

Desarrollo con Spring Boot

Curso práctico de 3 días - 21h

Ref.: SGT - Precio 2024: 1 300€ sin IVA

Este curso le permitirá desarrollar aplicaciones Spring Boot con un mínimo esfuerzo de configuración. Aprenderás a desplegar estas aplicaciones de forma tradicional o como aplicaciones autónomas embebidas en una imagen docker. Aprenderás sobre Spring Cloud y arquitecturas de microservicios.

OBJETIVOS PEDAGÓGICOS

Al término de la formación, el alumno podrá:

Inicio de un proyecto Spring Boot

Dominar la configuración automática de Spring Boot

Adaptación de la configuración a las necesidades específicas del proyecto

Simplificar el desarrollo de pruebas

Empaquetado y despliegue de una aplicación Spring Boot en diferentes contextos

PROGRAMA

última actualización: 08/2024

1) Recordatorio de la configuración clásica de Primavera

- Configuración mediante archivos XML, anotaciones o clases de configuración.
- Establecer parámetros de configuración, utilizando PropertyPlaceholders.
- Configuración de la capa de persistencia, DataSource y EntityManagerFactory.
- Configuración de la capa de negocio y transacciones.
- Configuración de la capa web, Controller y RestController, Spring Security.
- Pruebas unitarias y de integración: SpringTest, MockMvc.

Trabajo práctico : Aproximación de una aplicación Spring clásica que se convertirá en una aplicación Spring Boot en los siguientes capítulos.

2) Inicialización de una aplicación Spring Boot

- El ecosistema Spring y la complejidad de su configuración.
- Principio de "convención por encima de la configuración".
- Arrancadores y autoconfiguración.
- Implementación de Spring Initializer, creación de una aplicación web sencilla.

Trabajo práctico : Identificación de los arrancadores correspondientes a los requisitos, producción del esqueleto de la aplicación Spring Boot en ciernes.

3) Gestión de dependencias y autoconfiguración

- Organización de las dependencias de Maven, gestión transparente de versiones.
- Estructuración del código y buenas prácticas.
- Papel de las anotaciones @SpringBootApplication y @EnableAutoconfiguration, la clase "aplicación".
- Anatomía de una clase de autoconfiguración, papel de las anotaciones @Conditional.
- Control de la autoconfiguración, uso de los archivos de propiedades application.properties y application.yml.

PARTICIPANTES

Desarrolladores Java, gestores de proyectos Java/Java EE.

REQUISITOS PREVIOS

Buenos conocimientos de Java/Java EE. Experiencia operativa en Spring Core.

COMPETENCIAS DEL FORMADOR

Los expertos que imparten la formación son especialistas en las materias tratadas. Han sido validados por nuestros equipos pedagógicos, tanto en el plano de los conocimientos profesionales como en el de la pedagogía, para cada curso que imparten. Cuentan al menos con entre cinco y diez años de experiencia en su área y ocupan o han ocupado puestos de responsabilidad en empresas.

MODALIDADES DE EVALUACIÓN

El formador evalúa los progresos pedagógicos del participante a lo largo de toda la formación mediante preguntas de opción múltiple, escenificaciones de situaciones, trabajos prácticos, etc. El participante también completará una prueba de posicionamiento previo y posterior para validar las competencias adquiridas.

MEDIOS PEDAGÓGICOS Y TÉCNICOS

- Los medios pedagógicos y los métodos de enseñanza utilizados son principalmente: ayudas audiovisuales, documentación y soporte de cursos, ejercicios prácticos de aplicación y ejercicios corregidos para los cursillos prácticos, estudios de casos o presentación de casos reales para los seminarios de formación.
- Al final de cada cursillo o seminario, ORSYS facilita a los participantes un cuestionario de evaluación del curso que analizarán luego nuestros equipos pedagógicos.
- Al final de la formación se entrega una hoja de presencia por cada media jornada de presencia, así como un certificado de fin de formación si el alumno ha asistido a la totalidad de la sesión.

MODALIDADES Y PLAZOS DE ACCESO

La inscripción debe estar finalizada 24 horas antes del inicio de la formación.

ACCESIBILIDAD DE LAS PERSONAS CON DISCAPACIDAD

¿Tiene alguna necesidad específica de accesibilidad? Póngase en contacto con la Sra. FOSSE, interlocutora sobre discapacidad, en la siguiente dirección psh-accueil@orsys.fr para estudiar de la mejor forma posible su solicitud y su viabilidad.

- Crea tus propias clases de autoconfiguración.

Trabajo práctico : Reorganización de los paquetes de aplicaciones existentes, creación de una clase "aplicación", control de la configuración automática a las necesidades de la aplicación existente.

4) Diseño de una aplicación Spring Boot

- Definición de perfiles.
- Configuración de logs: log4j2 frente a logback.
- JPA y Spring Data JPA con Spring Boot.
- Spring MVC con Spring Boot.
- Configuración de infraestructuras técnicas integradas (base de datos, contenedor de servlets).
- Configuración de la seguridad en las API REST.
- Facilitar el desarrollo utilizando Spring Boot DevTools.

Trabajo práctico : Implantación de infraestructuras técnicas integradas, maximizando la autoconfiguración.

5) Probar una aplicación Spring Boot

- Autoconfiguración de las pruebas.
- Pruebe la persistencia con @DataJpaTest.
- Prueba de integración con @SpringBootTest.
- Pruebas de API REST con @AutoConfigureMockMvc, @WebMvcTest y RestTemplate.
- Monitorización de una aplicación Spring Boot con Actuator.

Trabajo práctico : Actualización de las pruebas unitarias y de integración de la aplicación a las nuevas prácticas de Spring Boot, supervisando el uso de puntos finales REST.

6) Empaquetado y despliegue de una aplicación Spring Boot

- Creación de una guerra para el despliegue clásico.
- Creación de un tarro ejecutable "autónomo".
- Despliegue en una imagen docker.

Trabajo práctico : Produce un jar autoejecutable y desplégalo en una imagen docker.

7) Introducción a Spring Cloud y arquitecturas de microservicios

- Arquitectura distribuida y microservicios, patrones de arquitectura de microservicios.
- Una visión general de Spring Cloud.
- Externalice la gestión de configuraciones y versiones con Spring Cloud Config.
- Registre y localice dinámicamente servicios con Spring Cloud Netflix.
- Implementación del equilibrio de carga con Spring Cloud LoadBalancer.
- Creación de una pasarela de API con ZUUL y protección del acceso con Spring Security.
- Última línea eliminada

Trabajo práctico : Ingeniería inversa de una aplicación monolítica en una arquitectura de microservicios.

FECHAS

Contacto