces missing values in the DataFrame with a specified value.

A screenshot of a computer

Description automatically generated