Appendix 04 Run Spring Initializr Project

Description: This document builds on generating a Spring Initializr website project “Appendix 02 Spring Initializr”, importing the project into Spring Tool Suite IDE “Appendix 03 Import Spring Tool Suite Project”, modifying the base code then run the project.

Project: This appendix shows a step by step guide to run a Spring Boot project from the command line, within the IDE, and building an executable jar. This project will show the steps to run a demo project based on the modification from document title “Spring Boot 100 Hello World”. Using the “demo-spring-boot-hello” project will allow showing some basic test steps as well.

Technology: This project uses the following technology:

Integrated Development Environment (IDE):

Spring Tool Suite 4 (Version: 4.11.0.RELEASE)

Java Development Kit (JDK):

Oracle's JDK 8 (1.8)

Table of Contents

Running Spring Boot Projects

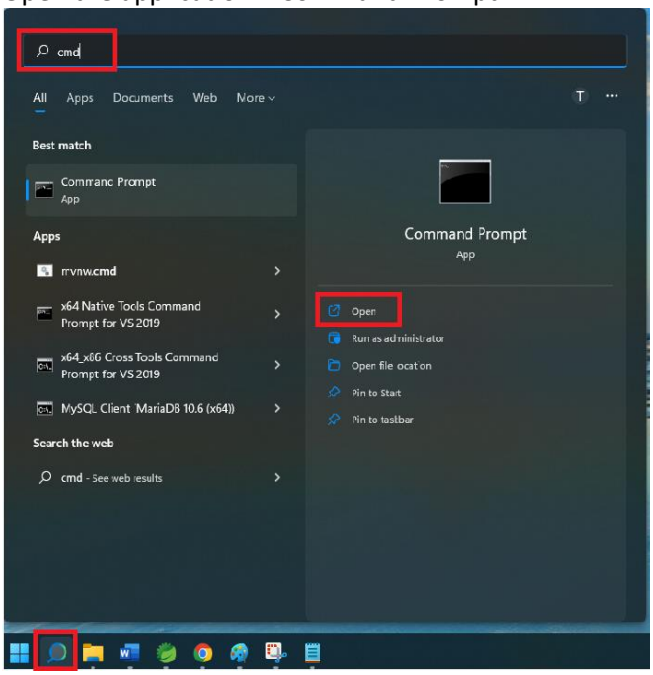
The basic project generated by the Spring Initializr website is designed to be built and run from the command line. These projects can also be run inside the Spring Tool Suite IDE.

This appendix show the command line interface for building and running a Spring Initializr generated project. This appendix shows funning a Spring Initializr generated project within the IDE.

Both the command line interface and the IDE will use the project completed in the document titled "Spring Boot 01 Hello World" with the project name "demo-spring-boot-hello".

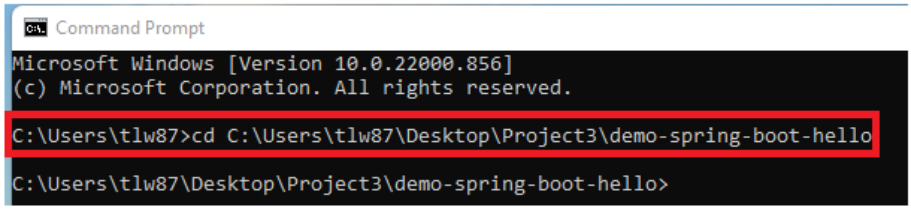
Run Spring Boot Project with Command Line Interface

Open the Windows command line interface.
In the Windows search window enter: "cmd"
Open the application: "Command Prompt"



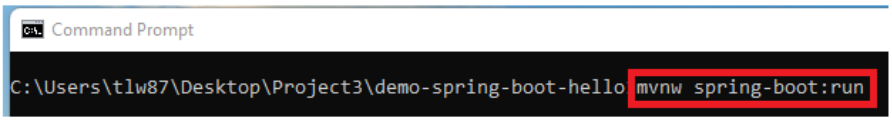
Change the directory to the location of your project's top level directory where the mvnw.cmd file resides.

```
cd C:\your location\demo-spring-boot-hello
```

