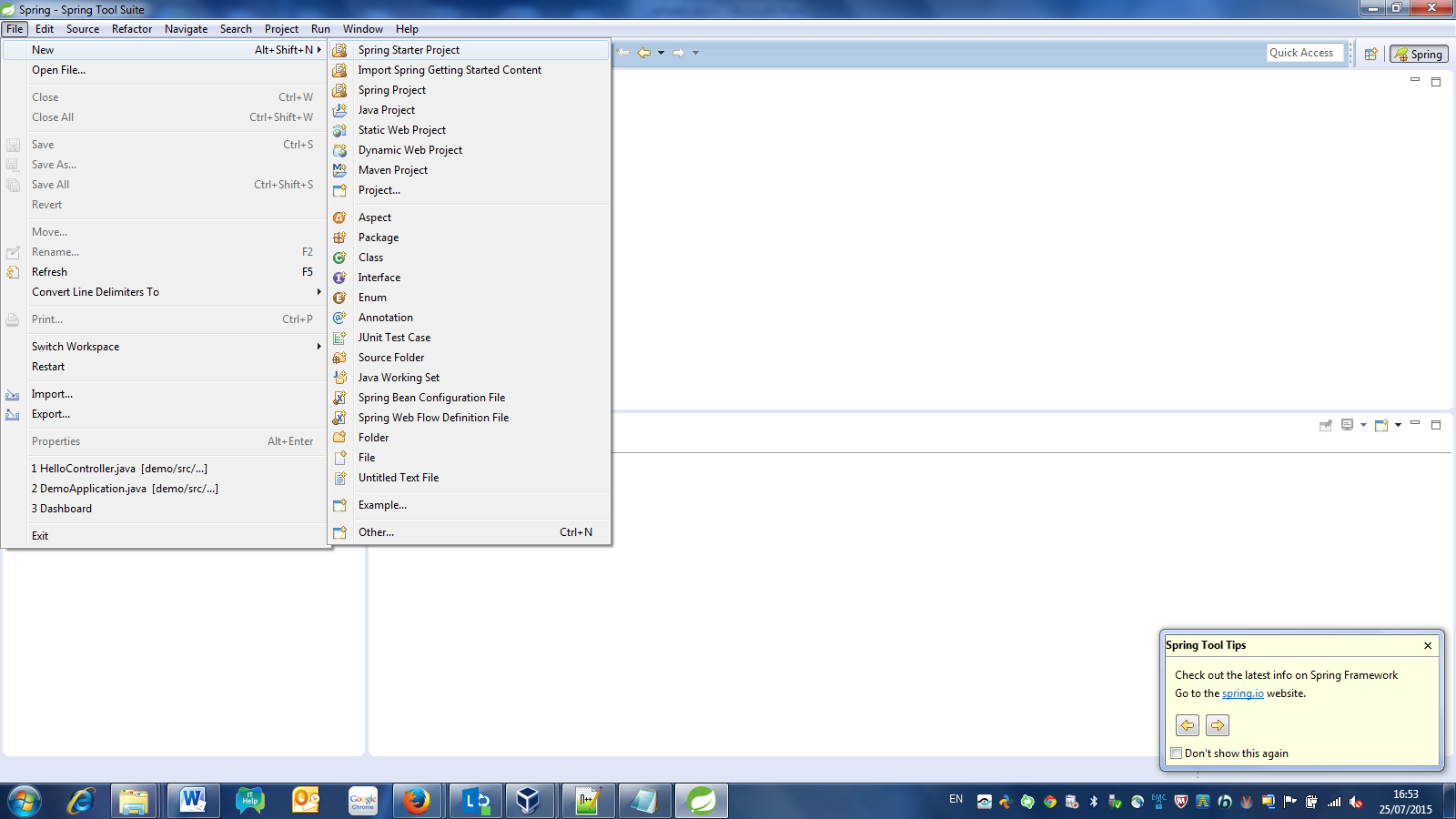
The documented flow is full cycle of create and use of the simple docker image containing Hello World Spring Boot app.

1.

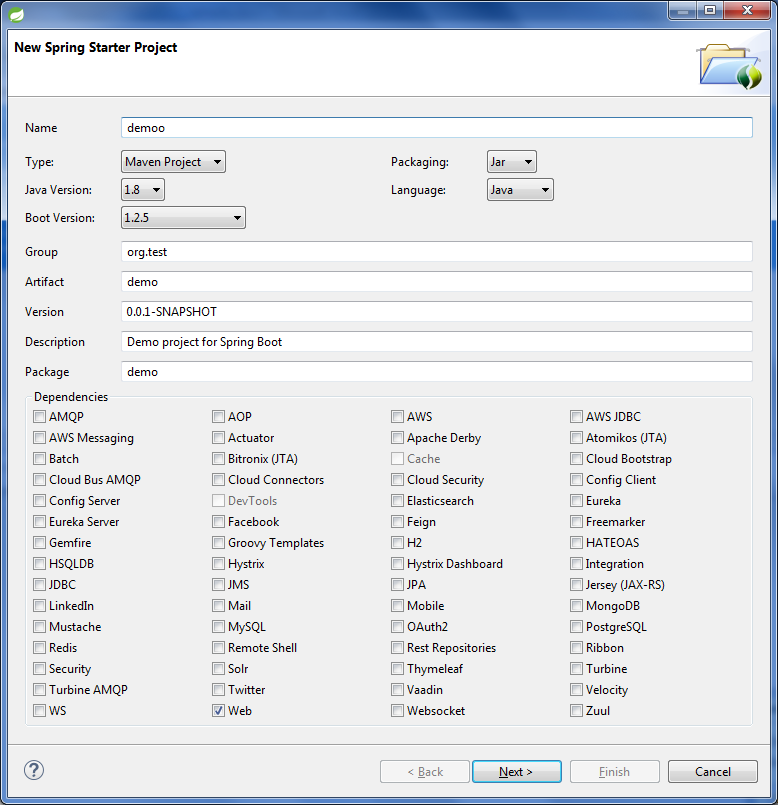
Create a simple Hellow World Spring Boot app that reacts on URL like <http://localhost:8080/hello?name=Boris> and returns the "Hello " + name (name is the passed name parameter). Alternatively, import the code from C:\rp-all-info\docker\spring-boot-hello-image-full-cycle\app-src\demo to the STS.

