



Up-to-date Practice Test with Latest Questions and Answers covering latest syllabus and topics of the exam. Makes you ready to face actual exam.



- C1000-155 Practice Questions
- C1000-155 Practice Test
- C1000-155 Practice Exam
- C1000-155 Exam Questions
- C1000-155 Study Guide



killexams.com

IBM

C1000-155

IBM Liberty 2023 Cloud Native Java Developer

ORDER FULL VERSION

<https://killexams.com/pass4sure/exam-detail/C1000-155>



Question: 1

Which of the following is an example of an authentication mechanism commonly used in cloud-native Java applications?

Response:

- A. JSON Web Tokens (JWT)
- B. Simple Object Access Protocol (SOAP)
- C. Remote Method Invocation (RMI)
- D. Extensible Markup Language (XML)

Answer: A

Explanation: JSON Web Tokens (JWT) is a commonly used authentication mechanism in cloud-native Java applications. It allows for secure exchange of authentication and authorization information between parties, making it well-suited for distributed and microservices-based architectures.

Question: 2

Which technology is commonly used for containerization in the cloud-native ecosystem?

Response:

- A. Docker
- B. Apache Tomcat
- C. Apache Kafka
- D. Jenkins

Answer: A

Explanation: Docker is a widely used technology for containerization in the cloud-native ecosystem. It provides a platform for building, packaging, and distributing applications as lightweight and portable containers, enabling easy deployment and scalability in cloud environments.

Question: 3

Which tool is commonly used for building and managing containerized applications in the cloud-native ecosystem?

Response:

- A. Jenkins
- B. Kubernetes
- C. Ansible
- D. Apache Maven

Answer: B

Explanation: Kubernetes is a widely used container orchestration platform in the cloud-native ecosystem. It provides powerful tools for deploying, scaling, and managing containerized applications, making it an essential tool for cloud-native Java developers.

Question: 4

Which technology is commonly used for data persistence in cloud-native Java applications?

Response:

- A. Relational databases
- B. XML files

- C. Flat files
- D. Message queues

Answer: A

Explanation: Relational databases are commonly used for data persistence in cloud-native Java applications. They provide a structured and scalable approach to storing and retrieving data, making them well-suited for cloud-native environments where data needs to be managed efficiently.

Question: 5

Which of the following best describes the concept of cloud-native Java?

Response:

- A. Developing Java applications that can run on any cloud platform
- B. Developing Java applications using cloud-based IDEs
- C. Developing Java applications that are optimized for cloud environments
- D. Developing Java applications with a cloud-based database

Answer: C

Explanation: Cloud-native Java refers to the practice of developing Java applications that are specifically designed and optimized to run in cloud environments. It involves leveraging cloud-native technologies, such as containers and microservices, to build scalable, resilient, and easily deployable applications.

Question: 6

Which tool is commonly used for monitoring and troubleshooting microservices in a cloud-native environment?

Response:

- A. Prometheus
- B. Apache Kafka
- C. Elasticsearch
- D. RabbitMQ

Answer: A

Explanation: Prometheus is a popular monitoring and alerting toolkit commonly used for monitoring and troubleshooting microservices in cloud-native environments. It provides powerful metrics collection and visualization capabilities, allowing developers and administrators to gain insights into the performance and health of their microservices.

Question: 7

Which of the following is NOT a benefit of using microservices architecture for developing cloud-native applications?

Response:

- A. Improved scalability and fault isolation
- B. Enhanced developer productivity and agility
- C. Simplified deployment and management
- D. Increased security and data integrity

Answer: D

Explanation: While microservices architecture offers numerous benefits, such as improved scalability, developer productivity, and deployment simplicity, it does not inherently provide increased security and data integrity. Security

measures must still be implemented and maintained within individual microservices to ensure the security of the overall application.

Question: 8

Which component in IBM Liberty is responsible for managing Liberty server instances, including starting, stopping, and monitoring?

Response:

- A. Feature Manager
- B. Server.xml
- C. Liberty Profile
- D. Liberty Control Center

Answer: A

Explanation: The Feature Manager in IBM Liberty is responsible for managing Liberty server instances. It allows administrators to start, stop, and monitor Liberty servers, as well as install and uninstall features and applications.

Question: 9

Which of the following is NOT a common approach to testing microservices in a cloud-native Java application?

Response:

- A. Unit testing
- B. Integration testing
- C. Load testing
- D. Monolithic testing

Answer: D

Explanation: Monolithic testing is not a common approach for testing microservices in a cloud-native Java application. Monolithic testing typically refers to testing a monolithic application as a whole, rather than individual microservices. In a cloud-native architecture, where applications are composed of loosely coupled microservices, testing is typically focused on individual microservices and their interactions.



Killexams.com is a leading online platform specializing in high-quality certification exam preparation. Offering a robust suite of tools, including Exam Questions, practice tests, and advanced test engines, Killexams.com empowers candidates to excel in their certification exams. Discover the key features that make Killexams.com the go-to choice for exam success.



Practice Exam Questions Based on Current Exam Objectives

Killexams.com provides practice exam questions aligned with the latest official exam objectives and latest syllabus. Our content is reviewed and updated regularly to reflect recent changes announced by certification vendors. By studying these practice questions, candidates will cover the structure, difficulty level, and topics of the actual exam, helping them prepare more effectively and efficiently.

Comprehensive Practice Exams (PDF Format)

Killexams.com offers multiple-choice questions (MCQs) in easy-to-read PDF format, covering all major domains of the exam. Each PDF contains a structured collection of practice questions and verified answers designed to support focused study. These MCQs help candidates reinforce key concepts, identify knowledge gaps, and improve exam readiness through consistent practice.

Realistic Practice Tests (Online Test Engine & Desktop Test Engine)

To support hands-on preparation, Killexams.com provides practice tests through both an Online Test Engine and a Desktop Test Engine. These tools are designed to simulate a real exam environment, allowing candidates to practice under exam-like conditions, with latest syllabus and topics of the exam. Performance tracking, test history, and result analysis help users evaluate their progress and focus on areas that need improvement.

Risk-Free Purchase Policy

Killexams.com follows a transparent and customer-friendly purchase policy. If users are not satisfied with the study materials, they may request assistance or a refund in accordance with our published terms and conditions. This policy reflects our commitment to customer satisfaction, fairness, and confidence in our preparation resources.

Regularly Updated Content

Our practice question bank is reviewed and updated on an ongoing basis to stay aligned with the latest exam outlines and vendor updates. This ensures candidates are studying up-to-date, relevant material, and preparing with content that reflects current exam expectations, helping them stay confident and well-prepared.