SWE 434 Software Testing and Validation

Testing Methods using JUnit

Lab material Courtesy: Dr. M. Shamim Hossain (SWE Department, King Saud University) and Prof. Alan Some (University of Ottawa)

Agenda

- Introduction to IDE (Eclipse)
- Introduction to Software Testing Framework (JUnit)
- Quiz-1

Java Program

1. package lab1;

```
2. public class MyMath
3. {
4. public static void main(String[] args)
5. {
6. int result = MyMath.div(6, 2);
7. System.out.println(result);
8. }
9. public static int div(int a, int b)
```

```
10. {
11. return a / b;
12. }
```

13.}

Where can I Find Eclipse (IDE)

• Eclipse downloads are available on

https://www.eclipse.org/downloads/

•Where else can I find an IDE with Junit

IBM Rational Software Architect: Comes with Junit

Java Runtime Environment(JRE 7)

- If JRE is not installed in system, then download it from
 - <u>http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html</u>

| Java SE Runtime Environment 7u13 | | | | |
|----------------------------------|-----------------|-------------------------------------------|--|--|
| You must accept the Oracle Bir | nary Code Licen | se Agreement for Java SE to download this | | |
| | softwa | ire. | | |
| Accept License Agreement | Decline Lice | nse Aareement | | |
| | | | | |
| | | | | |
| | | | | |
| Product / File Description | File Size | Download | | |
| | 1110 0120 | | | |
| Linux x86 | 54.7 MB | 🛓 jre-7u13-linux-i586.rpm | | |
| Linux x86 | 45.9 MB | 🛓 jre-7u13-linux-i586.tar.gz | | |
| Linux x64 | 52.84 MB | 🛓 jre-7u13-linux-x64.rpm | | |
| Linux x64 | 44.63 MB | 🛓 jre-7u13-linux-x64.tar.gz | | |
| Mac OS X x64 | 50.32 MB | 🛓 jre-7u13-macosx-x64.dmg | | |
| Mac OS X x64 | 46.66 MB | 🛓 jre-7u13-macosx-x64.tar.gz | | |
| Solaris x86 | 45.41 MB | 🛓 jre-7u13-solaris-i586.tar.gz | | |
| Solaris x64 | 14.8 MB | 🛓 jre-7u13-solaris-x64.tar.gz | | |
| Solaris SPARC | 48.7 MB | 🛓 jre-7u13-solaris-sparc.tar.gz | | |
| Solaris SPARC 64-bit | 17.4 MB | 🛓 jre-7u13-solaris-sparcv9.tar.gz | | |
| Windows x86 Online | 0.86 MB | jre-7u13-windows-i586-iftw.exe | | |
| Windows x86 Offline | 30.05 MB | jre-7u13-windows-i586.exe | | |
| Windows x86 | 39.77 MB | 🛓 jre-7u13-windows-i586.tar.gz | | |
| Windows x64 | 31.47 MB | jre-7u13-windows-x64.exe | | |
| Windows x64 | 41.49 MB | 보 jre-7u13-windows-x64.tar.gz | | |

- **Download** the relevant installation package, and
- Do the JRE 7 **installation**!

Integrated Development Environment (IDE)

• <u>Eclipse</u>

– It should be in following folder eclipse-SDK-3.7.2-win32\

| Name | Date modified | Туре | Size |
|-------------------|--------------------|--------------------|-------|
| \mu configuration | 2/19/2011 12:09 AM | File folder | |
| 퉬 features | 10/10/2012 7:45 PM | File folder | |
| 퉬 plugins | 10/10/2012 7:45 PM | File folder | |
| 퉬 readme | 10/13/2010 5:44 AM | File folder | |
| 퉬 workspace | 2/19/2011 12:15 AM | File folder | |
| .eclipseproduct | 10/23/2007 8:43 PM | ECLIPSEPRODUCT | 1 KB |
| 😂 eclipse | 10/23/2007 8:43 PM | Application | 56 KB |
| eclipse | 10/22/2012 6:11 PM | Configuration sett | 1 KB |
| eclipsec | 10/23/2007 8:43 PM | Application | 28 KB |
| 🔁 epl-v10 | 10/23/2007 8:43 PM | Chrome HTML Do | 17 KB |
| | | | |

- Please open the **eclipse.exe**



Integrated Development Environment (IDE)



Department of Software Engineering, King Saud University

Software Testing

The **process** of executing a **program** with the intent of finding errors.

Myers, 1979

Unit Testing

 The process of validating a piece of code (function or method) with required input and expected output.

Test Case and Test Execution System

• Requirements for automated testing?

