# Lab lecture exercises – 7 October 2016

1. **Set Up** Add to your `.cshrc` file the following lines (see also on Canvas `cshrc-add`):

```
module add java/1.8.0
module add eclipse/neon
setenv CLASSPATH .:/bham/pd/packages/java/1.8.0/lib/hamcrest-core-1.3.jar
                 :/bham/pd/packages/java/1.8.0/lib/junit-4.12.jar
alias junit 'java org.junit.runner.JUnitCore'
```

2. **JUnit – Terminal**
   (a) Close the terminal by inputting `exit`. Open a Terminal.
   (b) Copy the file `https://canvas.bham.ac.uk/courses/21955/files/3262280` to `JUnitExample.java`.
   (c) Compile the file by `javac JUnitExample.java`.
   (d) Run the tests by `junit JUnitExample`.

3. **Eclipse**
   (a) Open `Eclipse` by entering in a terminal the command `eclipse &`.
   (b) Enter a file space where you want `Eclipse` write all your Java files.
   (c) Possibly close the Welcome window (by clicking the x).
   (d) Start a new `Project` by clicking `Java Project` following the `File` and then the `New`. Name the new project `Week2`, and press the `Finish` button.
   (e) Create a `Class` following `File` and `New` again.
   (f) Give the class the Name `HelloWorld` and press the `Finish` button.
   (g) Enter in the `HelloWorld.java` window the `HelloWorld` program.
   (h) Run it by clicking the white arrow in the green circle.

4. **JUnit – Eclipse**
   (a) Create a JUnit test case with `JUnitExample` (following the `File` and `New` again).
   (b) Copy the text from the link above into the new window.
   (c) Click on the first red error message, choose the last option "Fix project setup ...", then "Add JUnit 4 library to the build path".
   (d) Run the tests by clicking the white arrow in the green circle. (You have to click again okay in a pop-up window.)

5. **Class**
   Write a class `Patient` with the field variables `name`, `id`, `address`, `phoneNumber` all of type `String`. Generate a `constructor`, `getters`, `setters`, and a `toString` method following the `Source` tab in `Eclipse`. Add an `equals` methods.