

Работа с JavaBeans

JSP & Servlets
13 уровень, 5 лекция

ОТКРЫТА

6.1 Что такое JavaBeans

Уже в конце 90-х язык Java начал активно использоваться для больших серверных приложений, где количество классов измеряется десятками и сотнями тысяч. Тогда и появилась идея стандартизировать вид Java-объектов.

Весь язык Java трогать не стали, чтобы не лишать его гибкости. Ну, обратная совместимость и все такое. Тогда разработали ряд критериев для Java-объектов нового поколения и назвали такие объекты Java Beans. Java названа в честь популярной марки кофе, так что Java Beans дословно переводится, как «кофе бобы».

Самые важные критерии были такие:

- Доступ к внутренним полям класса идет через метод `getProperty()`.
- Запись данных в поля класса идет через метод `setProperty(value)`.
- У класса должен быть **публичный конструктор без параметров**.
- Класс должен быть сериализуемым.
- Класс должен иметь переопределенные методы `equals()`, `hashCode()` и `toString()`.

Такой подход позволил сделать приложения менее связными. Всегда понятно:

- как создать объект – есть публичный конструктор по умолчанию;
- как получить/установить значение свойства;
- как передать/сохранить объект (используем сериализацию);
- как сравнивать объекты (используем `equals()` и `hashCode()`);
- как вывести информацию об объекте в лог (используем `toString()`).

Сейчас это фактически стандарт отрасли, а когда-то был новым веянием. Кажется, что уже все так пишут, хотя если вспомнить HttpClient и его Builders, то видно, что новый стандарт кому-то сильно жмет.

Такие объекты получили широкое распространение там, где их основная смысловая нагрузка, — это хранение данных. Например, в графических интерфейсах, базах данных и JSP-страницах.

6.2 JSP и JavaBeans

Одной из причин появления JSP было то, что работу над ним можно было отдать фронтендщикам. А что? У тебя есть человек, который разбирается в HTML, пусть он JSP и пишет. Java-программисты пишут свою часть, фронтендщики свою – все отлично.

И все было неплохо до тех пор, пока фронтендщикам не пришлось разбираться в написанном Java-коде, встроенном в JSP. Или, что еще хуже, писать такой код самим.

Java-программисты тоже от такого были не в восторге. Ну скажите на милость, какие из верстальщиков бэкенд-разработчики? Да они же кроме скриптов ничего писать не могут. Да и вся парадигма программирования утверждает, что смешивать разные языки в одном файле – дурной тон.

Тогда и появилась идея, что мол дать фронтендщикам возможность работать с Java-объектами, как с HTML-кодом. Каждый HTML-тег – это тоже некий объект со своими полями, почему бы по похожему принципу не построить работу с Java-объектами?

Сказано-сделано. Добавили специальные теги и понеслось.

Создание объекта:

```
<jsp:useBean id="имя" class="тип-объекта" scope="session"/>
```

Эта команда создавала объект с типом `тип-объекта` и клала его в `сессия` под именем `имя`.

Объекты можно было хранить в одном из четырех хранилищ: application (глобальное), session, request и page. Также можно было установить свойство таких объектов:

```
<jsp:setProperty name="имя" property="propName" value="string constant"/>
```

Получить свойство таких объектов можно было так:

```
<jsp:getProperty name="имя" property="propName"/>
```

Пример использования тегов:

```
1 <body>
2   <center>
3     <h2>Using JavaBeans in JSP</h2>
4     <jsp:useBean id = "test" class = "com.example.TestBean" />
5     <jsp:setProperty name = "test" property = "message" value = "Hello JSP..." />
6     <p> Что-то делаем важное</p>
7     <jsp:getProperty name = "test" property = "message" />
8   </center>
9 </body>
```

< Предыдущая лекция

Следующая лекция >



Комментарии

популярные новые старые

JavaCoder

Введите текст комментария



У ЭТОЙ СТРАНИЦЫ ЕЩЕ НЕТ НИ ОДНОГО КОММЕНТАРИЯ

ОБУЧЕНИЕ

- Курсы программирования
- Курс Java
- Помощь по задачам
- Подписки
- Задачи-игры

СООБЩЕСТВО

- Пользователи
- Статьи
- Форум
- Чат
- Истории успеха
- Активности

КОМПАНИЯ

- О нас
- Контакты
- Отзывы
- FAQ
- Поддержка



JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

Русский

СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ