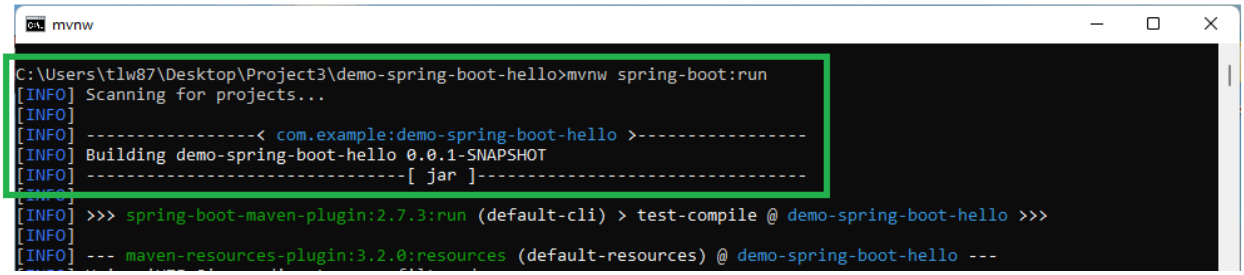


Build and run the spring boot application

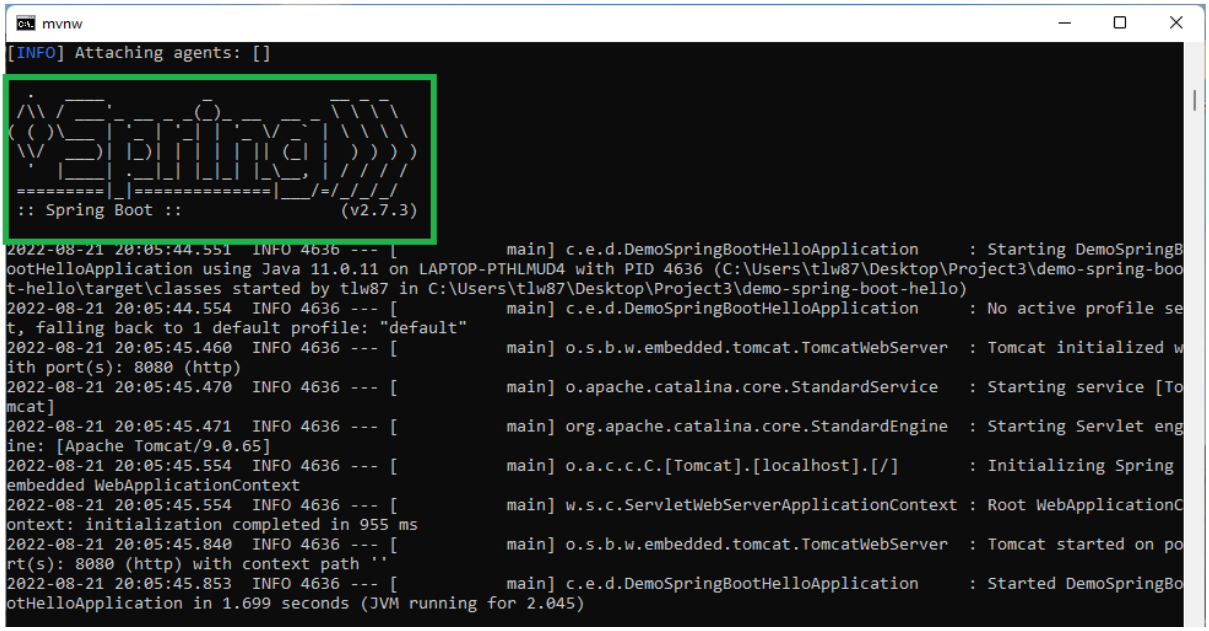
```
mvnw spring-boot:run
```



The command builds the application. It can take some time for the build to begin and show the initial status. Once started the window will scroll showing the status of the build process.



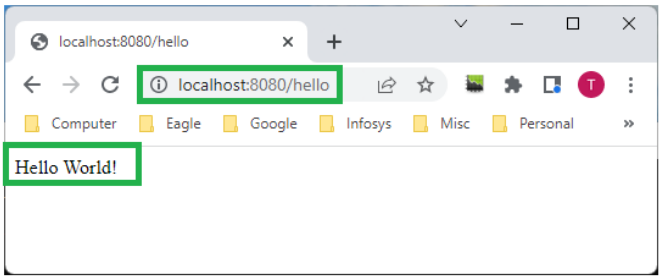
The command builds the projects Tomcat server. Once the screen stops scrolling, and Spring Boot is displayed as shown below, then the server is ready.



