Description: Description: 5300_IBMpos

Developing Cloud Native Applications with IBM Liberty

WA610 (Classroom)

Course description

This course teaches you how to develop a microservice application by using IBM Liberty. Liberty is a lightweight Java™ runtime that is built by using modular features. It is available as both open source and commercial offerings.

In this course, you learn how to use Liberty, Jakarta EE, and MicroProfile to build a RESTful microservice application. MicroProfile is an open-source specification that defines new standards and APIs to accelerate and simplify the creation of microservices. You also explore other features of Liberty and developing microservices applications.

In the hands-on exercises, you build a microservice application, containerize it, and then deploy it on Kubernetes.

For information about other related courses, see the IBM Training website:

**ibm.com**/training

General information

Delivery method

Classroom (ILT)

Course level

ERC 1.0

Product and version

IBM Liberty 2023

Audience

The primary audience for this course is the Application Developer.

Learning objectives

After completing this course, you should be able to:

* Describe Liberty architecture
* Build a microservice application with Liberty and REST
* Use OpenAPI to document APIs
* Use Jakarta Persistence API (JPA) to persist data
* Secure and test a RESTful API
* Use MicroProfile to add health checks and monitoring capabilities to an application
* Build and test an application in a container
* Deploy a microservice application to Kubernetes

Prerequisites

Before taking this course, you should have:

* Experience using the Java programming language
* Working knowledge of the Linux operating system
* Familiarity with Kubernetes

Duration

2 days

Skill level

Intermediate

Notes

The following unit and exercise durations are estimates and might not reflect every class experience. If the course is customized or abbreviated, the duration of unchanged units might increase.

Course Agenda

|  |
| --- |
| Course Introduction  Duration: 20 minutes |

|  |  |
| --- | --- |
| Unit 1. IBM Liberty Overview  Duration: 50 minutes | |
| Overview | This unit describes the features and architecture of IBM Liberty. |
| Learning objectives | After completing this unit, you should be able to:   * Describe the difference between Liberty editions * Describe the Liberty runtime architecture * Describe Liberty server installation and configuration * Describe Liberty dev mode |

|  |  |
| --- | --- |
| Unit 2. Introduction to Microservices, REST and MicroProfile  Duration: 40 minutes | |
| Overview | This unit introduces Microservices architecture, REST and MicroProfile. |
| Learning objectives | After completing this unit, you should be able to:   * Describe the microservices architecture * Define what REST is * Describe the MicroProfile specification * Describe what OpenAPI is |

|  |  |
| --- | --- |
| Exercise 1. Build an application with Liberty  Duration: 45 minutes | |
| Overview | This exercise provides a hands-on introduction to Liberty, Jakarta EE, and  REST. |
| Learning objectives | After completing this exercise, you should be able to:   * Build a basic Liberty application * Start and stop the Liberty server * Update the Liberty server configuration * Develop a RESTful microservice * Deploy and run the application on Liberty |

|  |  |
| --- | --- |
| Exercise 2. Document APIs with OpenAPI  Duration: 25 minutes | |
| Overview | In this exercise, you generate an OpenAPI document and augment existing annotations with OpenAPI annotations. |
| Learning objectives | After completing this exercise, you should be able to:   * Generate an OpenAPI document for the application * Augment existing Jakarta RESTful Web Services annotations with OpenAPI annotations * Augment POJOs with OpenAPI annotations |

|  |  |
| --- | --- |
| Exercise 3. Configure the microservice  Duration: 20 minutes | |
| Overview | In this exercise, you configure the microservice ports and context root. |
| Learning objectives | After completing this exercise, you should be able to:   * Configure ports and context root * Inject static configuration * Add the MicroProfile configuration properties file |

|  |  |
| --- | --- |
| Unit 3. Working with persistent data  Duration: 20 minutes | |
| Overview | This unit describes how to persist system data by using the Jakarta Persistence API (JPA). Jakarta Persistence defines a standard for management of persistence and object/relational mapping in Java environments. |
| Learning objectives | After completing this unit, you should be able to:   * Describe what JPA is * Describe how to configure JPA with Liberty |

