

Package Controller

Class Summary

[AddBook](#)

Clase AddBook: La acción que hace esta esta clase es recuperar los datos obtenidos de la vista para poder ingresarlos a la base de datos.

[DeleteBook](#)

Clase DeleteBook: La acción que hace esta esta clase es recuperar el isbn del libro para poder eliminarlo de la base de datos.

[EditBook](#)

Clase EditBook: La acción que hace esta esta clase es recuperar los datos de la base de datos, para mostrarlo en la vista y tambien recupera los datos que sufrieron un cambio para poder hacer la actualiazción.

[ListBooks](#)

Clase ListBook: La acción que hace esta esta clase es recuperar todos datos que estan la base de datos para poder imprimirlos en la primera vista

Controller

Class AddBook

```
java.lang.Object
|
+--Controller.AddBook
```

< [Constructors](#) > < [Methods](#) >

```
public class AddBook
extends java.lang.Object
```

Clase AddBook: La acción que hace esta esta clase es recuperar los datos obtenidos de la vista para poder ingresarlos a la base de datos.

Author:

josed

Constructors

AddBook

```
public AddBook()
```

Hace la conexcion a la base de datos y crea un objeto para poder validar los campo

Methods

guardarDatos

```
public org.springframework.web.servlet.ModelAndView guardarDatos(Libro u,  
org.springframework.validation.BindingResult result,  
org.springframework.web.bind.support.SessionStatus status)
```

Guarda los datos recuperados por el usuario.

Parameters:

u - Que ingreso el usuario

Returns:

mav, ModelAndView("redirect:/index.htm" -Si encuentra un problema de validacion, se regresa a la pagina para ingresar un libro con los errores, si no se hubieron errores se va a la vista principal para ver los libros

mostrarDatos

```
public org.springframework.web.servlet.ModelAndView mostrarDatos()
```

Al ingresar a la página para poder agregar un libro, se le pasa un objeto de tipo libro para poder guardar los datos momentaneamente.

Returns:

mav -Regresa la vista para el usuario

Controller

Class DeleteBook

```
java.lang.Object  
|  
+--Controller.DeleteBook
```

```
public class DeleteBook
extends java.lang.Object
```

Clase DeleteBook: La acción que hace esta esta clase es recuperar el isbn del libro para poder elimnarlo de la base de datos.

Author:
josed

Constructors

DeleteBook

```
public DeleteBook()
```

Se establece la conexion a la base de datos.

Methods

borrarLibro

```
public org.springframework.web.servlet.ModelAndView
borrarLibro(javax.servlet.http.HttpServletRequest request)
```

Se establece la conexion a la base de datos.

Controller

Class EditBook

```
java.lang.Object
|
+--Controller.EditBook
```

< [Constructors](#) > < [Methods](#) >

```
public class EditBook
extends java.lang.Object
```

Clase EditBook: La acción que hace esta esta clase es recuperar los datos de la base de datos, para mostrarlo en la vista y tambien recupera los datos que sufrieron un cambio para poder hacer la actualiazción.

Author:
josed

Constructors

EditBook

```
public EditBook()
```

Se establece la conexion a la base de datos.

Methods

actualizarDatos

```
public org.springframework.web.servlet.ModelAndView actualizarDatos(Libro u,  
org.springframework.validation.BindingResult result,  
org.springframework.web.bind.support.SessionStatus status,  
javax.servlet.http.HttpServletRequest request)
```

Actualiza los datos recuperados por el usuario.

Parameters:

u - Son los datos que ingreso el usuario.

Returns:

mav, ModelAndView("redirect:/index.htm" -Si encuentra un problema de validacion, se regresa a la pagina para ingresar un libro con los errores, si no se hubieron errores se va a la vista principal para ver los libros

mostrarDatos

```
public org.springframework.web.servlet.ModelAndView  
mostrarDatos(javax.servlet.http.HttpServletRequest request)
```

Al ingresar a la página para poder editar un libro, se le pasa un objeto de tipo libro para poder guardar los datos momentaneamente.

Returns:

mav -Regresa la vista para el usuario

selectLibro

```
public Libro selectLibro(java.lang.String isbn)
```

Recupera los datos de la base de datos.

Parameters:

isbn - El la llave primaria del libro.

Returns:

Regresa un objeto de tipo libro con los datos recueperados de la base de datos.

