



Home > Training > Certification > Java Technology >

Sun Certified Enterprise Architect for the Java Platform, Enterprise Edition 5 (Step 1 of 3) (CX-310-052)

Product Description

This exam is for enterprise architects responsible for architecting and designing Java Platform, Enterprise Edition technology-compliant applications that are scalable, flexible, and highly secure. This exam is designed for candidates with experience in the following areas (typically 5 or more years: Application design: concepts and principles, common architectures, integration and messaging, business-tier technologies, web-tier technologies, Java EE technology, patterns and security. If a candidate has NOT taken the SCEA 5 certification exam, Sun recommends the candidate take the new SCEA 5 certification exam. If a candidate has taken and passed all three steps to the SCEA (CX-310-051/CX-310-300A,CX-310-052), Sun recommends that the candidate take the SCEA 5 Upgrade exam. The upgrade exam is for candidates who have successfully been certified on a previous version of the SCEA.

» [Details below](#)

Product ID
CX-310-052

Price
\$200.00

[Add To Shopping Cart »](#)

Exams purchased on the Web site may only be used in the US. If you reside outside the US please [select a country](#) to inquire about products delivered in your country. Once exam vouchers are purchased you have up to one year from the date of purchase to use it. Each voucher is valid for one exam and may only be used at an Authorized Prometric Testing Center in the country for which it was purchased. Please be aware that exam vouchers are nonrefundable for any reason.

Details

- Delivered at: Authorized Worldwide Prometric Testing Centers
- Prerequisites: None
- Other exams/assignments required for this certification: Step 2 (CX-310-301A), Step 3 (CX-310-062)
- Exam type: Multiple choice, and drag-n-drop
- Number of questions: 64
- Pass score: 57% (37 out of 64 questions)
- Time limit: 120 minutes

⤴ [Back to top](#)

Languages

English

⤴ [Back to top](#)

Supporting Courses

No supporting courses currently available.

⤴ [Back to top](#)

Assignment Objectives

Section 1: Application Design Concepts and Principles

- Explain the main advantages of an object-oriented approach to system design including the effect of encapsulation, inheritance, and use of interfaces on architectural characteristics.
- Describe how the principle of "separation of concerns" has been applied to the main system tiers of a Java Platform, Enterprise Edition application. Tiers include client (both GUI and web), web (web container), business (EJB container), integration, and resource tiers.
- Describe how the principle of "separation of concerns" has been applied to the layers of a Java EE application. Layers include application, virtual platform (component APIs), application infrastructure (containers), enterprise services (operating system and virtualization), compute and storage, and the networking infrastructure layers.

Section 2: Common Architectures

- Explain the advantages and disadvantages of two-tier architectures when examined under the following topics: scalability, maintainability, reliability, availability, extensibility, performance, manageability, and security.
- Explain the advantages and disadvantages of three-tier architectures when examined under the following topics: scalability, maintainability, reliability, availability, extensibility, performance, manageability, and security.
- Explain the advantages and disadvantages of multi-tier architectures when examined under the following topics: scalability, maintainability, reliability, availability, extensibility, performance, manageability, and security.
- Explain the benefits and drawbacks of rich clients and browser-based clients as deployed in a typical Java EE application.
- Explain appropriate and inappropriate uses for web services in the Java EE platform.

Section 3: Integration and Messaging

- Explain possible approaches for communicating with an external system from a Java EE technology-based system given an outline description of those systems and outline the benefits and drawbacks of each approach.
- Explain typical uses of web services and XML over HTTP as mechanisms to integrate distinct software components.
- Explain how JCA and JMS are used to integrate distinct software components as part of an overall Java EE application.

Section 4: Business Tier Technologies

- Explain and contrast uses for entity beans, entity classes, stateful and stateless session beans, and message-driven beans, and understand the advantages and disadvantages of each type.
- Explain and contrast the following persistence strategies: container-managed persistence (CMP) BMP, JDO, JPA, ORM and using DAOs (Data Access Objects) and direct JDBC technology-based persistence under the following headings: ease of development, performance, scalability, extensibility, and security.
- Explain how Java EE supports the deployment of server-side components implemented as web services and the advantages and disadvantages of adopting such an approach.
- Explain the benefits of the EJB 3 development model over previous EJB generations for ease of development including how the EJB container simplifies EJB development.

Section 5: Web Tier Technologies

- State the benefits and drawbacks of adopting a web framework in designing a Java EE application.
- Explain standard uses for JSP pages and servlets in a typical Java EE application.
- Explain standard uses for JavaServer Faces components in a typical Java EE application.
- Given a system requirements definition, explain and justify your rationale for choosing a web-centric or EJB-centric implementation to solve the requirements. Web-centric means that you are providing a solution that does not use EJB components. EJB-centric solution will require an application server that supports EJB components.

Section 6: Applicability of Java EE Technology

- Given a specified business problem, design a modular solution that solves the problem using Java EE.
- Explain how the Java EE platform enables service oriented architecture (SOA) -based applications.
- Explain how you would design a Java EE application to repeatedly measure critical non-functional requirements and outline a standard process with specific strategies to refactor that application to improve on the results of the measurements.

Section 7: Patterns

- From a list, select the most appropriate pattern for a given scenario. Patterns are limited to those documented in the book - Alur, Crupi and Malks (2003). Core J2EE Patterns: Best Practices and Design Strategies 2nd Edition and named using the names given in that book.
- From a list, select the most appropriate pattern for a given scenario. Patterns are limited to those documented in the book - Gamma, Erich; Richard Helm, Ralph Johnson, and John Vlissides (1995). Design Patterns: Elements of Reusable Object-Oriented Software and are named using the names given in that book.
- From a list, select the benefits and drawbacks of a pattern drawn from the book - Gamma, Erich; Richard Helm, Ralph Johnson, and John Vlissides (1995). Design Patterns: Elements of Reusable Object-Oriented Software.
- From a list, select the benefits and drawbacks of a specified Core J2EE pattern drawn from the book – Alur, Crupi and Malks (2003). Core J2EE Patterns: Best Practices and Design Strategies 2nd Edition.

Section 8: Security

- Explain the client-side security model for the Java SE environment, including the Web Start and applet deployment modes.
- Given an architectural system specification, select appropriate locations for implementation of specified security features, and select suitable technologies for implementation of those features
- Identify and classify potential threats to a system and describe how a given architecture will address the threats.
- Describe the commonly used declarative and programmatic methods used to secure applications built on the Java EE platform, for example use of deployment descriptors and JAAS.

[⤴ Back to top](#)

Copyright 1994-2007 Sun Microsystems, Inc.