How to do automated unit testing

Testing is repetitive

- Testing is a repetitive task
  - You have to test new software that you have written
  - You also have to test the software **everytime after you have made a change or you have fixed a bug** (this is called “Regression testing”)

- Conclusion: Try to automate the testing as much as possible, so you can redo the tests everytime you change the program
  - How? Write a program to test your program!

Automated unit testing

- Example: We want to test this method in class `Main`:

```
static int min(int a, int b)
```

- Our test program:

```
// test case 1
int r1=Main.min(3,5);
if(r1!=3) {
  System.out.println("Test 1 failed");
}
// test case 2
int r2=Main.min(5,3);
if(r2!=3) {
  System.out.println("Test 2 failed");
}
```

- We can run this test program every time we have made a change in the method `min`
- This is a lot of code for just three simple tests...
  - Let's make life easier: Let's use the JUnit unit testing tool

Write unit tests with JUnit4

```
import static org.junit.Assert.*;

public class MainTest {

    @org.junit.Test
    public void testFirstNumberLessThanSecondNumber() {
        assertEquals("Minimum of 3 and 5 should be 3", 3, Main.min(3,5));
    }

    @org.junit.Test
    public void testFirstNumberGreaterThanSecondNumber() {
        assertEquals("Minimum of 5 and 3 should be 3", 3, Main.min(5,3));
    }
}
```

- The static assertEquals method does the same as our code on the previous slide.
- There are also methods to test inequality etc. Check the documentation on https://junit.org/junit4/javadoc/4.12/org/junit/Assert.html