Title: Get command-line arguments from spring-boot:run

Post Body:

Is there any way to input arguments when launching spring-boot application (mvn spring-boot:run) from commandline and then get them in main()?

Accepted Answer:

Looking at the source code of the spring-boot-maven-plugin I found that you need to do:

```
mvn spring-boot:run -Drun.arguments="arg1,arg2"
```

Another way to get more information about what options the `run` goal of the `spring-boot` plugin supports is to execute the following command:

```
mvn help:describe -Dcmd=spring-boot:run -Ddetail
```

For Spring Boot 2.x, the source is here and you now need to use `-Dspring-boot.run.arguments="args1,args2"`

(edit from april 2021) For Spring Boot 2.2+, you now need to use `-Dspring-boot.run.arguments="args1 args2"`

If you are using Gradle and you want to be able to pass command line arguments to the Gradle `bootRun` task, you first need to configure, for example like so:

```
bootRun {     if ( project.hasProperty('args') ) {         args project.args.split('\\s+')     } }
```

and run the task using `gradle bootRun -Pargs="arg1 arg2"`

Highest Rated Answer:

When passing multiple arguments using -Drun.arguments, if the argument in turn has 'comma-separated' values, then only the first value of each argument is used. To avoid this repeat the argument as many times as the number of values.

This is more of a workaround. Not sure if there is an alternative unless the delimiter is different - like '|'.

E.g Issue:

```
mvn spring-boot:run -Drun.arguments='--spring.profiles.active=test,dev'
```

Picks only 'test' profile for the above command.

Workaround:

```
mvn spring-boot:run -Drun.arguments='--spring.profiles.active=test,--spring.profiles.active=dev'
```

Picks both 'dev' & 'test' profiles for the above command.