Controller

Class ListBooks

```
java.lang.Object
|
+--Controller.ListBooks
```

< [Constructors](#) > < [Methods](#) >

```
public class ListBooks
extends java.lang.Object
```

Clase ListBook: La acciÃ³n que hace esta esta clase es recuperar todos datos que estan la base de datos para poder imprimirlos en la primera vista

Constructors

ListBooks

```
public ListBooks()
```

Se establece la conexion a la base de datos.

Methods

mostrarLibros

```
public org.springframework.web.servlet.ModelAndView mostrarLibros()
```

Recupera los datos de la base de datos para poder imprimirlos en la vista

Returns:

Regresa la vista principal del programa

Package Model

Class Summary

[Conexion](#)

Clase Conexion: Lo unico que hace esta clase es conectarse a la base de datos.

[Libro](#)

Clase Libro: Este es el objeto libro que no adyudarÃ¡ para poder guardar los datos necesarios

[LibrosValidaciones](#)

Clase LibrosValidaciones: La que hace esta clase es validar que los campos para poder ingresar un libro no se encuentren vacios o nulos

Model

Class Conexion

```
java.lang.Object
|
+--Model.Conexion
```

< [Constructors](#) > < [Methods](#) >

```
public class Conexion
extends java.lang.Object
```

Clase Conexion: Lo unico que hace esta clase es conectarse a la base de datos.

Author:

josed

Constructors

Conexion

```
public Conexion()
```

Methods

conectar

```
public org.springframework.jdbc.datasource.DriverManagerDataSource conectar()
```

Hace la conexcion a la base de datos y crea un objeto para poder validar los campo

Model

Class Libro

```
java.lang.Object
|
+--Model.Libro
```

< [Constructors](#) > < [Methods](#) >

```
public class Libro
extends java.lang.Object
```

Clase Libro: Este es el objeto libro que no adyudarÃ¡ para poder guardar los datos necesarios

Author:
josed

Constructors

Libro

```
public Libro()
```

Libro

```
public Libro(java.lang.String nombre,
            java.lang.String autor,
            int version,
            java.lang.String editorial)
```

Constructor usado unicamente para poder actualizar datos de libro.

Libro

```
public Libro(java.lang.String isbn,
             java.lang.String nombre,
             java.lang.String autor,
             int version,
             java.lang.String editorial)
```

Constructor usado para crear un libro.

Methods

getAutor

```
public java.lang.String getAutor()
```

Regresa el valor del Autor

getEditorial

```
public java.lang.String getEditorial()
```

Regresa el valor de la Editorial

getIsbn

```
public java.lang.String getIsbn()
```

Regresa el valor del ISBN

getNombre

```
public java.lang.String getNombre()
```

Regresa el valor del Nombre

getVersion

```
public int getVersion()
```

Regresa el valor de version

setAutor

```
public void setAutor(java.lang.String autor)
```

Funcion set para poder cambiar el valor del autor

setEditorial

```
public void setEditorial(java.lang.String editorial)
```

Funcion set para poder cambiar el valor de la editorial

setIsbn

```
public void setIsbn(java.lang.String isbn)
```

Funcion set para poder cambiar el valor del isbn

setNombre

```
public void setNombre(java.lang.String nombre)
```

Funcion set para poder cambiar el valor del nombre

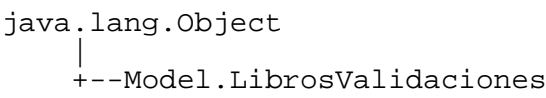
setVersion

```
public void setVersion(int version)
```

Funcion set para poder cambiar el valor de version

Model

Class LibrosValidaciones



All Implemented Interfaces:
org.springframework.validation.Validator

< [Constructors](#) > < [Methods](#) >

```
public class LibrosValidaciones
extends java.lang.Object
implements org.springframework.validation.Validator
```

Clase LibrosValidaciones: La que hace esta clase es validar que los campos para poder ingresar un libro no se encuentren vacios o nulos

Author:
josed

Constructors

LibrosValidaciones

```
public LibrosValidaciones()
```

Methods

supports

```
public boolean supports(java.lang.Class type)
```

validate

```
public void validate(java.lang.Object o,  
                    org.springframework.validation.Errors errors)
```

Verifica que los campos de las vistas add y edit no se encuentren en null o vacios

Parameters:

o - recupera el objeto con los valores a probar
errors - devuelve el error obtenido por la página