Noah Lambe

QAP1 - SDAT and DevOps

27 May 2025

Student Assignment Tracker Documentation

Clean Code Practices

This project demonstrates clean code practices in several ways:

a. Dynamic Status Calculation:
Rather than storing a static 'status' field, the Assignment class uses a method that dynamically returns the current status based on the due date and completion flag. This reduces data redundancy.

b. Single Responsibility Principle:
Each class in the program (Assignment, Course, AssignmentManager) has a clear, singular role.

c. Descriptive Method Naming:
Methods such as addAssignment(), markAsCompleted(), and getStatus() are named to reflect their function clearly, which improves readability and reduces the need for comments.

Project Overview and Test Cases

The Student Assignment Tracker is a command-line Java application designed to help students organize and manage their coursework. Assignments are grouped by course and tracked based on due date and completion status.

Test cases included:

- Validate that an assignment returns 'Pending' before the due date.
- Validate that it returns 'Overdue' if not completed and past due.
- Validate 'Completed' status if marked complete.
- Ensure courses correctly store assignments.

- Confirm the manager aggregates assignments from multiple courses.

Dependencies

Dependencies used in this Maven-managed Java project include:

- JUnit 5 (org.junit.jupiter:junit-jupiter:5.10.0) for unit testing.
- Maven Surefire Plugin (version 3.0.0-M7) for test execution.

These dependencies were retrieved from the Maven Central Repository.

Challenges Encountered

- Initially included a JUnit version not available in Maven Central (5.9.3). This was resolved by updating to 5.10.0.
- Configuring GitHub Actions correctly required trial and error to ensure tests only ran on dev and main.
- Managing pull requests, branch protection, and CI checks necessitated a strong branching workflow.
- Structuring logic for clean status tracking without user input or external storage required thoughtful design.