 (/users/login.html)   (/search)

REFCARDZ (/refcardz)   **TREND REPORTS** (/trendreports)   **WEBINARS** (/webinars)   **ZONES** ⌄

How to Use Profiles in Spring Boot

by Sovan Misra (/users/2900755/sovan107.html) ⏣ MVB · Sep. 07, 18 · Java Zone (/java-jdk-development-tutorials-tools-news) · Tutorial

  Like (42)   Comment (10)   ☆ Save   Tweet

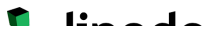Spring Boot is gaining popularity like never before, and I know it will be a persistent player in the coming months. There are some features that every technology has and it is very useful in enterprise applications. In this post, we will talk about one feature in particular: profiles.

What Are Profiles?

Every enterprise application has many environments, like:

```
Dev | Test | Stage | Prod | UAT / Pre-Prod
```

Each environment requires a setting that is specific to them. For example, in DEV, we do not need to constantly check database consistency. Whereas in TEST and STAGE, we need to. These environments host specific configurations called Profiles.

H

Th

We make properties files for each environment and set the profile in the application

We make properties files for each environment and set the profile in the application accordingly, so it will pick the respective properties file. Don't worry, we will see how to set it up.

This article will demonstrate how to setup Profiles for your Spring Boot application.

Let's start with setting up a Spring Boot application from the Spring Starter (https://start.spring.io/).

Next, we need to import the project into STS as a Maven Project. Below is the project structure:
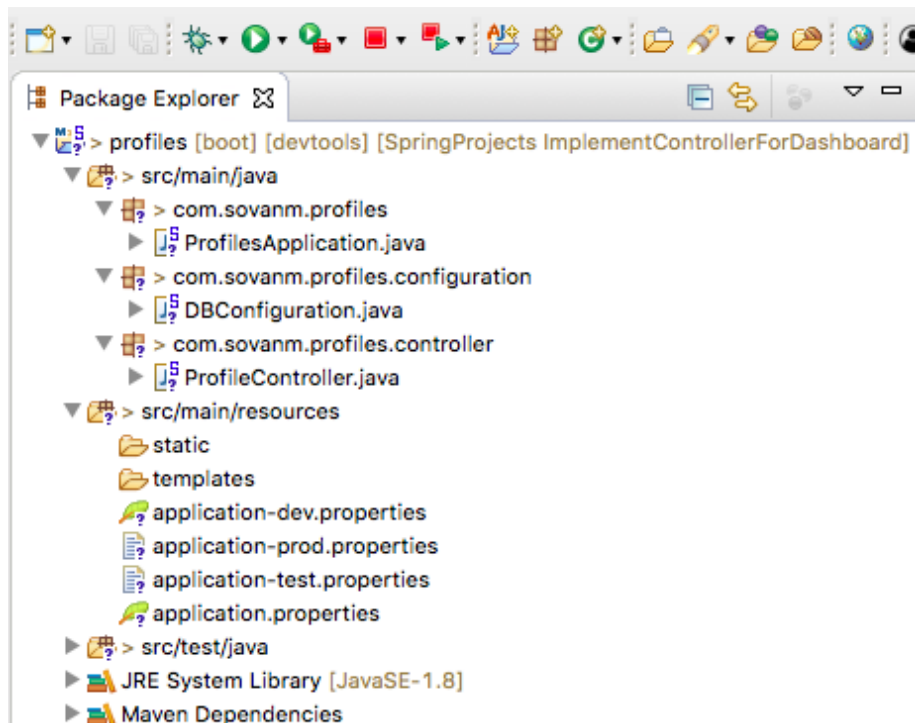
X

In this demo application, we will see how to configure different databases at runtime based on the specific environment by their respective profiles.

As the DB connection is better to be kept in a property file, it remains external to an application and can be changed. We will do so here. But, Spring Boot — by default — provides just one property file (`application.properties`). So, how will we segregate the properties based on the environment?