Test the application from a browser

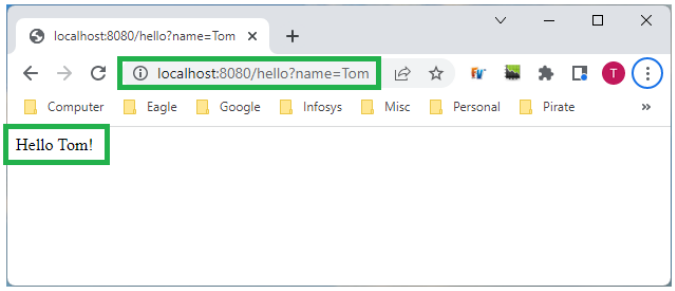
Enter the base URL in your browser.

```
http://localhost:8080/hello
```



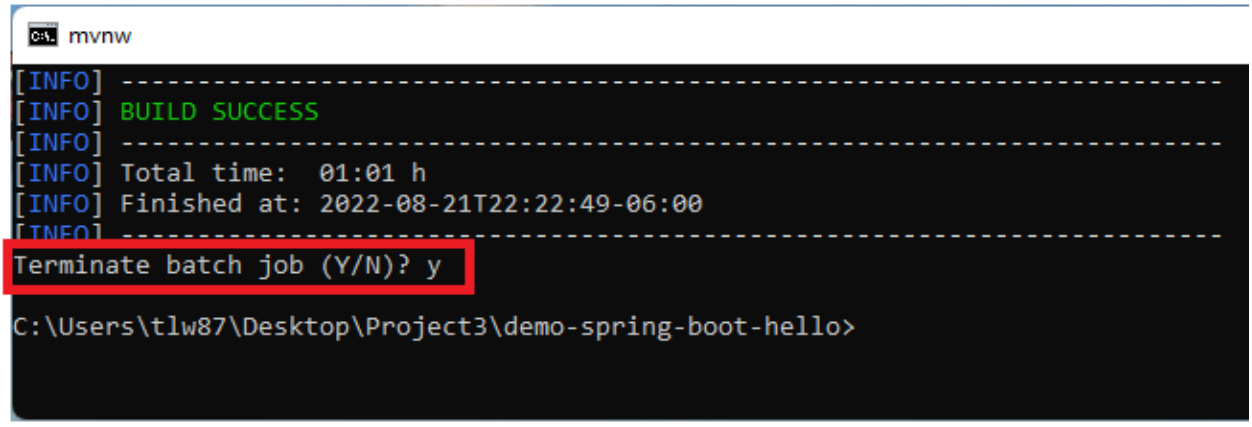
Enter a URL in your browser to test the name parameter.

```
http://localhost:8080/hello?name=Tom
```



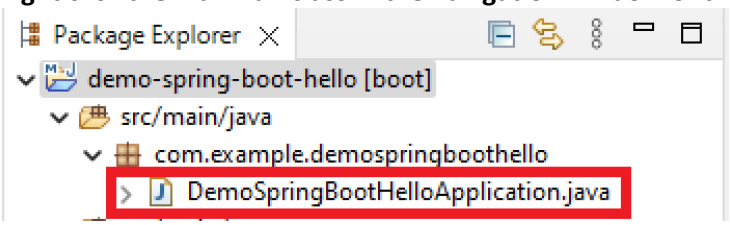
Stop the command line server

Type: “ctrl-c”
Enter “y”



