



ALURA BOOST

Java y Docker

Temário 2024

alura
BOOST

Módulo 1

Premisas

- 5 semanas de duración.
- Estudio autoguiado + proyecto práctico.
- Mentorías en vivo todas las semanas.
- Carga horaria: 15 a 18 horas/semana de dedicación exclusiva al programa.

Módulo 1

Temario

Punto de partida

El participante debe tener conocimientos previos en:

- Fundamentos de Java: Orientación a Objetos, Collections, Excepciones, Generics.
- Java Funcional y principales APIs: consumo de APIs REST, manipulación de archivos, lambdas y streams, fechas.
- Persistencia en Java: mapeo de BDs relacionales, Spring Data JPA.
- Spring Boot y APIs REST: Controllers, DTOs, CORS, Spring Security y JWT, Documentación con OpenAPI, nociones de pruebas de integración.
- Conocimientos básicos de redes de computadoras.

Punto de llegada

El participante habrá adquirido conocimientos y experiencia en:

- Refactorización: mejora del código de manera sistemática.
- Principios SOLID y Separación de responsabilidades: diseño de código para mantenibilidad.
- Patrones de diseño: utilizando los patrones Command, Strategy y Chain of Responsibility.
- Pruebas automatizadas: pruebas unitarias e integración con JUnit, Mockito y Spring Boot.
- Docker: imágenes, contenedores, volúmenes, redes, Docker Compose.

Jornada Upskilling Técnico

Semana 1

- 10 horas de carga horaria para la conclusión del curso.
- 4 horas de carga horaria estimada para proyectos prácticos.
- Mentoría en vivo.

Java y Refactorización: Mejorando Códigos con Buenas Prácticas

- Refactora código complejo con buenas prácticas de programación.
- Estructura métodos y clases de manera eficaz.
- Aprende a utilizar el patrón de diseño Command.
- Conoce cómo utilizar el pattern matching de switch en Java 15.
- Realiza pruebas unitarias con JUnit y Mockito.

Semana 2

- 10 horas de carga horaria para la conclusión del curso.
- 4 h de carga horaria estimada para proyectos prácticos.
- Mentoría en vivo.
- Primera entrega parcial del Proyecto.

Buenas Prácticas de Programación: Mejora el Código de una API Java

- Identifica prácticas incorrectas en el código de una API Java.
- Separa correctamente las responsabilidades del proyecto.
- Utiliza técnicas de refactorización de código.
- Aplica patrones de diseño en el código del proyecto.
- Optimiza consultas a la base de datos.

Semana 3

- 10 horas de carga horaria para la conclusión del curso.
- 4 horas de carga horaria estimada para proyectos prácticos.
- Mentoría en vivo.

Buenas Prácticas de Programación: Automatizando Pruebas con Java

- Entiende la importancia de las pruebas automatizadas en el código.
- Utiliza JUnit como biblioteca para pruebas automatizadas.
- Escribe pruebas unitarias con JUnit.
- Simula comportamientos en las pruebas con la biblioteca Mockito.
- Aprende a probar clases de servicio y controladores en Spring.

Semana 4

- 10 horas de carga horaria para la conclusión del curso.
- 4 horas de carga horaria estimada para proyectos prácticos.
- Mentoría en vivo.

Docker: Creación y Gestión de Contenedores

- Levanta un contenedor Docker.
- Crea y personaliza imágenes.
- Comprende los conceptos de persistencia a través de volúmenes.
- Utiliza redes para comunicar tus contenedores.
- Coordina tus contenedores con Docker Compose.

Semana 5

- 12 horas de carga horaria estimada para proyectos prácticos.
- Mentoría en vivo.
- Entrega final del Proyecto.

Proyecto a Desarrollar: Comex

| Visión del producto

El proyecto **Comex** es un sistema de comercio electrónico que funciona como un marketplace, es decir, una plataforma donde varios vendedores pueden ofrecer sus productos a una base de clientes. El objetivo principal es convertirse en el "shopping online" más popular, brindando a los usuarios acceso a una amplia variedad de productos y vendedores en un solo lugar.