

# Testing with Spock

# Spock

- Overview
- Why you want to use it
- Terminology
- IDE integration
- Testing philosophy
- Testing
  - Basics
  - Data-driven tests
  - Testing with database data
  - Mocking
- Extensions
- Links

# What is Spock?

In short...

Spock is a BDD framework written in Groovy

# Why Spock?

- More descriptive method names
- More powerful asserts
- More descriptive test failures
- More concise code
- It's gaining support

# Before we get started...

## A little terminology...

Spock	JUnit
Specification	Test class
setup()	@Before
cleanup()	@After
setupSpec()	@BeforeClass
cleanupSpec()	@AfterClass
Feature	Test
Parameterized feature	Theory
Condition	Assertion
Exception condition	@Test(expected=...)
@FailsWith	@Test(expected=...)
Interaction	Mock expectation (JMock, EasyMock, Mockito)

# Setting things up...

Set up your IDE

You'll need the JUnit, Groovy, and Spock JARs

# Anatomy of a Spock Test

when:

//The stimulus

then:

//The response

These two blocks always occur together

# What if...

The stimulus and the response occur in the same expression?

Use the *expect:* block.



# Extensions

@Timeout

@Ignore

@IgnoreRest

@FailsWith

You can create your own too.

# Links

- Spock homepage: <http://code.google.com/p/spock/>
- Spock web console: <http://meetspock.appspot.com/>
- Groovy webpages: <http://docs.codehaus.org/display/GROOVY/2009/03/10/Meet+Spock!>
- Developer blog: <http://edgibbs.com/category/groovy/>