

Investigue los temas siguientes:

Java EE:

También conocido como Java Enterprise Edition, es una plataforma de desarrollo de software empresarial diseñada para desarrollar aplicaciones web escalables y robustas. Java EE viene con una amplia gama de componentes, que se utilizan para construir diferentes tipos de aplicaciones empresariales. Estos componentes proporcionan una amplia variedad de servicios, desde la gestión de transacciones hasta la creación de interfaces de usuario. En este artículo, vamos a explorar los diferentes componentes de Java EE y su función en el desarrollo de aplicaciones empresariales.

¿Qué incluye Java EE?

Java EE (Java Enterprise Edition) es una plataforma de desarrollo para aplicaciones empresariales que proporciona un conjunto de especificaciones, APIs y herramientas para simplificar el desarrollo, despliegue y gestión de aplicaciones escalables y seguras.

Java EE incluye una serie de características y tecnologías que permiten el desarrollo de aplicaciones web, servicios web, aplicaciones distribuidas, aplicaciones de mensajería y mucho más. Algunas de las características más importantes de Java EE son:

1. **Servlets:** son componentes Java que se utilizan para procesar las solicitudes y respuestas HTTP.
2. **JSP (JavaServer Pages):** es una tecnología que permite la creación de páginas web dinámicas basadas en Java.
3. **JDBC (Java Database Connectivity):** es una API que permite a las aplicaciones Java interactuar con bases de datos.
4. **JTA (Java Transaction API):** es una API que permite la gestión de transacciones en aplicaciones empresariales.
5. **JMS (Java Message Service):** es una API que permite la comunicación asíncrona entre aplicaciones.
6. **EJB (Enterprise JavaBeans):** es una tecnología que permite la creación de componentes empresariales reutilizables.

7. JPA (Java Persistence API): es una API que permite la persistencia de objetos Java en una base de datos relacional.

8. JAX-WS (Java API for XML Web Services): es una API que permite el desarrollo de servicios web basados en XML.

En resumen, Java EE incluye un conjunto de tecnologías y APIs que facilitan el desarrollo de aplicaciones empresariales escalables, seguras y eficientes.

JAVA-BASED WEB APPLICATION FRAMEWORK

JavaServer Faces (JSF) es un marco de interfaz de usuario (UI) para aplicaciones web Java. Está diseñado para aliviar significativamente la carga de escribir y mantener aplicaciones que se ejecutan en un servidor de aplicaciones Java y devolver sus interfaces de usuario a un cliente de destino. JSF proporciona facilidad de uso de las siguientes maneras:

Facilita la construcción de una interfaz de usuario a partir de un conjunto de componentes de interfaz de usuario reutilizables

Simplifica la migración de datos de aplicaciones hacia y desde la interfaz de usuario

Ayuda a administrar el estado de la interfaz de usuario en las solicitudes del servidor

Proporciona un modelo sencillo para conectar eventos generados por el cliente al código de la aplicación del lado del servidor

Permite que los componentes de la interfaz de usuario personalizados se construyan y reutilicen fácilmente

Para obtener una descripción detallada del marco JSF, consulte el Tutorial de Java EE 7, Capítulo 12 Desarrollo con la tecnología JavaServer Faces.

En este tutorial se muestra cómo se puede aplicar la compatibilidad con JSF 2.x a una aplicación web mediante el IDE de NetBeans. Para empezar, agregue compatibilidad con el marco JSF 2.x a una aplicación web básica y, a continuación, proceda a realizar las siguientes tareas:

Crear un bean gestionado por JSF para gestionar los datos de las solicitudes,

Conecte el bean administrado a las páginas web de la aplicación, y

Convertir las páginas web en archivos de plantilla de Facelets.

El IDE de NetBeans ha proporcionado soporte de larga data para JavaServer Faces. A partir de la versión de JSF 2.0 y Java EE 6, NetBeans IDE ha proporcionado soporte para JSF 2.0 y JSF 2.1. Para obtener más información, consulte Compatibilidad con JSF 2.x en el IDE de NetBeans.

La tecnología JavaServer Faces incluye:

Un conjunto de API para representar componentes de la interfaz de usuario y administrar su estado, controlar eventos y validación de entradas, definir la navegación de páginas y admitir la internacionalización y la accesibilidad.

Una biblioteca de etiquetas personalizadas de JavaServer Pages (JSP) para expresar una interfaz JavaServer Faces dentro de una página JSP.

Diseñada para ser flexible, la tecnología JavaServer Faces aprovecha los conceptos existentes y estándar de interfaz de usuario y nivel web sin limitar a los desarrolladores a un lenguaje de marcado, protocolo o dispositivo cliente en particular. Las clases de componentes de interfaz de usuario incluidas con la tecnología JavaServer Faces encapsulan la funcionalidad del componente, no la presentación específica del cliente, lo que permite que los componentes de la interfaz de usuario de JavaServer Faces se representen en varios dispositivos cliente. Al combinar la funcionalidad del componente de la interfaz de usuario con renderizadores personalizados, que definen los atributos de representación para un componente específico de la interfaz de usuario, los desarrolladores pueden construir etiquetas personalizadas para un dispositivo cliente determinado. Para mayor comodidad, la tecnología JavaServer Faces proporciona un renderizador personalizado y una biblioteca de etiquetas personalizadas JSP para renderizar en un cliente HTML, lo que permite a los desarrolladores de aplicaciones Java Platform, Enterprise Edition (Java EE) utilizar la tecnología JavaServer Faces en sus aplicaciones.

Siendo la facilidad de uso el objetivo principal, la arquitectura JavaServer Faces define claramente una separación entre la lógica de la aplicación y la presentación, al tiempo que facilita la conexión de la capa de presentación al código de la aplicación. Este diseño permite que cada miembro de un equipo de desarrollo de aplicaciones web se concentre en su parte del proceso de desarrollo, y también proporciona un modelo de programación simple para vincular las piezas. Por

ejemplo, los desarrolladores de páginas web sin experiencia en programación pueden utilizar las etiquetas de componentes de la interfaz de usuario de JavaServer Faces para vincular al código de la aplicación desde una página web sin escribir ningún script.

Referencias

Introducción a JavaServer Faces 2.x (2024) Apache NetBeans 22. Internet. Recuperado de: <https://netbeans.apache.org/tutorial/main/kb/docs/web/jsf20-intro/>

Jimenez Santiago (2024) ¿Cuáles son los componentes de Java EE? Internet. Recuperado de: <https://aprenderjava.net/javaee/cuales-son-los-componentes-de-java-ee/>