Spring boot application.properties

In a `application.properties` file you can use place holders to reference environment variables rather than hard coding the value. For example: Instead of writing your `application.properties` like so:

```
// application.properties
aws.region=east
```

Which is fine but then if you have sensitive data which you wouldn't want to commit to the GitHub repositories you would need to use place holders and store those secrets in your environment variables which Spring Boot can then substitute in when is parsing the .properties files.

```
// application.properties
db.secrets=${DB_SECRETS}
```

If you write this and store your `DB_SECRETS` as an environment variable, it will be substituted in when the Spring boot application is ran, then you can access it using `@Value` annotation in your application.

This is much safer compared to hard coding it into your .properties file.

> This is not a feature of Intellij IDEA, it is a native functionality by Spring Boot.

Extra note

The placeholders within application.properties can refer to either of the following

1. System property
2. Environment variable
3. Another application.properties variable

It is very flexible.

In addition, if somehow for some reason all three of the same variable are assigned under system property, environment variable, and also another application.properties variable exist and you try to use it in another variable, the precedence that which of the three will be used are listed above.

For example, if you are using @Value("var") in your Spring-boot application and there is

```
-Dvar=system
env of var=environment
application.properties variable of var=appvar
```

Then "system" will be used, and if System property doesn't exist, then "environment" will be used, then finally "appvar" will be used last if the environment variable doesn't exist either.

Revision #3
Created 2023-03-25 23:42:20 UTC by Tamarine
Updated 2023-04-08 01:27:41 UTC by Tamarine