|  |  |
| --- | --- |
| Exercise 4. Persist system data by using the Jakarta Persistence API (JPA)  Duration: 1 hour | |
| Overview | This exercise provides hands-on experience working with persistent data. |
| Learning objectives | After completing this exercise, you should be able to:   * Define a JPA entity class * Do create, retrieve, update, and delete operations by using JPA * Configure JPA * Start a PostgreSQL database * Deploy and run an application that uses persistent data |

|  |  |
| --- | --- |
| Unit 4. Securing RESTful APIs  Duration: 20 minutes | |
| Overview | This unit describes how to secure RESTful APIs. |
| Learning objectives | After completing this unit, you should be able to:   * Describe how to secure RESTful APIs in Liberty * Add users and groups to the Liberty server configuration * Describe how to use JSON Web Tokens (JWT) for Single Sign On (SSO) |

|  |  |
| --- | --- |
| Exercise 5. Secure and test RESTful APIs  Duration: 15 minutes | |
| Overview | This exercise provides hands-on experience with testing secured RESTful APIs. |
| Learning objectives | After completing this exercise, you should be able to:   * Secure RESTful APIs * Test the secured RESTful APIs |

|  |  |
| --- | --- |
| Exercise 6. Write a RESTful client interface  Duration: 25 minutes | |
| Overview | In this exercise, you write a RESTful client interface, and then implement JSON Web Tokens (JWT) and configure them as Single Sign On (SSO) cookies to use the RESTful APIs. |
| Learning objectives | After completing this exercise, you should be able to:   * Write a RESTful client interface * Implement the client endpoint * Configure the JSON Web Token (JWT) * Start a PostgreSQL database * Run the client endpoint |

|  |  |
| --- | --- |
| Unit 5. MicroProfile health checks and monitoring  Duration: 20 minutes | |
| Overview | This unit describes the usage and features of MicroProfile Health and MicroProfile Metrics. |
| Learning objectives | After completing this unit, you should be able to:   * Describe how to use MicroProfile Health * Describe how to use MicroProfile Metrics |

|  |  |
| --- | --- |
| Exercise 7. Add health checks and metrics to an application  Duration: 20 minutes | |
| Overview | This exercise demonstrates how to add health checks to an application and provide monitoring capabilities. |
| Learning objectives | After completing this exercise, you should be able to:   * Add health checks to an application * Provide monitoring capabilities to an application |

|  |  |
| --- | --- |
| Unit 6. Working with containers  Duration: 20 minutes | |
| Overview | This unit describes how to run Liberty in a container. |
| Learning objectives | After completing this unit, you should be able to:   * Describe how to create container application images * Describe how to use a preconfigured Liberty container image |

|  |  |
| --- | --- |
| Exercise 8. Build and test an application in a container  Duration: 50 minutes | |
| Overview | This exercise provides hands-on experience working with containers. |
| Learning objectives | After completing this exercise, you should be able to:   * Build the application in a container * Create a container image * Test a microservice application that is running in a container |

|  |  |
| --- | --- |
| Unit 7. Working with Kubernetes  Duration: 20 minutes | |
| Overview | This unit describes how to deploy Liberty applications to Kubernetes clusters. |
| Learning objectives | After completing this unit, you should be able to:   * Describe Kubernetes and container orchestration * Describe how to use the Open Liberty Operator * Describe how to deploy a Liberty application to a Kubernetes cluster |

|  |  |
| --- | --- |
| Exercise 9. Deploy a microservice to Kubernetes  Duration: 50 minutes | |
| Overview | This exercise demonstrates how to deploy a Liberty application to a Kubernetes cluster. |
| Learning objectives | After completing this exercise, you should be able to:   * Use the Open Liberty Operator to create a Liberty application * Deploy an application to Kubernetes * Customize the deployment |

For more information

To learn more about this course and other related offerings, and to schedule training, see **ibm.com**/training

To learn more about validating your technical skills with IBM certification, see **ibm.com**/certify

To stay informed about IBM training, see the following sites:

IBM Training News: https://www.ibm.com/blogs/ibm-training

YouTube: youtube.com/IBMSupportTV

Facebook: facebook.com/groups/IBMTrainingandSkills

Twitter: twitter.com/ibm