The solution would be to create more property files and add the "profile" name as the suffix and configure Spring Boot to pick the appropriate properties based on the profile.

Then, we need to create three `application.properties` :

1. `application-dev.properties`

2. `application-test.properties`

3. `application-prod.properties`

Of course, the `application.properties` will remain as a master properties file, but *if we override any key in the profile-specific file, the latter will gain precedence*.

I will now define DB configuration properties for in respective properties file and add code in `DBConfiguration.class` to pick the appropriate settings.

Here is the base `application.properties` :

In DEV, we will use an in-memory database:

In

w

```
1 #TEST ENVIRONEMNT SETTING#
2
3 app.message= This is the property file for the ${spring.application.name} specific to TEST Environment
4
5 spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
6 spring.datasource.url=jdbc:mysql-instance1.sovanmaws.us-east-1.rds.amazonaws.com:3306/profiles
7 spring.datasource.username=<USERNAME>
8 spring.datasource.password=<SECRET>|
```

DZone. (/)   (/users/login.html)   🔍 (/search)

REFCARDZ (/refcardz)   **TREND REPORTS** (/trendreports)   **WEBINARS** (/webinars)   **ZONES** ⌄

```
Spring Tool Suite   File  Edit  Navigate  Search  Project  Run  Window  Help            62% ⚡  Sun 8:43 AM