1. <u>Test Case</u>

"A test case is a small unit of code that tests a specific method"

- **INPUT:** Actions send to System Under Test (**SUT**).
- **OUTPUT:** Responses expected from **SUT**.
- VALIDATION: To Check the *Test*, was successful or not?
- 2. <u>Test Execution System</u>
 - Mechanism to read test scripts, and connect test cases to **System Under Test**, for example <u>JUNIT</u>.
 - Keeps track of test results.

Software Testing Process

Test case verdicts

- A **verdict** is the declared result of executing a single test.
 - We can get the verdict after the assertion statement
- **Pass**: the test case achieved its intended purpose, and the software under test performed as expected.
- **Fail**: the test case achieved its intended purpose, but the software under test did not perform as expected.
- **Error**: the test case did not achieve its intended purpose.
 - Potential reasons:
 - An unexpected event occurred during the test case.
 - The test case could not be set up properly

JUnit

- A Unit Testing Framework for Java
 - It helps us to test the specific method of class
 - Authors: Erich Gamma, Kent Beck
 - <u>http://www.junit.org</u> or
 - <u>http://sourceforge.net/projects/junit/</u>
 - Plug-in is already installed in Eclipse.
 - Recommended Version: 4.3.1+

Practical Lab (Example-1)

- <u>Problem Description:</u>
 - 1. Implement the method "**div**(**a**, **b**)" for division of two integer numbers.
 - 2. Implement the *Test Case* by using the JUnit framework.
- <u>Objective:</u>
 - Creating the new java project in Eclipse through wizard
 - Creating the new class (SUT) and its implementation
 - Creating the test case class in JUnit through wizard
 - Implementing the test case method
- Create a new Java project in Eclipse.
 - When creating the project, be sure the following is set up in the new project wizard:

Create a New Java Project

• Right click anywhere in package explorer, or go to in File menu.

| 🕽 Java - | Eclipse SDK | | - | - | - | a second | | |
|--------------------------|----------------------------|---------------------------------|------------------|--------------------------|-----------------|----------------|--------------|-----|
| <u>F</u> ile <u>E</u> di | t <u>R</u> un <u>S</u> our | rce Refac <u>t</u> or | <u>N</u> avigate | Se <u>a</u> rch <u>I</u> | <u>P</u> roject | <u>W</u> indow | <u>H</u> elp | |
| 📬 🗸 | 🖫 🖆 🚦 🚺 | 😫 J🖁 🛛 | 🕯 🔉 - | 🍰 🔅 | - 🕶 🜔 | - 💁 - | / 🗳 | G . |
| 🕆 Pack | age E 🛛 | Hierarchy | Ju JUnit 🗧 | ° •) (–) | | | | |
| | | $\Leftrightarrow \Rightarrow 6$ | è 🖻 🔄 |] ~ | | | | |
| | | | | | | | | |
| | New | | | • | 1 | Java Project | t | |
| | Show In | | Alt | -Shift+W ► | 2 | Android Pro | oject | ٦. |
| æ | Conv | | | Ctrl+C | C2 | Project | | |
| | Copy Qualif | ied Name | | Curre | ₿ | Package | | |
| | Paste | | | Ctrl+V | G | Class | | |
| × | Delete | | | Delete | 0 | Interface - | | |
| | Build Path | | | + | G | Enum | | |
| | Import | | | | e E | Source Fold | ler | |
| | Export | | | | | Folder | | |
| e Berner Berner | Refresh | | | F5 | | File | + File | |
| | Properties | | , | Alt+Enter | | Android XN | AL File | |
| | | | | | Ē | JUnit Test C | Case | |
| | | | | P | | Example | | rat |
| | | | | Andr | € ک | Other | | |

Create a Java Project

- On the "Create a Java Project" screen
 - Enter a project name,
 - Select the option "Create separate source and output folders."
 - The exact version of the JRE is not important, but it should be version 1.5.0 or greater.
 - Then, click Next.

| New Java Project | | | | |
|---------------------------------------------------------|--------------------|-----------------|-----------|---------------------|
| eate a Java project eate a Java project in the works | pace or in an ext | ernal location. | | |
| roject name: Lab 1 | | | | |
| Contents | | | | |
| • Create new project in work | space | | | |
| C Create project from existin | g source | | | |
| Directory: C:\Documents and | Settings',awilliam | \My Documents\N | Workspace | Browse, |
| | | | 63 | |
| JRE | | | | |
| • Use default JRE (Currently | 'jre1.5.0_11') | | Con | figure JREs |
| O Use a project specific JRE: | jre1.5.0_11 | - | | |
| Project lavout | | | | |
| C Lice project folder as reat f | or courses and d | acc filoc | | |
| Create constate course an | d output folders | | | 1.0 |
| Se Create separate source an | a oacpacitoiders | | Confid | <u>jure default</u> |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | < Pack | New A | Finish | Canad |
| 21 | C Dell'r | Nevr 3 | EUDSIL | |

