

Clase 26/11/2020

Funcionamiento servicio REST

Servicios.zip	Contiene archivo JAVA y archivos de configuración de un servicio web estilo REST
prueba.html	Contiene una aplicación web
WSClient.js	funciones para invocar el servicio web mediante AJAX

Servicios.zip

META-INF contiene el archivo context.xml, el cual se configura :

Name de la etiqueta Resource, define el nombre del datasource , un datasource configura las conexiones que realiza el servicio web sin tener que escribir los parámetros en el código.

URL -> define el nombre de la base de datos

Password -> la contraseña

WEB-INF contiene los directorios classes y web.xml

web.xml configura lo siguiente

load-on-startup igual a 1 indique se debe cargar el servicio cuando inicie Tomcat

url-pattern indica la suma del servicio web

resource-ref indica el nombre del datasource

res-tape indica el tipo de recurso java.sql.DataSource

En el directorio de clases se colocan las clases compiladas del servicio web

Carpeta Negocio, los archivos incluidos en este directorio son los siguientes

AdaptadorGsonBase64.java -> permite modificar la forma en que GSON serializa los campos

Error.java -> Regresa un mensaje de error dentro de un objeto

Foto.java -> encapsula un arreglo de bytes y una clave numerica

Usuario.java -> Encapsula los datos del usuario. Es muy importante notar que la anotación @FormParam requiere un método que convierta una String a objeto de tipo Usuario. En este caso se implementa el método **valueOf** para este propósito.

Servicio.java -> Implementa los métodos del servicio web

Pool de conexiones

Es conveniente realizar un pool de conexiones mediante un datasource ya que abrir una conexión a la base de datos es lento.

Así mismo, el uso de datasource y pool de conexiones permite establecer un número de conexiones a la base de datos y mantenerlas abiertas siempre. Cada vez que un método del servicio web requiera conectarse a la base de datos,

solicita una conexión disponible en el pool.

El servicio web produce JSON

Incluye una anotación `@Produces`, esto indica que el método regresara los resultados en formato JSON

Servicio web consume URL encoded

Cada metodo incluye `@Consume` lo cual indica que los parámetros del método se recibirán codificados como URL.

Nombre del metodo web

Cada metodo incluye `@Path`, esta anotación define el nombre del metodo web el cual puede ser diferente al nombre del metodo. por ejemplo el metodo "alta_usuario" se invocará como el metodo web "alta"

Los nombres de los parámetros del metodo web

Para indicar el nombre de cada parámetro del metodo web se utiliza la anotación `@FormParam`

Metodo web regres Response

Basicamente existen tres tipos de respuestas que el metodo web regresa de una instancia de la clase Response

- No regresa resultados -> HTTP 200

- Regresa resultados

- Regresa un mensaje de error

El metodo web debe de cerrar la conexion con la BD