          Profile - profiles/src/main/resources/application-prod.properties - Spring Tool Suite

📄 application-dev.properties   📄 application-test.properties   📄 application-prod.properties ⊠   📄 application.properties

1 #PROD ENVIRONEMNT SETTING#
2
3 app.message= This is the property file for the ${spring.application.name} specific to PRODUCTION Environment
4
5 spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
6 spring.datasource.url=jdbc:mysql-instance100.sovanmaws.us-east-1.rds.amazonaws.com:3306/profiles
7 spring.datasource.username=<USERNAME_PROD>
8 spring.datasource.password=<SECRET_PASS>
```
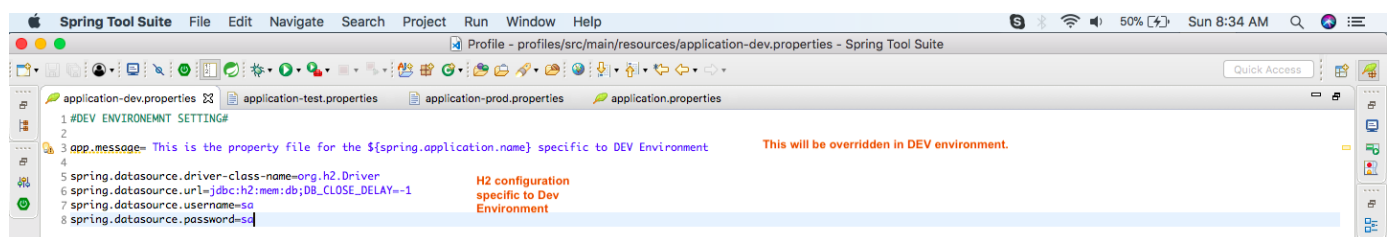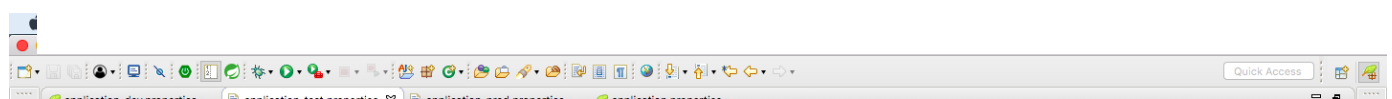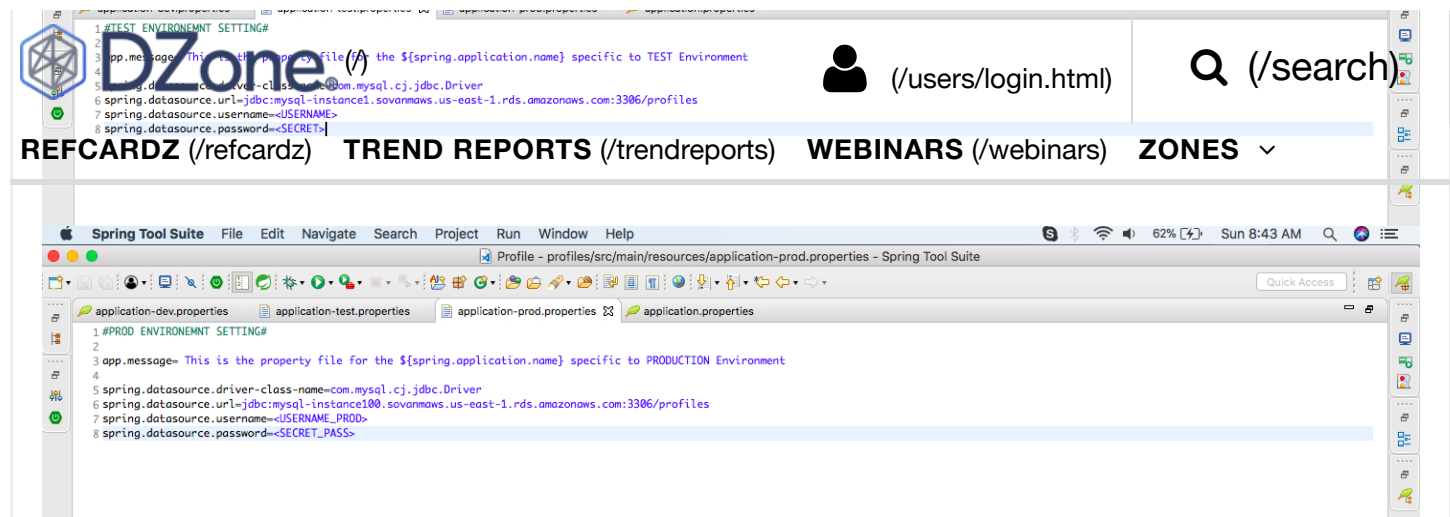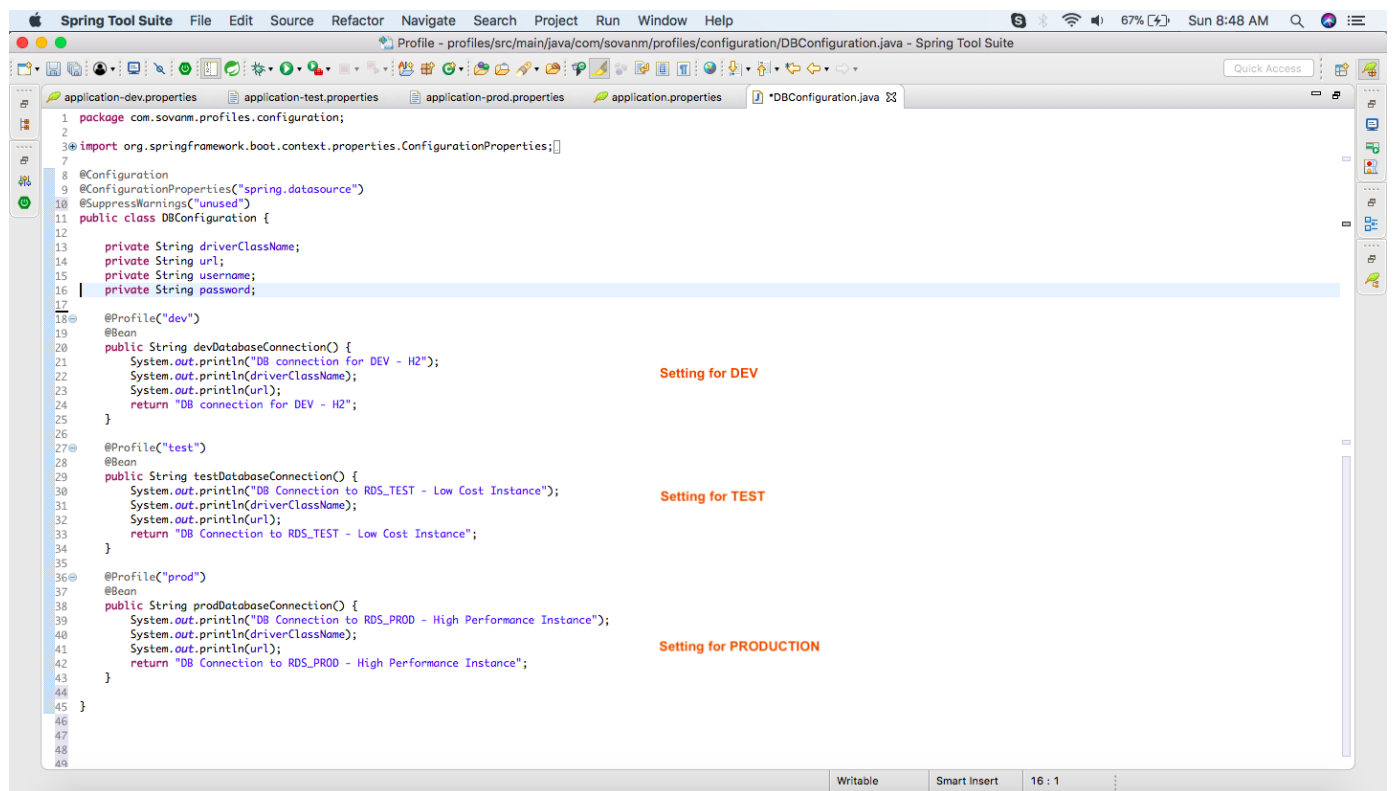
Now, we are done with properties files. Let's configure in
the `DBConfiguration.class` to pick the correct one.

```
Spring Tool Suite   File  Edit  Source  Refactor  Navigate  Search  Project  Run  Window  Help       67% ⚡  Sun 8:48 AM
          Profile - profiles/src/main/java/com/sovanm/profiles/configuration/DBConfiguration.java - Spring Tool Suite

📄 application-dev.properties   📄 application-test.properties   📄 application-prod.properties   📄 application.properties   📄 *DBConfiguration.java ⊠

 1 package com.sovanm.profiles.configuration;
 2
 3 import org.springframework.boot.context.properties.ConfigurationProperties;
 7
 8 @Configuration