Prepared by Raja Majid Mehmood

Department of Software Engineering, King Saud University

Step-1.1

Java Settings

• On the "Java Settings" screen, click on "Create a new source folder".

| aua Cottinge | | | |
|------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|---------------------------|-------------------|
| efine the Java build set | ttings. | | |
| Ø Source | iects 🚔 Libraries 🍫 Ord | er and Export | a 🖉 🕹 🕥 |
| ✓ Details | ource folder : use this if you wa | ant to add a new source | folder to your |
| Ink additional as additional s as additional s Add project 'La Allow output folder | <u>l source</u> : use this if you have a source folder. <u>ab 1' to build path</u> : Add the pro rs for source folders : | folder in the file system | that should be us |
| | | | |

Step-1.2

New Source Folder

• Name the new source folder **test**. Click Finish.

| 🗲 New Source Fol | der | | | × |
|--------------------------------------|---------------------|----------|--------|--------|
| Source folder Add a new source fo | older relative to ' | /Lab 1'. | | |
| Folder name: test | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| 0 | < Back | Next > | Finish | Cancel |

Step-1 Completed

- The result should be two source folders, **src** and **test**.
- We are going to store the code to be tested in the **src** folder,
- and the test cases in the **test** folder.

Create a <u>Package</u> in <u>src</u>

- Create package **lab1** in **src** folder,
- Right Click on **src**,
- see the steps in this figure,

Create a <u>Package</u> in <u>src</u>

- Enter **<u>lab1</u>** in Name field.
- Press **Finish** Button

| New Java Package | |
|-------------------------------------------------------------------------------------|--------|
| Java Package Create a Java package. | |
| Creates folders corresponding to packages. Source folder: Lab1/src Name: lab1 | Browse |
| ⑦ Finish | Cancel |

Create a <u>Package</u> in <u>test</u>

- Create package **lab1** in **test** folder,
- Right Click on **test**,
- see the steps in this figure,

Create a <u>Package</u> in <u>test</u>

- Enter **<u>lab1</u>** in Name field.
- Press **Finish** Button

| New Java Package | |
|-----------------------------------------------|--------|
| Java Package Create a Java package. | |
| Creates folders corresponding to packages. | |
| Source folder: Lab1/test | Browse |
| Name: lab1 |] |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| ⑦ Finish | Cancel |

Package View

 Now, we already created the package lab1 in both src and test folders.

Create a Class to Test

- *Right click* on **lab1** under **src** folder, and
- Click the **Class**

Create a Class to Test

- Enter class name **MyMath**
- Press Finish

| 😂 New Java Class | | |
|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|------------|
| Java Class Create a new Java c | lass. | C |
| Source folder: | Lab1/src | Browse |
| Package: | lab1 | Browse |
| Enclosing type: | lab1.CopyOfMyMath | Browse |
| Name: Modifiers: | MyMath pupile derault private protected abstract final static |] |
| Superclass: | java.lang.Object | Browse |
| Interfaces: | | Add Remove |
| Which method stub | s would you like to create? | |
| | public static void main(String[] args) Constructors from superclass Inherited abstract methods | |
| Do you want to add | comments as configured in the <u>properties</u> of the current p | project? |
| 0 | Finish | Cancel |

Create a Class to Test

• Add the following code in class **MyMath** within the **lab1** package of the **src** folder.

```
1.package lab1;
2.public class MyMath
3.{
4. public int div( int a, int b )
5. {
6. return a / b;
7. }
8. }
```

• The class has one method, **div(int, int)**, which performs some computation.

