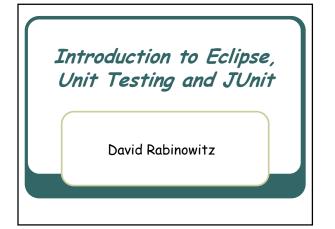
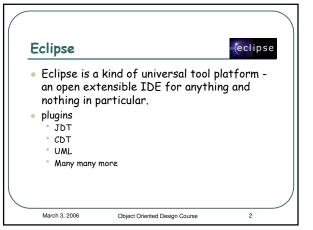
David Talby March 3, 2006





# Working with Eclipse - Live Demonstration Project Views Code Assist Errors Debugging

Object Oriented Design Course

3

Unit Tests

First level of testing

Done by the programmer

Part of the coding process

Delivered with the code

Part of the build process

# What's a Unit?

March 3, 2006

- Unit tests should be written in test classes, that replace 'drivers'
- For example, class Stack
  - \* Class Stack Has push, pop, count, ...
  - Class TestStack Has testPush, testPop
- Tests are per-functionality
  - Not per method (testPushNull)
  - Not per class (testIterators)
- Unit testing can have several levels

March 3, 2006

Object Oriented Design Course

### What's a Unit Test?

Call methods, and assert conditions

void testPush() {
 s = new Stack();
 s.push(new Integer(10));
 assertEquals(s.count() == 1); }

- Tests check themselves
- Only output if a test fails
- Write tests after writing interface
- Run all tests after each change

March 3, 2006

Object Oriented Design Course

David Talby March 3, 2006

### Running Unit Tests

- A main() is required to run tests
- There are better options
  - JUnit for Java
  - CppUnit for C++
- Unit Testing Frameworks
  - Graphical user interface
  - Easily choose which tests to run
  - Elegantly support for test suites

March 3, 2006

Object Oriented Design Course

7

### Coding with Unit Tests

- Part of the design process
  - Design for testability --> Modularity
- Part of the coding process
  - Test-first coding
  - Run tests after each build
- Part of the build process
  - \* Build = compile + link + pass unit tests
  - A.k.a. "smoke tests"

March 3, 2006

Object Oriented Design Course

### Benefits of Unit Tests

- Regression Testing
  - For your own code
  - \* Daily build: for others' work
- Part of the usual work
  - Replaces work done anyway
  - \* Causes tests to be written
  - Validates the design before impl

March 3, 2006

Object Oriented Design Course

## Top 12 Reasons to Write Unit Tests Part I

- Tests Reduce Bugs in New Features
- Tests Reduce Bugs in Existing Features
- Tests Are Good Documentation
- Tests Reduce the Cost of Change

March 3, 2006

Object Oriented Design Course

10

# Top 12 Reasons to Write Unit Tests Part II

- Tests Improve Design
- Tests Allow Refactoring
- Tests Constrain Features
- Tests Defend Against Other Programmers

March 3, 2006

Object Oriented Design Course

11

9

# Top 12 Reasons to Write Unit Tests Part III

- Testing Is Fun
- Testing Forces You to Slow Down and Think
- Testing Makes Development Faster
- Tests Reduce Fear

March 3, 2006

Object Oriented Design Course

12

David Talby March 3, 2006

