

Section Review

Learn to Code with Ruby

Intro to Minitest

- **Minitest** is a testing framework built into the Ruby programming language.
- A test consists of assertions that validate that source code works as expected. It is written in Ruby.
- Use the **require** keyword to import "minitest/autorun".
- To make a test class, declare a Ruby class that inherits from **Minitest::Test**.
- To make a test, declare a method that starts with **test_**. Focus on testing one "unit".

Assertions

- The **assert_equal** method verifies the equality of its two arguments.
- The **assert_includes** method verifies that one element is found within another.
- The **assert_raises** method verifies that a block of code raises a specific exception.
- All assertions accept an optional final argument of a string that provides a custom description in case of failure.

The setup and teardown Methods

- Minitest runs the **setup** method once before each test. Use it to isolate common, duplicated "setup" code.
- Minitest runs the **teardown** method once after each test. Use it to clean up anything the test created (such as deleting files).
- To access data from **setup** in a test, assign the value to an instance variable (@).
- Avoid test pollution. A previous test should not impact a future test. Treat each test as a single, isolated experiment. Tests should be able to run in any order.