 9 @ConfigurationProperties("spring.datasource")
10 @SuppressWarnings("unused")
11 public class DBConfiguration {
12
13     private String driverClassName;
14     private String url;
15     private String username;
16 |   private String password;
17
18⊖    @Profile("dev")
19     @Bean
20     public String devDatabaseConnection() {
21         System.out.println("DB connection for DEV - H2");
22         System.out.println(driverClassName);                    Setting for DEV
23         System.out.println(url);
24         return "DB connection for DEV - H2";
25     }
26
27⊖    @Profile("test")
28     @Bean
29     public String testDatabaseConnection() {
30         System.out.println("DB Connection to RDS_TEST - Low Cost Instance");
31         System.out.println(driverClassName);                    Setting for TEST
32         System.out.println(url);
33         return "DB Connection to RDS_TEST - Low Cost Instance";
34     }
35
36⊖    @Profile("prod")
37     @Bean
38     public String prodDatabaseConnection() {
39         System.out.println("DB Connection to RDS_PROD - High Performance Instance");
40         System.out.println(driverClassName);                    Setting for PRODUCTION
41         System.out.println(url);
42         return "DB Connection to RDS_PROD - High Performance Instance";
43     }
44
45 }
46
47
48
49
                                                    Writable        Smart Insert      16 : 1
```

