

iPhone 3.0 Unit Testing

William Taysom
2009 July 16

Why Test?

- Drive API Design
 - Eager Test First
 - Keep on Task
- Debugging
 - Lazy Test Later
 - Keep Bugs Quashed

Which Testing Tool?

- Pre-3.0 ≈ Google Toolbox for Mac



- With 3.0 ≈ Unit Test Bundle



- Both Based on SenTestingKit



How to Unit Test Bundle?

- Two Types:
 - Logic Tests \approx Isolated Classes in the Simulator
 - Application Tests \approx Controllers on the Device

Structure of a Test Case

Structure of a Test Case

```
@interface NameOfTests : SenTestCase @end  
@implementation NameOfTests
```

```
@end
```

Structure of a Test Case

```
@interface NameOfTests : SenTestCase @end  
@implementation NameOfTests
```

- (void)setUp {}
- (void)tearDown {}

```
@end
```

Structure of a Test Case

```
@interface NameOfTests : SenTestCase @end
@implementation NameOfTests

- (void)setUp {}
- (void)tearDown {}

- (void)testTestName {

}

@end
```

Structure of a Test Case

```
@interface NameOfTests : SenTestCase @end
@implementation NameOfTests

- (void)setUp {}
- (void)tearDown {}

- (void)testTestName {
    STAssertTrue(YES, @"YES it's true.");
    STAssertEquals(1 + 1, 3, @"Can always hope.");
}

@end
```

Unit-Test Result Macros

- Unconditional Failure \ni `STFail`
- Equality \ni `STAssertEqualObjects` `STAssertEquals`
`STAssertEqualsWithAccuracy`
- Nil \ni `STAssertNil` `STAssertNotNil`
- Boolean \ni `STAssertTrue` `STAssertFalse`
- Exception \ni `STAssertThrows` `STAssertThrowsSpecific`
`STAssertThrowsSpecificNamed` `STAssertNoThrow`
`STAssertNoThrowSpecific` `STAssertNoThrowSpecificNamed`
`STAssertTrueNoThrow` `STAssertFalseNoThrow`

Follow “Setting Up Logic Testing” in “Unit Testing Applications” of “iPhone Development Guide”

To make sure that tests aren't run when the build fails:

Get Info on the run script.

Append to the script window:

```
touch "$DERIVED_FILE_DIR/mostRecentTest"
```

Add Input Files:

```
$(TARGET_BUILD_DIR)/$(EXECUTABLE_PATH)
```

Add Output Files:

```
$(DERIVED_FILE_DIR)/mostRecentTest
```

To use Precompiled Prefix Header:

Add two Build settings:

```
GCC_PRECOMPILE_PREFIX_HEADER = YES
```

```
GCC_PREFIX_HEADER = project_Prefix.pch
```

Demo

Where to Learn More?

- In “iPhone Development Guide”
 - “Unit Testing Applications” Chapter
 - “Unit-Test Result Macro Reference” Appendix
- `SenTestingKit.framework/Headers`
- `SenTestingKit.framework/*.html`
- iPhoneUnitTests Sample Code

Executed tests,
with 0 failures