Class View (SUT)

| 😂 Java - Lab1/src/lab1/MyMath.java - | Eclipse SDK |
|--------------------------------------|------------------------------------------------------------------------------|
| <u>File Edit Run Source Refactor</u> | <u>N</u> avigate Se <u>a</u> rch <u>P</u> roject <u>W</u> indow <u>H</u> elp |
| 📬 🖬 🖨 📑 😫 J🖁 🖸 | ▓▾ఊ ॐ▾◐▾∿▾ ఊ ☞▾ ៚ ℤ ◙ ◎ ॥ ₰▾∛▾ゃ |
| 🛱 Pa 🕄 隆 Hi 🔂 J 🖓 🗖 | 🕑 MyMathTest.java 🛛 🚺 MyMath.java 🖾 |
| ⇔ ⇔ @ 🖻 🔄 ▽ | 1 package lab1; |
| 🔁 Lab1 | 2 3 public class MyMath |
| 🕮 src | 4 { |
| 🖶 lab1 | 5 /** |
| 🚺 MyMath.java | 6 * perform division of two integer numbers |
| 进 test | 7 * |
| 🖶 lab1 | 8 * |
| 🚺 MyMathTest.java | 9 * @param a, must be integer |
| 🛋 JRE System Library [jre7] | 10 * Oparam b, must be integer |
| 🛋 JUnit 4 | 11 * Greturn, division result of a and b |
| 🛋 UML | 13Θ public int div(int a, int b) |
| | 14 { |
| | 15 return a / b; |
| | 16 } |
| | 17 } |

Create a Test Case Class

- Select the class **MyMath**, and
- Right-click to get the popup menu.
- Select New > JUnit Test Case.

New (JUnit Test Case)

- 1. Select JUnit 4
- Select the source folder of the test case to be lab1/test. To do this, click on the Browse... button on the same line and select the test folder.
- 3. [Optional] In the section "Which method stubs would you like to create?" check all of setUpBeforeClass(), tearDownAfterClass(), setUp(), and tearDown(). This will create methods with the annotations @BeforeClass, @AfterClass, @Bef ore, and @After, respectively.
- 4. If you would like automatically generated comments, check the "Generate comments" box.
- 5. Click <u>NEXT</u>

| Unit Test Case Select the name of the new JUnit test case. You have the options to he class under test and on the next page, to select methods to be New JUnit 3.8.1 test New JUnit 3.8.1 test Source folder: Lab 1/test Package: Iab1 Vame: MyMathTest Superclass: java.lang.Object Which method stubs would you like to create? SetUpBeforeClass() tearDownAfterClass setUp() tearDown() constructor Do you want to add comments as configured in the properties of the Class under test: Iab1.MyMath | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| he class under test and on the next page, to select methods to be New JUnit 3.8.1 test New JUnit 3.8.1 test New JUnit 4 test Lab 1/test Lab 1/test Lab 1/test Lab 1/test Name: MyMathTest Superclass: Java.lang.Object Which method stubs would you like to create? SetUpBeforeClass() tearDownAfterClass SetUp() tearDown() Constructor Co you want to add comments as configured in the properties of th Generate comments Class under test: Iab1.MyMath | specify |
| New JUnit 3.8.1 test New JUnit 4 test Source folder: Lab 1/test Package: Iab1 Name: MyMathTest Superclass: java.lang.Object Which method stubs would you like to create? setUpBeforeClass() setUpBeforeClass() tearDownAfterClass setUp() tearDown() constructor constructor Do you want to add comments as configured in the properties of th Generate comments Class under test: Iab1.MyMath | ested. |
| Source folder: Lab 1/test Package: lab1 Name: MyMathTest Superclass: java.lang.Object Which method stubs would you like to create? SetUpBeforeClass() tearDownAfterClass SetUp() tearDown() Constructor Do you want to add comments as configured in the properties of th Generate comments Class under test: lab1.MyMath | |
| Package: Iab1 Vame: MyMathTest Superclass: Java.lang.Object Which method stubs would you like to create? SetUpBeforeClass() tearDownAfterClass SetUp() tearDown() Constructor Do you want to add comments as configured in the properties of th Generate comments Class under test: Iab1.MyMath | Browse |
| Name: MyMathTest Superclass: java.lang.Object Which method stubs would you like to create? setUpBeforeClass() tearDownAfterClass setUp() tearDown() constructor Do you want to add comments as configured in the properties of th Generate comments Class under test: lab1.MyMath | Browse |
| Superclass: java.lang.Object Which method stubs would you like to create? setUpBeforeClass() tearDownAfterClass setUp() tearDown() constructor Do you want to add comments as configured in the properties of th Generate comments Class under test: lab1.MyMath | _ |
| Which method stubs would you like to create? setUpBeforeClass() tearDownAfterClass() setUp() tearDown() constructor Do you want to add comments as configured in the properties of th Generate comments Class under test: lab1.MyMath | Browse, |
| Class under test: lab1.MyMath | current project? |
| | Browse |
| | |
| | |
| | -1 |

Step-5.3

Test Methods

• Select the method **div(int,int)** for testing, and then click Finish.