We have used the `@Profile("Dev")` to let the system know that this is the `BEAN` that
should be picked up when we set the application profile to DEV. The other two beans will
not be created at all.

One last setting is how to let the system know that this is DEV, TEST, or PROD. But, how
do we do this?

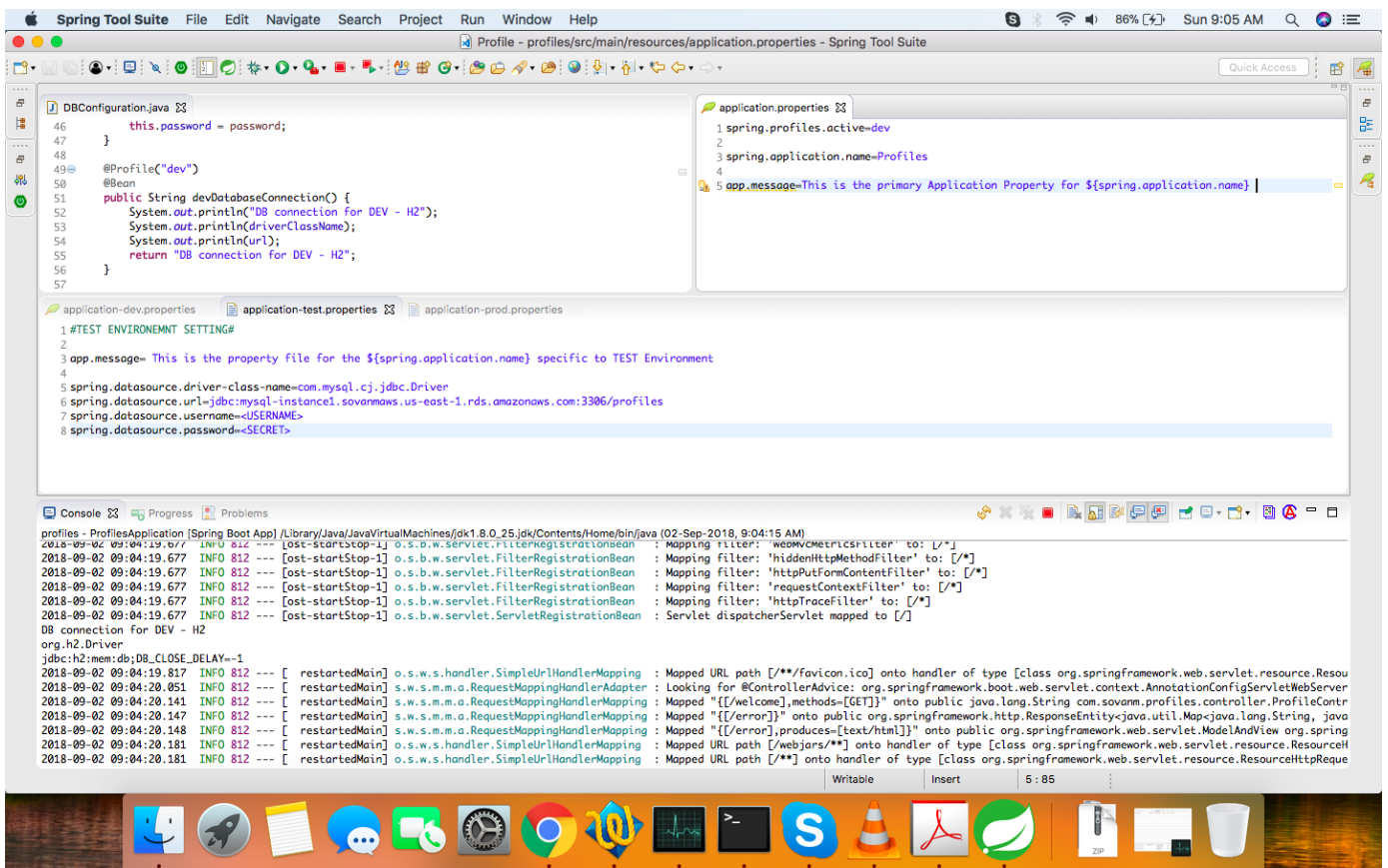We will use the `application.properties` to use the key below:

X

From here, Spring Boot will know which profile to pick. Let's run the application now!
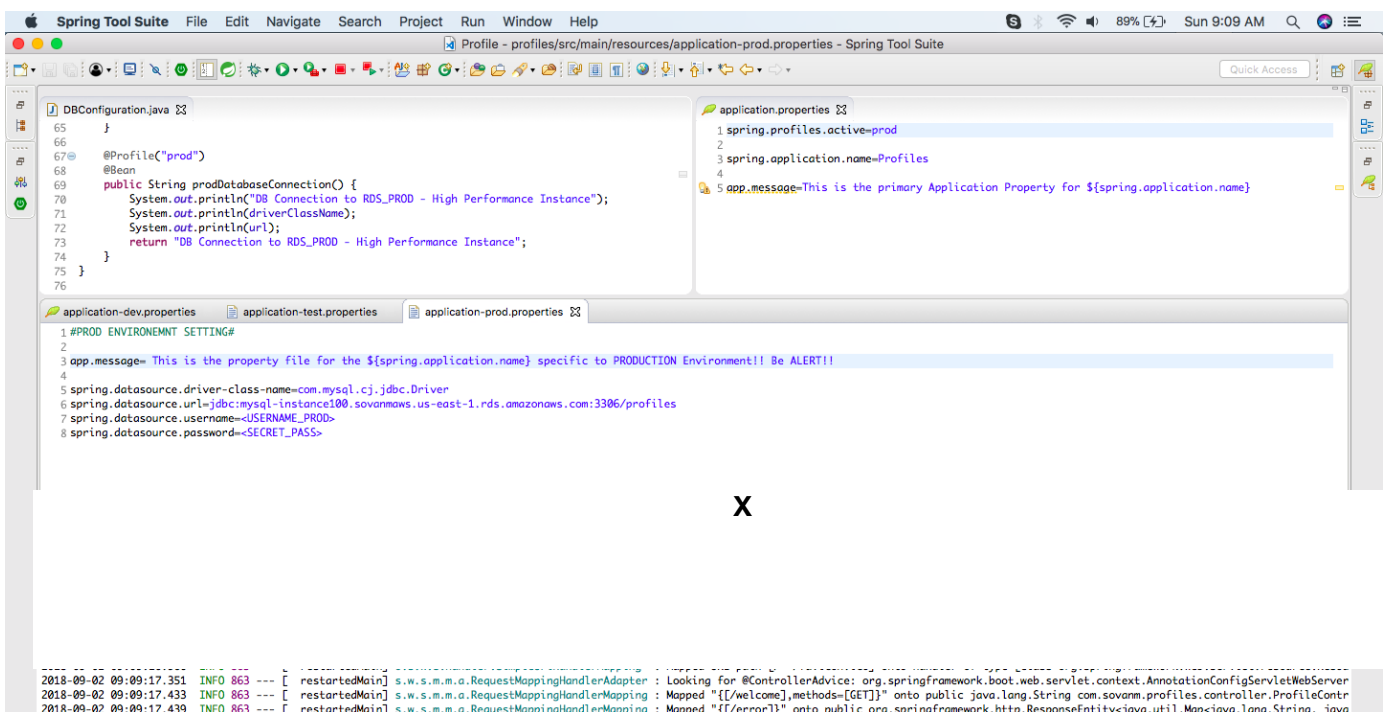
With the profile in DEV mode and it should pick H2 DB.
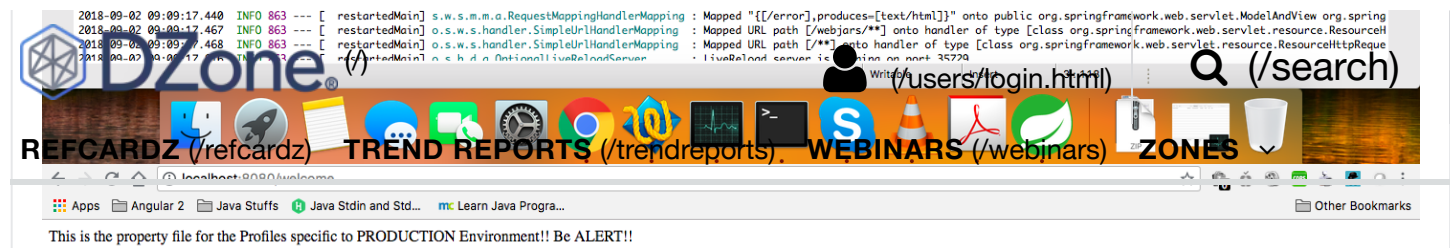

