

Python Programming

Unit 06 – Lecture 04: Missing Data, Aggregation, Combining, CSV I/O

Tofik Ali

School of Computer Science, UPES Dehradun

February 14, 2026

Repository: <https://github.com/tali7c/Python-Programming>

Quick Links

Core Concepts

Demo

Interactive

Summary

Agenda

1 Core Concepts

2 Demo

3 Interactive

4 Summary

Learning Outcomes

- Detect and handle missing values (NaN)

Learning Outcomes

- Detect and handle missing values (NaN)
- Use aggregation (groupby) for summaries

Learning Outcomes

- Detect and handle missing values (NaN)
- Use aggregation (groupby) for summaries
- Combine DataFrames (concat, merge)

Learning Outcomes

- Detect and handle missing values (NaN)
- Use aggregation (groupby) for summaries
- Combine DataFrames (concat, merge)
- Import/export CSV using `read_csv` and `to_csv`

Missing Data

- Missing values appear as NaN in Pandas

Missing Data

- Missing values appear as NaN in Pandas
- Useful functions:

Missing Data

- Missing values appear as NaN in Pandas
- Useful functions:
 - `isna()`, `notna()`

Missing Data

- Missing values appear as NaN in Pandas
- Useful functions:
 - `isna()`, `notna()`
 - `dropna()`

Missing Data

- Missing values appear as NaN in Pandas
- Useful functions:
 - `isna()`, `notna()`
 - `dropna()`
 - `fillna(value)`

Aggregation with groupby

```
avg_by_city = df.groupby("city")["marks"].mean()
```

- Useful for summaries (average marks per city, counts per category)

Combining DataFrames

- `pd.concat(...)`: stack rows or columns

Combining DataFrames

- `pd.concat(...)`: stack rows or columns
- `pd.merge(a, b, on="key")`: SQL-like join

CSV Import/Export

```
df = pd.read_csv("data/student_scores.csv")  
df.to_csv("data/cleaned.csv", index=False)
```


Demo: Missing Data + Groupby + CSV

- Data: `data/student_scores.csv`

Demo: Missing Data + Groupby + CSV

- Data: `data/student_scores.csv`
- Script: `demo/pandas_missing_data_demo.py`

Demo: Missing Data + Groupby + CSV

- Data: `data/student_scores.csv`
- Script: `demo/pandas_missing_data_demo.py`
- Shows:

Demo: Missing Data + Groupby + CSV

- Data: `data/student_scores.csv`
- Script: `demo/pandas_missing_data_demo.py`
- Shows:
 - reading CSV

Demo: Missing Data + Groupby + CSV

- Data: `data/student_scores.csv`
- Script: `demo/pandas_missing_data_demo.py`
- Shows:
 - reading CSV
 - missing value handling

Demo: Missing Data + Groupby + CSV

- Data: `data/student_scores.csv`
- Script: `demo/pandas_missing_data_demo.py`
- Shows:
 - reading CSV
 - missing value handling
 - groupby aggregation

Demo: Missing Data + Groupby + CSV

- Data: `data/student_scores.csv`
- Script: `demo/pandas_missing_data_demo.py`
- Shows:
 - reading CSV
 - missing value handling
 - groupby aggregation
 - merge with another DataFrame

Checkpoint 1

Question: What does `fillna(0)` do?

Checkpoint 2

Question: When would you use `merge` instead of `concat`?

Think-Pair-Share

Discuss:

- Is it always correct to replace missing marks with 0?
Why/why not?

Key Takeaways

- Missing values can be detected and filled/dropped

Key Takeaways

- Missing values can be detected and filled/dropped
- `groupby` provides powerful summaries

Key Takeaways

- Missing values can be detected and filled/dropped
- `groupby` provides powerful summaries
- `merge` joins tables using keys; `concat` stacks data

Key Takeaways

- Missing values can be detected and filled/dropped
- `groupby` provides powerful summaries
- `merge` joins tables using keys; `concat` stacks data
- CSV I/O connects Pandas with real-world datasets

Exit Question

Write one line to read a CSV file named "marks.csv" into a DataFrame.