MODULE 2 DATABASE PROGRAMMING

Integration Testing

YESTERDAY...

How do we connect to a database?

What is DAO?

What is CRUD?

What is Unit Testing?

Integration Testing

Integration Testing is a broad category of tests that validate the integration between units of code or code and outside dependencies such as databases or network resources.

Integration tests:

- Use the same tools as unit tests (i.e. JUnit or MSTest)
- Usually slower than unit tests (but often still measured in ms)
- More complex to write and debug
- **Can have dependencies on outside resources like files or a database**

Tests should be...

- **Repeatable**: If the test passes/fails on first execution, it should pass/fail on second execution if no code has changed.
- **Independent**: A test should be able to be run on it's own, independently of other tests, OR together with other tests and have the same result either way.
- **Obvious**: When a test fails, it should be as obvious as possible why it failed.

How do we handle the data?

Remotely Hosted Shared Test Database

An RDBMS is installed on a remote server and shared by all developers on the team for testing.

- *Advantages:*
  - Easy setup, often already exists
  - Production-like software and (possibly) hardware
- *Disadvantages:*
  - Unreliable and brittle
  - Lack of test isolation
  - Temptation to rely on existing data (which can change)

Locally Hosted Test Database

An RDBMS is installed and hosted locally on each developer's machine. (This is the approach we will use)

- *Advantages:*
  - Production-like software
  - Reliable (local control)
  - Isolation
- *Disadvantages:*
  - Requires local hardware resources
  - RDBMS needs to be installed and managed

Embedded, In-memory Database

An in-memory, embedded database server is started and managed by test code while running integration tests.

- *Advantages:*
    - Very reliable
    - Consistent across development machines (managed by source control)
    - Lightweight
- *Disadvantages:*
    - Not the same software used in production
    - Cannot use proprietary features of production RDBMS

Still, what about *repeatable?*

A **transaction** is a single unit of work. When it is successful, it should be "committed". If an error is encountered at any point it should be cancelled or rolled back.

Test Decorations

[TestInitialize]

[TestCleanup]

LET'S CODE!

ELEVATE YOURSELF

WHAT QUESTIONS DO YOU HAVE?