This is the property file for the Profiles specific to DEV Environment

Now, change the profile to PROD. We will see MySQL with High Config for DB. This should be picked, and the message will be overridden with the PROD message.

Apps | Angular 2 | Java Stuffs | Java Stdin and Std... | Learn Java Progra... | Other Bookmarks

This is the property file for the Profiles specific to PRODUCTION Environment!! Be ALERT!!

That's it! We just have to change it once at the `application.properties` to let Spring Boot know which environment the code is deployed in, and it will do the magic with the setting.

Please visit this repository (https://bitbucket.org/sovanm/profiles/src/master/) to access the code to learn more!

Free eBook: Try Infrastructure as Code
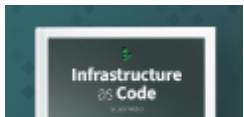
This is a step-by-step guide for you to learn how to use popular IaC tools: Terraform, Ansible, Puppet, Chef, and Salt. Each tool is covered as a stand-alone project that

---

Spring Framework   Spring Boot   Profile (Engineering)

Published at DZone with permission of Sovan Misra, DZone MVB. See the original article here. (https://learnspringswithme.wordpress.com/2018/09/02/spring-boot-profiles/) Opinions expressed by DZone contributors are their own.

Popular on DZone

- **Comparing Distributed Databases (/articles/comparing-distributed-databases?fromrel=true)**

- **Building HIPAA Compliant APIs (/articles/building-hipaa-compliant-apis?fromrel=true)**

- **Spelling "Equality" With IT: Addressing the Gender Gap in the Tech Industry (/articles/spelling-equality-with-it-addressing-the-gender-ga?fromrel=true)**

- **Writing Beautiful, Optimized, and Better .NET Code With NDepend Static Analysis (/articles/writing-beautiful-optimized-and-better-net-code-wi?fromrel=true)**

X

ABOUT US

About DZone **(/pages/about)**

Send feedback **(mailto:support@dzone.com)**

Careers **(https://careers.dzone.com/)**

Sitemap **(/sitemap)**

ADVERTISE

Advertise with DZone **(https://advertise.dzone.com)**

CONTRIBUTE ON DZONE

Article Submission Guidelines **(/articles/dzones-article-submission-guidelines)**

MVB Program **(/pages/mvb)**

Become a Contributor **(/pages/contribute)**

Visit the Writers' Zone **(/writers-zone)**

LEGAL

Terms of Service **(/pages/tos)**

Privacy Policy **(/pages/privacy)**

CONTACT US

600 Park Offices Drive

Suite 300

Durham, NC 27709

support@dzone.com **(mailto:support@dzone.com)**

+1 (919) 678-0300 **(tel:+19196780300)**

Let's be friends:

X

DZ