1.1

Select Spring Starter type of project



Select Web for dependency



1.2

That is all the code that is needed

package demo;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.bind.annotation.RestController;

@RestController

public class HelloController {

@RequestMapping("/hello")

public String hello(@RequestParam String name) {

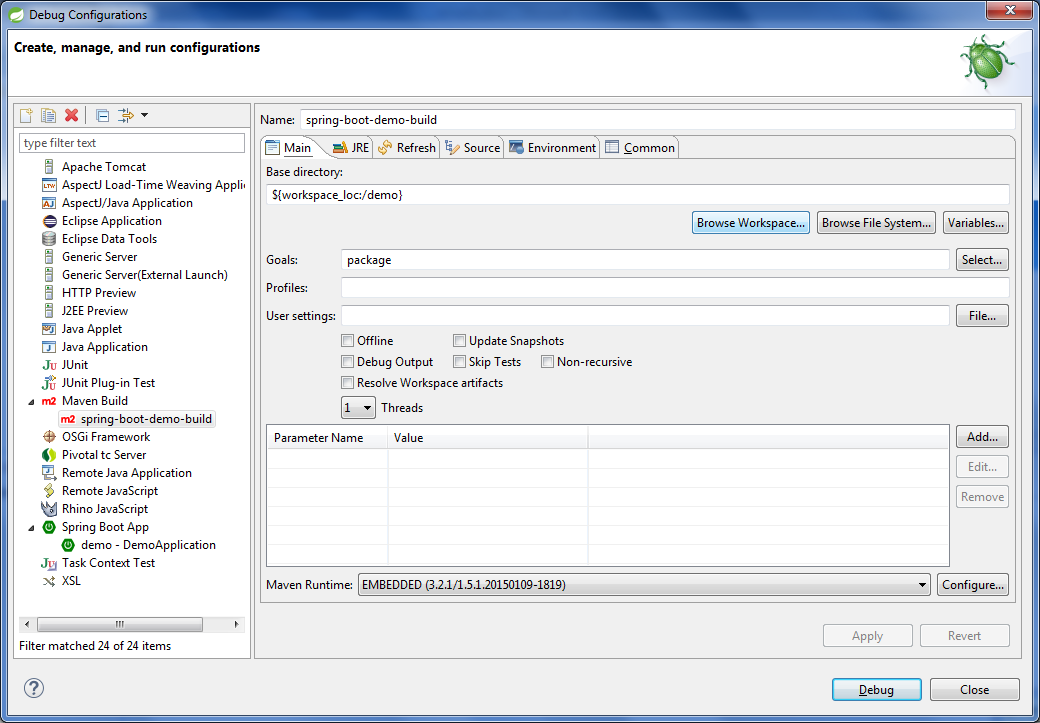
return "Hello "+name;

}

}

1.3

To get the fat jar that contains all the app and embedded Tomcat, create the following “Maven build” under “Debug configurations”. Pay attention – browse workspace, in goals put “package”. Press on “Apply”-> “Debug”. Observe the output on console and take the JAR file from location that is specified there.



2.

Upload the JAR to GitHub so it will be possible to download it to Linux.

3.

Those are instructions on Linux machine for creating and using container.

3.1

**All the instructions in this section explain how to create a correct Docker file and put a) JAR file of the app b)created Dockerfile under docker/hello folder. Those are all needed artifacts for image creation.**

Download the demo-0.0.1-SNAPSHOT.jar from GitHub as a part of full project ZIP download.

--mkdir docker

--cd docker

--mkdir hello

--cd hello

Copy the demo-0.0.1-SNAPSHOT.jar to docker/hello folder – regular right click->copy will work.

--touch Dockerfile

Double click Dockerfile to edit. Put the following as content of the file. Save.

FROM java:8-jre  
ADD demo-0.0.1-SNAPSHOT.jar /app/  
CMD ["java", "-jar", "/app/demo-0.0.1-SNAPSHOT.jar"]  
EXPOSE 8080

3.2

To create a shpboris/hellospringboot image do from the same directory in Terminal where Dockerfile and JAR reside (i.e docker/hello)

--sudo service docker start  
--sudo docker build -t shpboris/hellospringboot .

3.3

To start container and access Web app do

--sudo docker run -p 8080:8080 -t shpboris/hellospringboot

Then type in browser something like <http://localhost:8080/hello?name=Boris>. That’ it – app is working.

3.4

To push image to DockerHub do

--sudo docker push shpboris/hellospringboot

When asked for email, copy [shp\_boris@yahoo.com](mailto:shp_boris@yahoo.com) address from your email in order to avoid typing ‘@’.

Access your DockerHub account – new repository shpboris/hellospringboot was created there.