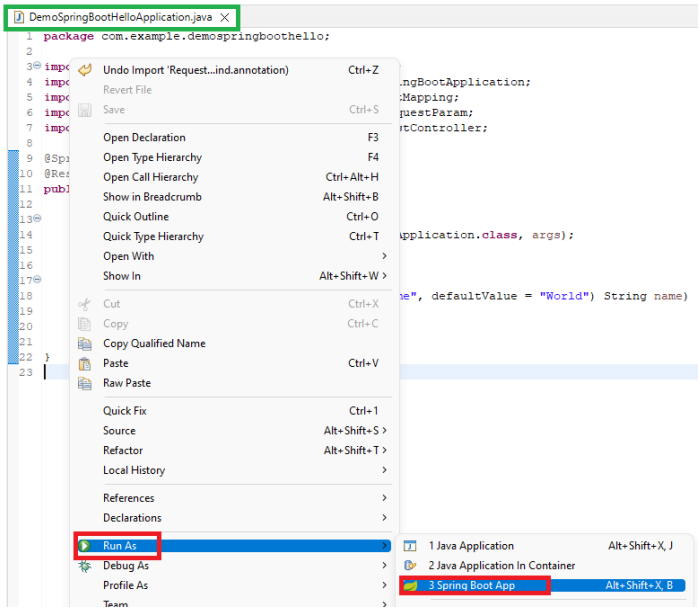
Run Spring Boot Project Within the IDE

Right click the main run class in the navigation window of the IDE

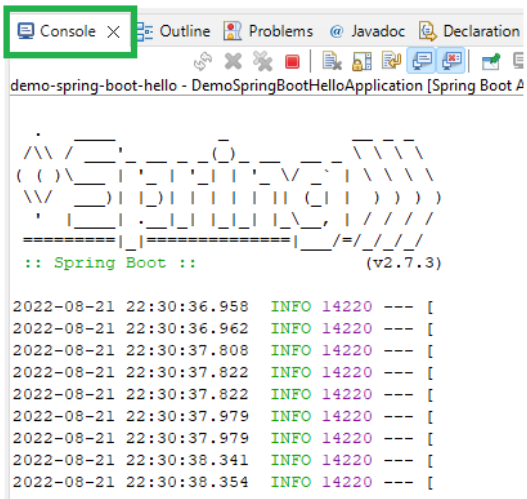


Or right click within the open main run class in the edit window of the IDE

Mouse over “Run As” ➔ select “Spring Boot App”



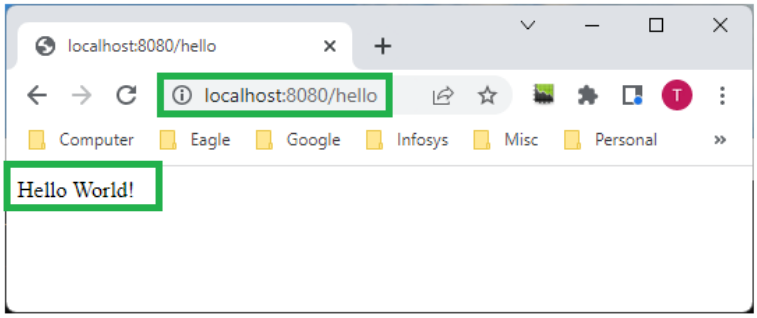
The Console tab inside the IDE shows the project’s packaged Tomcat server running for Spring Boot



Test the application from a browser

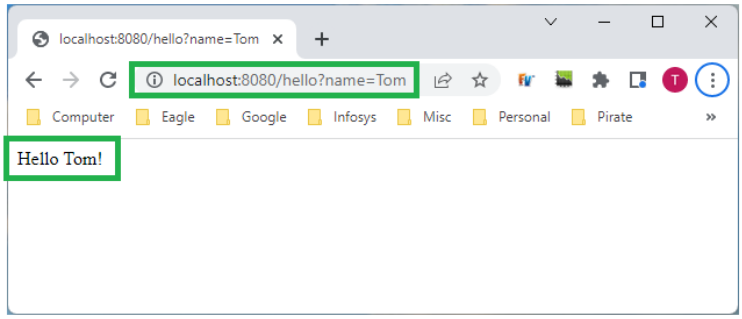
Enter the base URL in your browser.

```
http://localhost:8080/hello
```



Enter a URL in your browser to test the name parameter

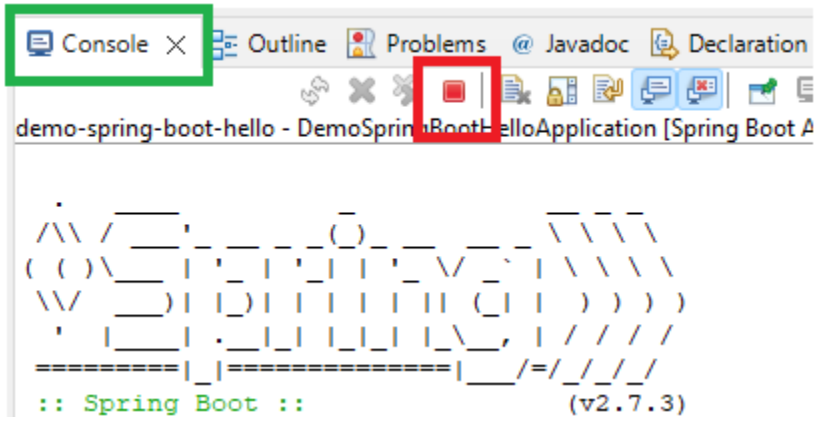
```
http://localhost:8080/hello?name=Tom
```



