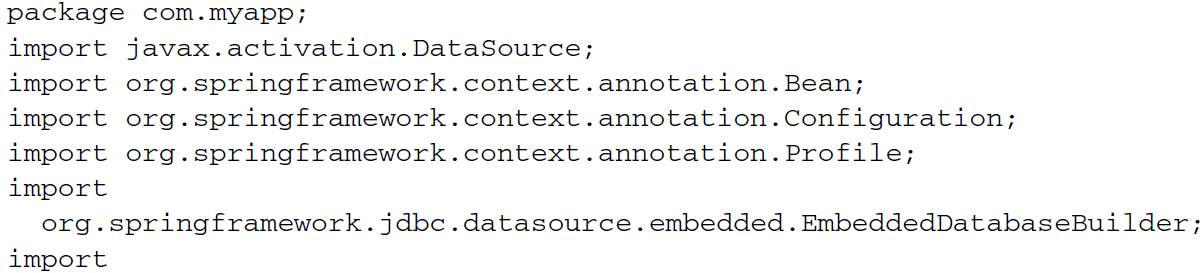
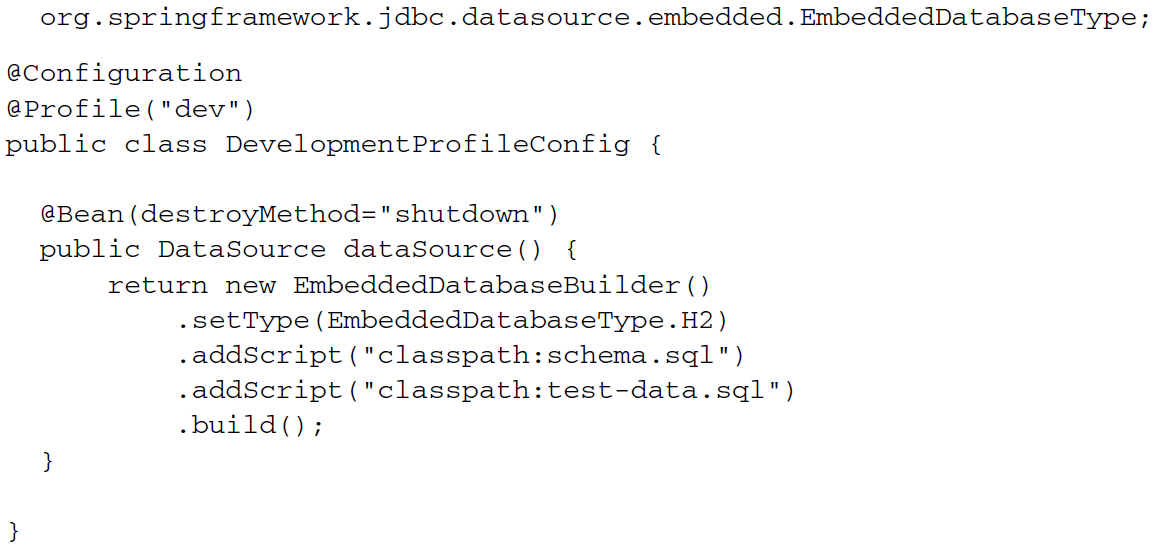
***Environments and profiles***

* One of the most challenging things about developing software is transitioning an application from one environment to another.

