

Java Testing and Docker Task:

1) Make sure you installed IntelliJ Community edition

You can use this video (skip to **02:20**)

<https://youtu.be/T9xXZSx5M>

- Create a Command Line App (“Create project from template”)
+ install *openjdk-15.0.1*

- Copy the following code into Main.Java:

```
package com.example;
public class Main {
    public static void main(String[] args) {
        System.out.println("Hello world!");
    }
}
```

Run Main and see the “Hello World!” in the output console

- Create Test1 class and copy this code:



(right-click the *com.example* > new > Java Class)

```
package com.example;
import org.junit.Test; // junit:junit:4.12
import static org.junit.Assert.assertEquals;
public class Test1 {
    @Test
    public void test1() {
        assertEquals(1, 1);
    }
}
```

right-click Test1.java > Run Test1. Did it pass?

Now ... modify `assertEquals(1, 2);`

right-click Test1.java > Run Test1. Did it pass?

2) Build + Run Junit from the command line:

- a. Close the IntelliJ
- b. Install Java 8
- c. Create a new **empty folder** and create the following:

Example.java

```
public class Example {  
    public String getResult(int number) {  
        return "1";  
    }  
}
```

ExampleTest.java

```
import org.junit.Test;  
import static org.junit.Assert.assertEquals;  
public class ExampleTest {  
    @Test  
    public void returnsOnelfGivenOne() {  
        Example example = new Example();  
        String result = example.getResult(1);  
        assertEquals("1", result);  
    }  
}
```

- d. download and copy **junit-4.12.jar** file into the same folder
 - e. download and copy **hamcrest-core-1.3.jar** file into the same folder
 - f. Open command prompt and goto **THIS** folder, and run:
 - i. `"C:\Program Files\Java\jdk1.8.0_281\bin\javac.exe"
Example.java ExampleTest.java -cp .;junit-4.12.jar`
 - ii. `java -cp .;junit-4.12.jar;hamcrest-core-1.3.jar
org.junit.runner.JUnitCore Exampletest`
- assuming you are using javac.exe jdk1.8, check it by: javac -version
 - assuming you are using java.exe jdk1.8, check it by: java -version

3) Jenkins with Java

Take the JAVA code from task 2 (*Example.java, ExampleTest.java*)

- Upload it into git
- Create a Jenkins build:
 - o git clone the folder
 - o copy the jar files into this folder
 - o compile the code, using: javac.exe ...
 - o run the Junit tests, using: java.exe ...
 - o Run a pipeline when the assert is true and the test pass
 - o Change the assert to fail and run the flow

4) *Etgar:

- Create a simply pytest code which asserts a true condition
- Create a Jenkins build which git clones the code + run the pytest

5) Docker

- Install and run Docker on Virtual box CentOS.
- see instructions here:
<https://devopsprogroup.wordpress.com/2021/02/19/java-and-junit-and-docker/>
- Check it works:

```
$ sudo systemctl start docker
```

```
$ sudo docker run hello-world
```

- *Etgar: install Jenkins on Docker, can you run one of the tasks?

For submission:

Please make screenshots of the IntelliJ work

Please upload to GIT your config.xml of the .jenkins > jobs for the 3 projects

Please upload screenshots of the working Docker

Good luck

Have a nice Java day 😊