Ruby Unit Tests

CSCI400

29 August 2017

Color Key

- Clickable URL link
- Write down an answer to this for class participation
- Just a comment – don't confuse with yellow

Process

1 Create your class with failing mathods

2 Create a test class. Must:
   - require `'minitest/autorun'`
   - Extend `MiniTest::Test`

3 Create one or more test methods
   - Start name with `test` (e.g. `test_conversion`)

4 If separate file, must `require` the class being tested
   - `require_relative "Converter"`
   - `require "./Converter"`

5 Run test program from the terminal
   - All methods beginning with `test` will be executed

Create a Class – Failing Test

```ruby
class Converter
    def feetToMeters(feet)
        return 1
    end
end
```

Unit Test

```ruby
require 'minitest/autorun'
class ConverterTest < MiniTest::Test
    @@EPSILON = 0.0001
    def test_feetToMeters
        converter = Converter.new
        assert_in_delta(3.048,
            converter.feetToMeters(10), @@EPSILON)
        assert_in_delta(0.3048,
            converter.feetToMeters(1), @@EPSILON)
        assert_in_delta(0.4572,
            converter.feetToMeters(1.5), @@EPSILON)
    end
end
```

Can use assert_equal for int

Create a Class – Passing Test

```ruby
class Converter
    @@FEET_TO_METERS = 0.3048
    def feetToMeters(feet)
        return feet * @@FEET_TO_METERS
    end
end
```

Prior Versions of Ruby

- Prior to Ruby 2
    - `require 'test/unit'`
    - `class <ClassName> < Test::Unit::TestCase`

In case you ever run into old examples