| 🚝 New JUnit Test Case | • | | | × |
|------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|-------------------|--------|----------------------------|
| Test Methods Select methods for which | test method stu | ubs should be cre | ated. | E |
| Available methods: | String[]) t, int) tt() ass() Code() ls(Object) () () () () () () () () () () () () () | hods | | Select All Deselect All |
| 0 | < Back | Next > | Finish | Cancel |

Choosing JUnit-4 Library [Optional]

- Select the highlighted option
 - Click on **OK**

| New JUnit Test Case |
|----------------------------------------------------------|
| JUnit 4 is not on the build path. Do you want to add it? |
| ⊘ Not now |
| Open the build path property page |
| Perform the following action: |
| Add JUnit 4 library to the build path |
| |
| |
| |
| OK Cancel |
| |

Test Case Class

A new file **MyMathTest.java** will be created with a new class **MyMathTest**. The file will be opened in a new editor window, and the contents should resemble the following code:

```
package lab1;
import static org.junit.Assert.*;
import org.junit.Test;
public class MyMathTest {
                                                Remove this line
        @Test
        public void testDiv() {
          fail("Not vet implemented");
        }
}
```

Proposed Test Case

- Test Case for System Under Test (SUT)
 - MyMath.div(int, int)
 - This method has two integer parameters and it will return the division of two input numbers (e.g. 6 and 2).

| TEST CASE | INPUT | | EXPECTED OUTPUT | | |
|-----------|-----------|-----------|--------------------------------|--|--|
| | INTEGER-A | INTEGER-B | INTEGER-A/INTEGER-B = ? | | |
| TC-1 | 6 | 2 | 3 | | |

Implementation of Test Case

- Implementation of **TC-1** in **testDiv()**
 - Add the required code in **MyMathTest**

Run the JUnit Test Case

| Java - Lab1/test/lab1/MyMathTest.java - Eclipse SDK | | | | | | | | |
|--------------------------------------------------------------------------|------------|----------------------------|---------------|-----------------------------------------------------|--|--|--|--|
| <u>File Edit Run Source Refactor Navigate Search Project Window Help</u> | | | | | | | | |
| | | | | | | | | |
| 📕 Pa 🛛 🔹 Hie 🚮 JU 🖓 🗖 🕖 MyMath.java 🕖 MyMathTest.java 🕄 | | | | | | | | |
| | | | | | | | | |
| 🛛 🔁 Lab1 | | New | • | *; | | | | |
| ⊿ 📛 src | | -UMD | | | | | | |
| J MyMath.java | | Open | | | | | | |
| ⊿ 🇀 test | | Open With | | | | | | |
| Iabl | | Open Type Hierarchy | F4 | 11(); | | | | |
| JRE System Library [jre7] | | Show In | Alt+Shift+W ► | | | | | |
| ⊳ 🛋 JUnit 4 | | <u> </u> | 01.0 | iv(6, 2); | | | | |
| D 🛋 UML | | Copy | Ctrl+C | | | | | |
| | | Copy Qualified Name | Ctrl+V | actual); | | | | |
| | 110 110 | Delete | Delete | | | | | |
| | • | | Delete | | | | | |
| | | Build Path | • | | | | | |
| | | Source | Alt+Shift+S ► | | | | | |
| | | Refactor | Alt+Shift+1 | | | | | |
| | è | Import | | | | | | |
| | <u>ک</u> | Export | | | | | | |
| | | References | ÷. | | | | | |
| | | Declarations | + | | | | | |
| | | Refresh | F5 | | | | | |
| | | Assign Working Sets | | | | | | |
| | | Run As | • | 1 Run on Server Alt+Shift+X. R | | | | |
| | | Debug As | • | Ju 2 JUnit Test Alt+Shift+X, T | | | | |
| | | Profile As | • | Onen Run Dialog | | | | |
| | | Validate | | o open kan bialog | | | | |
| | | Team | ۱. | ram Files\Java\jre7\bin\javaw.exe (Feb 19, 2013 1:5 | | | | |
| | | Compare With | + | | | | | |
| | | Replace With | Þ | | | | | |
| | | Restore from Local History | | | | | | |
| | | eUML2 | + | | | | | |
| | | Web Services | ۱. | | | | | |
| ☐ [♦] lab1.MyMathTest.java | | Properties | Alt+Enter | | | | | |

Test Verdict - PASS

• Finally, we got the green bar, our test is successful.

QUIZ - 1

- Practical Lab Quiz
 - Date: Next Week
 - **Time:** 02:00PM
- Topics Covered
 - Junit test method implementation
- Rules

– Absence = Zero (o) Points