Red Hat Certified Cloud-native Developer exam

Kód kurzu: EX378

The Red Hat Certified Cloud-native Developer exam (EX378) tests your skills and knowledge with regard to coding the server side of Kubernetes-native Java applications using the Quarkus framework. The exam focuses on the basic skills required to create a complete microservice using a persistent data store.By passing this exam, you become a Red Hat Certified Cloud-native Developer, which also counts toward earning a Red Hat Certified Architect (RHCA®) certification.This exam is based on Red Hat® OpenShift® Container Platform 4.5. This exam is based on Red Hat Build of Quarkus v1.7.

Pro koho je zkouška určena

- Java developers who are implementing microservices using Quarkus and Kubernetes
- Red Hat Certified professionals who wish to pursue Red Hat Certified Architect (RHCA) certification

Požadované vstupní znalosti

- Student must have taken the Red Hat Cloud-native Microservices Development with Quarkus (DO378) course or have comparable hands-on experience.
- Familiarity with using Visual Code/Codiium in a Red Hat Enterprise Linux environment.
- Good experience with JSE, including a knowledge and understanding of the core Java concepts and APIs, is necessary for this exam. For example, Exceptions, Annotations and the Collections API are all required during the exam.
- Some familiarity with Openshift/Kubernetes is beneficial.

Co musíte vědět

In preparation

Red Hat encourages you to consider taking Red Hat Cloud-native Microservices Development with Quarkus (DO378) to help you prepare. Attendance in these classes is not required; students can choose to take just the exam.
While attending Red Hat classes can be an important part of your preparation, attending class does not guarantee success on the exam. Previous experience, practice, and native aptitude are also important determinants of success. Many books and other resources on system administration for Red Hat products are available. Red Hat does not endorse any of these materials as preparation guides for exams. Nevertheless, you may find additional reading helpful to deepen your understanding.

Exam format

The Red Hat Certified Cloud-Native Developer Exam is a hands-on, practical exam that requires you to undertake real-world tasks. Internet access is not provided during the exam, and you will not be permitted to bring any hard copy or electronic documentation into the exam. This prohibition includes notes, books, or any other materials. For most exams, the documentation that ships with the product is available during the exam.

Scores and reporting

Official scores for exams come exclusively from Red Hat Certification Central. Red Hat does not authorize examiners or training partners to report results to candidates directly. Scores on the exam are usually reported within 3 U.S. business days.
Exam results are reported as total scores. Red Hat does not report performance on individual items, nor will it provide additional information upon request.

Recommended next exams or courses:

- Introduction to Containers, Kubernetes, and Red Hat OpenShift (DO180)
- Red Hat OpenShift Development I: Containerizing Applications (DO288)

GOPAS Praha
Na Strži 2097/63
140 00 Praha 4 - Krč
Tel.: +420 226 201 390
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 530 513 590
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 902 903 132
info@gopas.sk

Copyright © 2026 GOPAS, a.s.,
All rights reserved

- Building Resilient Microservices with Red Hat OpenShift Service Mesh (DO328)

Study points for the exam

To help you prepare, these exam objectives highlight the task areas you can expect to see covered in the exam. Red Hat reserves the right to add, modify, and remove exam objectives. Such changes will be made public in advance.

You should be able to perform these tasks:

Provide and obtain configuration properties through several environment-aware sources made available through dependency injection or lookup

- Externalize data into configured values.
- Inject configured values into beans using the @Inject and the @ConfigProperty qualifier.
- Access or create a configuration.
- Understand default and custom ConfigSource and ConfigSource ordering.

Build fault-tolerant Quarkus-based microservices using Microprofile Fault Tolerance strategies

- Understand the relationship to MicroProfile Config.
- Understand async versus sync execution type.
- Use @Timeout.
- Understand Retry policies and apply using @Retry.
- Understand and define Fallback.
- Understand and apply CircuitBreaker.
- Understand and apply Bulkhead.
- Understand and set up fault tolerance configuration.

Probe the state of a Quarkus application from another machine using MicroProfile Health Check

- Understand and implement the HealthCheck interface.
- Understand and apply @Liveness and @Readiness annotation.
- Understand and implement HealthCheckResponse.
- Construct human-friendly HealthCheckResponse.

Export monitoring data to management agents from a running Quarkus application using Microprofile Metrics

- Understand and use three sets of sub-resource (scopes): Base, vendor, application.
- Understand tags (labels).
- Understand and use metadata.
- Understand metric registry and @Metric.
- Expose metrics via REST API.
- Know required metrics.
- Understand Quarkus application metrics programming model.

MicroProfile Interoperable JWT RBAC on Quarkus applications: OpenID Connect (OIDC)-based JSON Web Tokens(JWT) for role-based access control (RBAC) of microservice endpoints

- Understand token-based authentication.
- Use JWT bearer tokens to protect services.
- Mark a JAX-RS application as requiring MP-JWT access control.
- Map MP-JWT tokens to Java EE container APIs.

Implement a Quarkus application and expose REST service endpoints with JAX-RS

- Understand REST concepts, particularly the application and use of the HTTP PUT, DELETE, GET, and POST methods.
- Know and use standard HTTP return codes.
- Implement RESTful root resource class.
- Expose a REST service using JAX-RS.
- Understand and use @Path, @Produce, and @Consume.

GOPAS Praha
Na Strži 2097/63
140 00 Praha 4 - Krč
Tel.: +420 226 201 390
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 530 513 590
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 902 903 132
info@gopas.sk

Copyright © 2026 GOPAS, a.s.,
All rights reserved

- Using CDI to integrate components.
- Using bean validation to ensure data format and consistency.

Simplified JPA mapping with Panache

- Understand the difference between the Active Record Pattern and the Repository Pattern.
- Use basic JPA to create, read, update, and delete persistent objects and their relationships.
- Map a bi-directional one-to-many relationship between two entities, including both sides of the association.
- Demonstrate the ability to perform the most common Panache operations and add custom entity methods.

Microprofile OpenAPI specification to document RESTful APIs

- Understand OpenAPI documents and the Swagger UI to discover remote services APIs.
- Demonstrate the ability to link to semantic versioning (semver) remote service endpoints.
- Understand how to produce the default and custom OpenAPI document to JAX-RS endpoints.

Interacting with REST APIs in Quarkus using the MicroProfile REST Client

- Understand the type-safe approach to invoke RESTful services over HTTP using the JAX-RS APIs.
- Understand REST concepts, particularly the application and use of the HTTP PUT, DELETE, GET, and POST methods.
- Demonstrate the ability to create and use a REST client to connect with a remote service.
- Parameterize and configure the REST client URI to invoke a specific remote microservice.
- Understand and use special additional client headers.

As with all Red Hat performance-based exams, configurations must persist after restart without intervention.

GOPAS Praha
Na Strži 2097/63
140 00 Praha 4 - Krč
Tel.: +420 226 201 390
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 530 513 590
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 902 903 132
info@gopas.sk

Copyright © 2026 GOPAS, a.s.,
All rights reserved