# Week 2: Student Data Management with Pandas

Data Science for Mathematics Teachers

November 11, 2025

## Course Information

**Course:** Data Science for Mathematics Teachers
**Series:** Professional Development Series
**Duration:** 8 weeks
**Level:** Beginner To Intermediate
**Target:** Mathematics Teachers

## 1 Week 2: Student Data Management with Pandas

### 1.1 Learning Objectives

By the end of this week, you will be able to:

- Import and organize student data using Pandas
- Calculate grades, averages, and statistics
- Create automated gradebooks and reports
- Handle missing data in student records

### 1.2 Topics Covered This Week

- Introduction to Pandas for educational data
- Managing student grades and attendance
- Creating gradebooks and reports

### 1.3 Key Concepts You Will Work With

- DataFrame creation with pd.{DataFrame()}
- Reading CSV files with pd.{read_csv()}
- Data selection with .loc[] and .iloc[]
- Calculating means, medians, and standard deviations
- Group operations with .{groupby()}
- Handling NaN values with .{fillna()} and .{dropna()}
- Creating pivot tables with .{pivot_table()}
- Exporting data with .{to_csv()} and .{to_excel()}

## 1.4 Practical Exercises

**Difficulty Level:** Beginner
**Total Exercises:** 3

### Exercise 1: AI-Enhanced Challenge: Introduction to Pandas for educational data

**Difficulty:** Beginner
**AI-Enhanced Programming Exercise**
**Task:** Using Python, develop a Python solution focusing on dataframe creation with pd.{dataframe()}.
**Step-by-Step Instructions:**

1. Focus on implementing dataframe creation with pd.{dataframe()} effectively
2. Create clear, educational examples for student understanding
3. Test your implementation with classroom scenarios
4. Add comprehensive comments for teaching purposes
5. Validate results and create sample outputs

**Technical Requirements:**

- Use DataFrame creation with pd.{DataFrame()} in your implementation
- Use Reading CSV files with pd.{read_csv()} in your implementation
- Use Data selection with .loc[] and .iloc[] in your implementation

**Expected Output:** A working Python script that mathematics teachers can run in their classroom to solve real educational problems.
**Assessment:** Your solution should be practical, well-commented, and directly applicable to teaching mathematics.
**Teaching Context:** Practical student data management

### Exercise 2: Concept-Based Challenge: Managing student grades and attendance

**Difficulty:** Beginner
**Task:** Create a data management system for classroom information.
**Data Handling:**

- Use Data selection with .loc[] and .iloc[] to import student records
- Apply Calculating means, medians, and standard deviations to structure information
- Generate reports for parents and administrators
- Ensure data privacy and security

**Deliverable:** A complete data management solution for educators.
**Teaching Context:** Practical student data management

### Exercise 3: Concept-Based Challenge: Creating gradebooks and reports

**Difficulty:** Beginner
**Task:** Develop beginner-friendly math programs for classroom demonstrations.
**Program Requirements:**

1. Use Group operations with .{groupby()} to create clear, simple code
2. Implement Handling NaN values with .{fillna()} and .{dropna()} students can understand
3. Add comments explaining each step
4. Include sample problems and solutions
5. Test with actual classroom scenarios

**Output:** Educational programs that make math concepts tangible.
**Teaching Context:** Practical student data management

**Generated by:** AI-Powered Curriculum Generator
*Professional Curriculum Generation System*
2025-11-11 20:31:18