Stop the server

Right click in the console window and select “Terminate/Disconnect All”

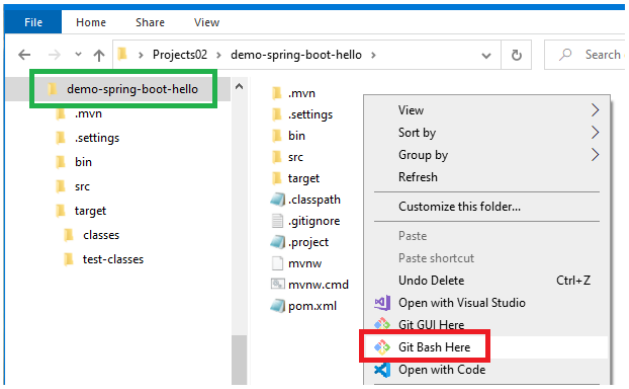
Or click the stop icon 



Executable JAR File Lifecycle

This section describes building, running, testing, and stopping an executable Java Archive (JAR) file. The executable JAR file is ran on the command and tested through a web browser. In this document we build the file using the Java command line and run the file using Maven.

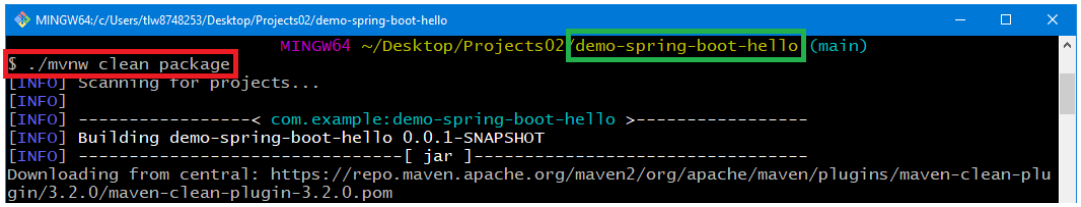
Open a Git Bash terminal in the project folder of the Spring Tool Suite 4 project where an executable JAR file is to be created.



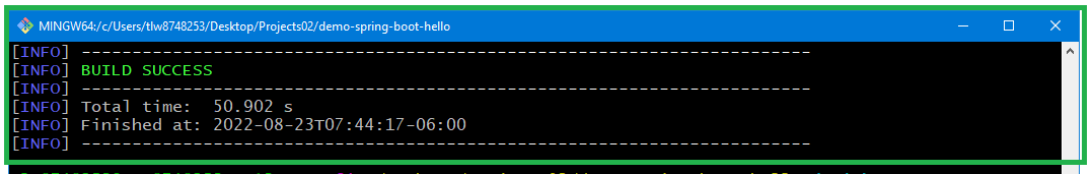
Build the Executable JAR File

On the Git Bash command line enter the following command.

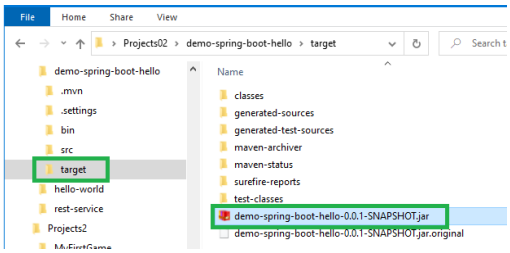
```
./mvnw clean package
```



The build process begins and the screen will scroll showing the build status.
At the end of the build process the status is displayed.



Looking inside the project's target folder the executable JARD file is found.
You can rename the file to make future commands if desired.



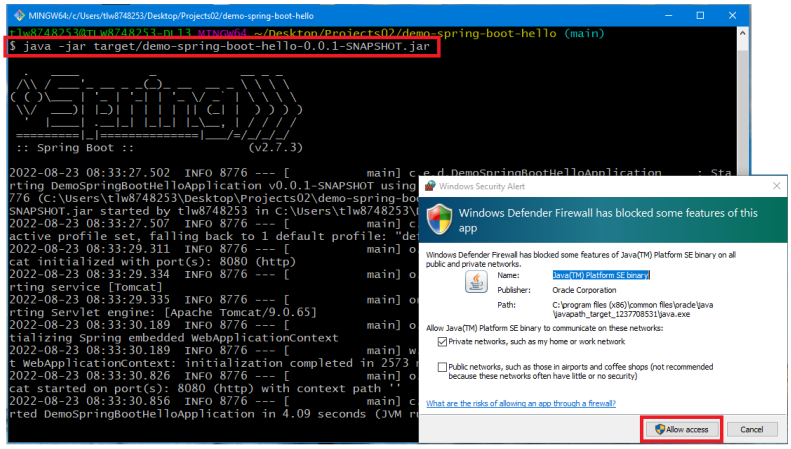
Run the Executable JAR File

Run the executable JAR by entering the following command on the Git Bash command line. In the same folder where the build command was used.

