Useful methods and operations

unique(),nunique(),value_counts(), sort_values(), apply(), index. column

Concept	Description	Example
unique()	Finds and returns all the unique values in a column.	df['col_2'].unique()Output: array([111, 222, 333, 555])
nunique()	Returns the number of unique values in a column, counting NaN as a missing value.	df['col_2'].nunique()Output: 4
value_counts()	Returns a count of all unique values in a column along with their frequency. Does not count NaN values.	df['col_2'].value_counts()Output: 111 2222 1333 1555 1dtype: int64
count()	Counts the number of non-NaN values in a column.	df['col_2'].count()Output: 5
sort_values()	Sorts the DataFrame by the values of a column. By default, it sorts in ascending order. You can set ascending=False for descending order.	df.sort_values('col_2', ascending=True)Output: DataFrame sorted by col_2 in ascending order. df.sort_values('col_2', ascending=False)Descending order.
apply()	Applies a custom function to a column. It allows broadcasting custom operations to data.	df['col'].apply(lambda x: x * x)Output: 1 4 9 16 25dtype: int64
index	Returns the index (row labels) of the DataFrame.	df.indexOutput: RangeIndex(start=0, stop=5, step=1)
columns	Returns the column names of the DataFrame.	df.columnsOutput: Index(['col', 'col_2', 'col_3'], dtype='object')
Copy DataFrame	Creates a copy of the DataFrame.	df2 = df.copy()Output: DataFrame with the same content as df.
Delete Column	Deletes a column from the DataFrame.	del aa['col_3']Output: DataFrame without the col_3 column.