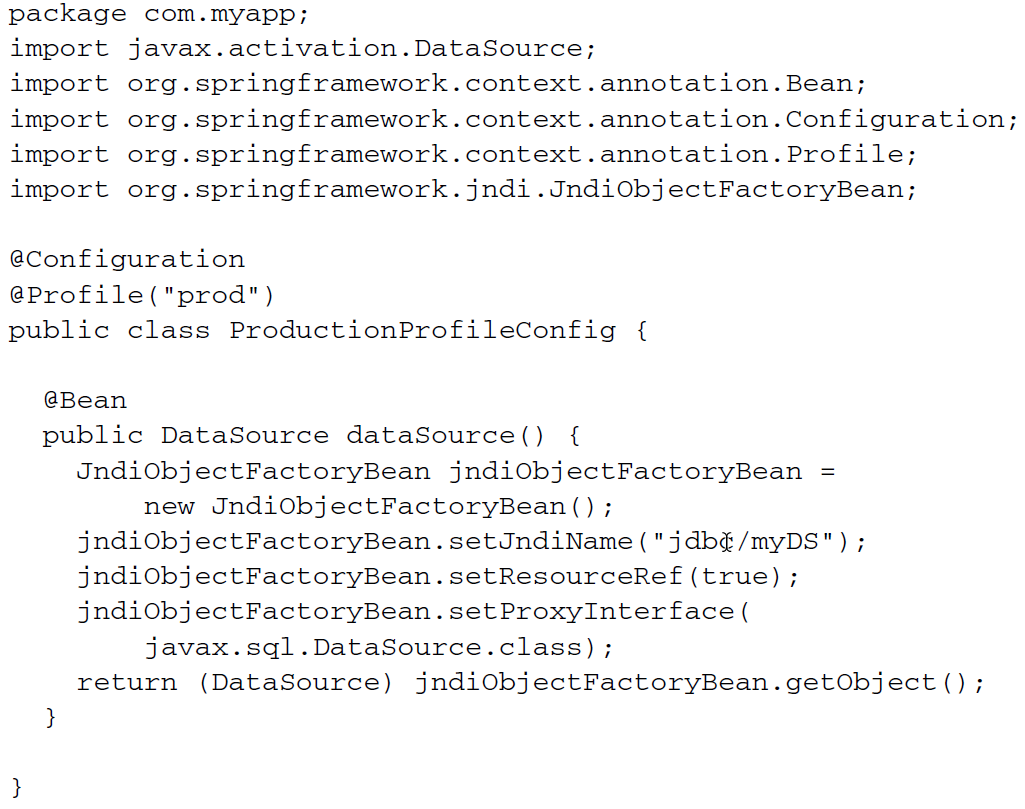
***Configuring profile beans***

* Spring’s solution for environment-specific beans isn’t much different from build-time solutions. Certainly, an environment-specific decision is made as to which beans will and won’t be created.
* But rather than make that decision at build time, Spring waits to make the decision at runtime. Consequently, the same deployment unit ( a WAR file ) will work in all environments without being rebuilt.
* In version 3.1, Spring introduced bean profiles. To use profiles, you must gather all the varying bean definitions into one or more profiles and then make sure the proper profile is active when your application is deployed in each environment.
* In Java configuration, you can use the **@*Profile*** annotation to specify which profile a bean belongs to. For example, the embedded database *DataSource* bean might be configured in a configuration class like this:



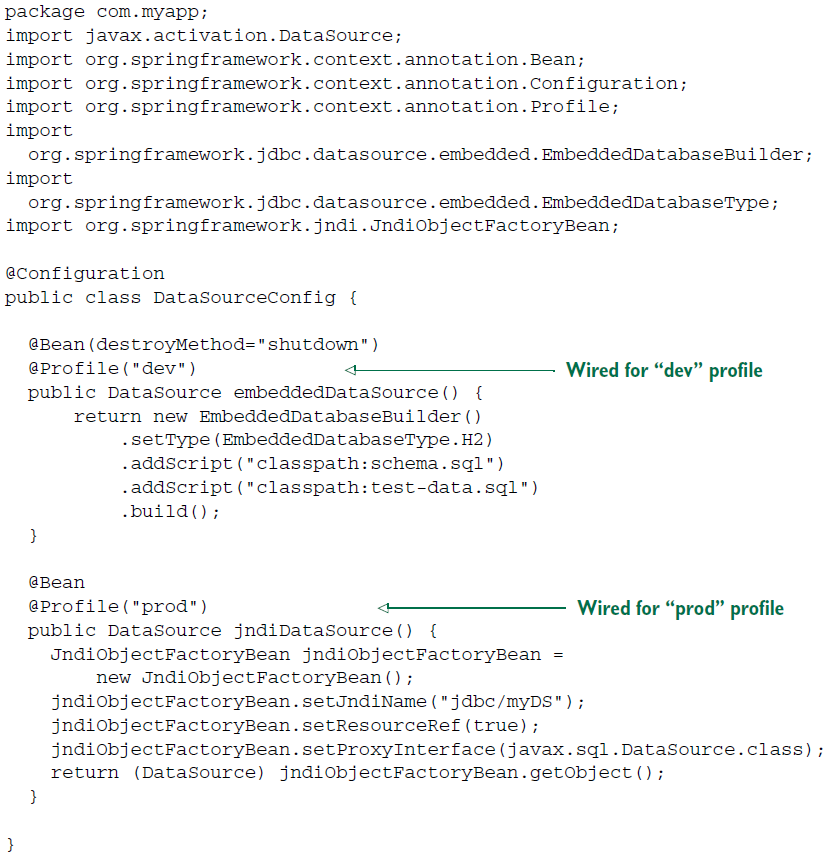


* Pay attention to the **@*Profile***annotation applied at the class level. It tells Spring that the beans in this configuration class should be created only if the *dev* profile is active. If the dev profile isn’t active, then the ***@Bean*** method will be ignored.
* Meanwhile, you may have another configuration class for production that looks like this:



* In this case, the bean won’t be created unless the *prod* profile is active.

In Spring 3.1, you could only use the **@*Profile*** annotation at the class level. Starting with Spring 3.2, however, you can use ***@Profile***  at the method level, alongside the ***@Bean*** annotation. This makes it possible to combine both bean declaration into a single configuration class, as shown in the following listing.



* What’s not apparent here that although each of the *DataSource* beans is a profile and will only be created if the prescribed profile is active, there are probably other beans that aren’t defined in the scope of a given profile. Any bean that isn’t given a profile will always be created, regardless of what profile is active.