Substitute the specific Spring Tool Suite project JAR file for “demo-spring-boot-hello-0.0.1-SNAPSHOT.jar” if different.

```
java -jar target/demo-spring-boot-hello-0.0.1-SNAPSHOT.jar
```

The Git Bash terminal shows the start-up status.
Click “Allow access” if the firewall alert is seen.

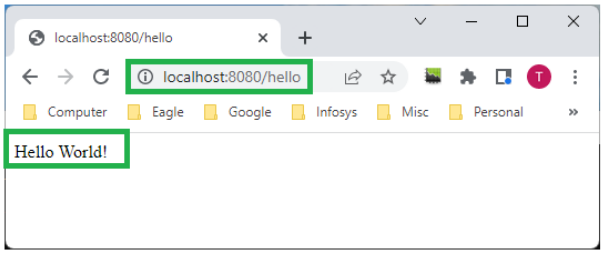


Leave the Git Hub terminal window running as long as testing is performed.

Test the Executable JAR File

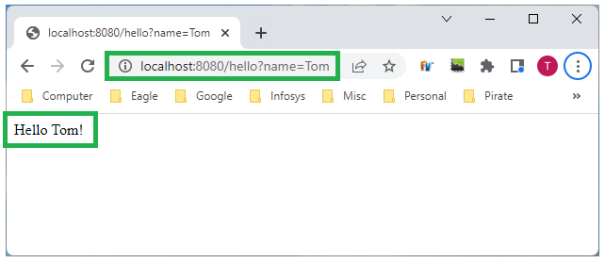
As with the other testing in this document, use a web browser and enter the URLs in this section.
Enter the base URL in your browser.

```
http://localhost:8080/hello
```



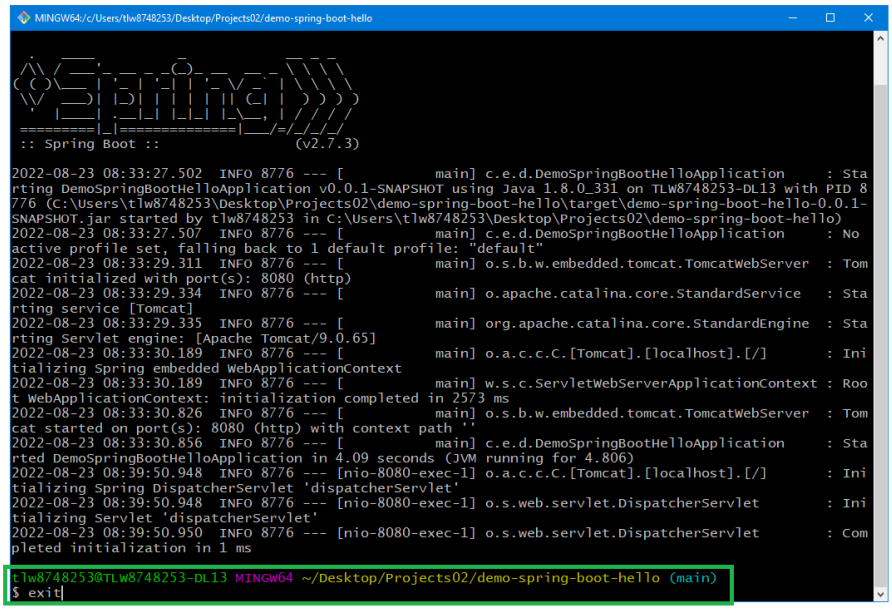
Enter a URL in your browser to test the name parameter

```
http://localhost:8080/hello?name=Tom
```



Stop the Executable JAR File

Perform a “ctr-c” command in the Git Bash terminal.
The command prompt is returned.
Use “exit” to close the terminal



Converting a Spring Boot JAR Application to a WAR

NEXT Topic
https://spring.io/guides/gs/convert-jar-to-war/
TBD.