



## O que aprendemos?

Nesta aula, aprendemos que:

- Para adicionar o **Spring Boot Actuator** no projeto, devemos adicioná-lo como uma dependência no arquivo **pom.xml**;
- Para acessar as informações disponibilizadas pelo **Actuator**, devemos entrar no endereço <http://localhost:8080/actuator> (<http://localhost:8080/actuator>);
- Para liberar acesso ao **Actuator** no **Spring Security**, devemos chamar o método `.antMatchers(HttpMethod.GET, "/actuator/**")` ;
- Para que o **Actuator** exponha mais informações sobre a API, devemos adicionar as propriedades `management.endpoint.health.show-details=always` e `management.endpoints.web.exposure.include=*` no arquivo **application.properties**;
- Para utilizar o **Spring Boot Admin**, devemos criar um projeto **Spring Boot** e adicionar nele os módulos `spring-boot-starter-web` e `spring-boot-admin-server` ;
- Para trocar a porta na qual o servidor do **Spring Boot Admin** rodará, devemos adicionar a propriedade `server.port=8081` no arquivo **application.properties**;
- Para o **Spring Boot Admin** conseguir monitorar a nossa API, devemos adicionar no projeto da API o módulo `spring-boot-admin-client` e também adicionar a propriedade `spring.boot.admin.client.url=http://localhost:8081` no arquivo **application.properties**;
- Para acessar a interface gráfica do **Spring Boot Admin**, devemos entrar no endereço <http://localhost:8081> (<http://localhost:8081>).