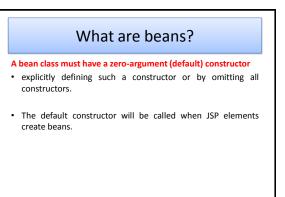


Duri Hatal Thalan D

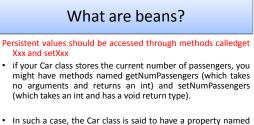


What are beans?

A bean class should have no public instance variables (fields)

- a class should use accessor methods instead of allowing direct access to the instance variables.
- (a) impose constraints on variable values (e.g., have the setSpeed method of your Car class disallow negative speeds);
- (b) change your internal data structures (e.g., change from English units to metric units internally, but still have getSpeedInMPH and getSpeedInKPH methods);
- (c) perform side effects automatically when values change (e.g., update the user interface when setPosition is called).

By : - Hetal Thaker 5



- m such a case, the car class is said to have a property named numPassengers.
- If the class has a getXxx method but no corresponding setXxx, the class is said to have a read-only property named xxx.

By : - Hetal Thaker 6

What are beans?

Persistent values should be accessed through methods calledget Xxx and setXxx

- The one exception to this naming convention is with boolean properties: they are permitted to use a method called isXxx to look up their values.
- for example, your Car class might have methods called isLeased (which takes no arguments and returns a boolean) and setLeased (which takes a boolean and has a void return type), and would be said to have a boolean property named leased

Build and manipulate beans in JSP

 \rightarrow jsp:useBean \rightarrow this element builds a new bean.

<jsp:useBean id="beanName" class="package.Class" />

Providing a scope attribute the jsp:useBean element can either build a new bean or access a preexisting one.

Build and manipulate beans in JSP • jsp:getProperty → This element reads and outputs the value of a

 jsp:getProperty → This element reads and outputs the value of a bean property.(calling a method of the form getXxx)

<jsp:getProperty name="beanName" property="propertyName" />

jsp:setProperty → This element modifies a bean property (calls a method of the form setXxx)

<jsp:setProperty name="beanName" property="propertyName"
value="propertyValue" />

: - Hetal Thaker 9

By : - Hetal Thaker 11

By : - Hetal Thaker 7

Building bean : jsp:useBean

- lets you load a bean to be used in the JSP page.
- reusability of Java classes.
- Syntax : <jsp:useBean id="name" class="package.Class" />
- "instantiate an object of the class specified by Class, and bind it to a variable in _jspService with the name specified by id."
- You must use the fully qualified class name for the class attribute of jsp:useBean.
- <jsp:useBean id="book1" class="coreservlets.Book" />

By : - Hetal Thaker 10

By : - Hetal Thaker

Installing bean classes

- The bean class definition should be placed in the same directories where servlets can be installed, not in the directory that contains the JSP file.
- the proper location for individual bean classes is WEB-INF/classes/subdirectoryMatchingPackageName. Place all your beans in packages.

Using jsp:useBean Options: scope, beanName, and type

- specify a scope attribute that associates the bean with more than just the current page.
- If beans can be shared, it is useful to obtain references to existing beans, rather than always building a new object.
- the jsp:useBean action specifies that a new object is instantiated only if there is no existing one with the same id and scope.

By : - Hetal Thaker 12

Using jsp:useBean Options: scope, beanName, and type

 Rather than using the class attribute, you are permitted to use beanName instead. The difference is that beanName can refer either to a class or to a file containing a serialized bean object. The value of the beanName attribute is passed to the instantiate method of java.beans.Bean.

Using jsp:useBean Options: scope, beanName, and type

- the variable to be declared to have a type that is a superclass of the actual bean type or is an interface that the bean implements. Use the type attribute to control this declaration, as in the following example.
- <jsp:useBean id="thread1" class="mypackage.MyClass" type="java.lang.Runnable" />

This use results in code similar to the following being inserted into the _jspService method.

java.lang.Runnable thread1 = new myPackage.MyClass();

Accessing Bean Properties: jsp:getProperty

- Once you have a bean, you can output its properties with jsp:getProperty, which takes a name attribute that should match the id given in jsp:useBean and a property attribute that names the property of interest.
- Instead of using jsp:getProperty, you could use a JSP expression and explicitly call a method on the object with the variable name specified by the id attribute.
- <jsp:getProperty name="book1" property="title" />
- <%= book1.getTitle()%>

: - Hetal Thaker 15

By : - Hetal Thaker

Setting Simple Bean Properties: isp:setProperty

- To modify bean properties, you normally use jsp:setProperty.
- This action has several different forms, but with the simplest form you supply three attributes: name (which should match the id given by jsp:useBean), property (the name of the property to change), and value (the new value).
- <jsp:setProperty name="book1" property="title" value="Core Servlets and JavaServer Pages" />
- <% book1.setTitle("Core Servlets and JavaServer Pages"); %>

By : - Hetal Thaker 1