

Настройка веб-приложений с помощью аннотаций

JSP & Servlets
11 уровень, 7 лекция

ОТКРЫТА

8.1 Знакомство с аннотациями сервлетов

Начиная с 7-й версии Tomcat стал поддерживать спецификацию веб-сервлетов **Servlet API 3.0**. В ней в частности появился пакет под названием `javax.servlet.annotation`. Он содержит различные типы аннотаций, которые можно использовать для аннотирования класса сервлета. Если ты используешь аннотацию, то дескриптор развертывания (`web.xml`) не требуется.

Список самых полезных аннотаций:

	Аннотация	Описание
1	<code>@WebServlet</code>	Объявляет сервлет
2	<code>@WebInitParam</code>	Указывает параметр инициализации
3	<code>@WebFilter</code>	Объявляет веб-фильтр
4	<code>@WebListener</code>	Объявляет веб-лисенер
5	<code>@ServletSecurity</code>	Позволяет настроить параметры безопасности

Пример:

```
1  @WebServlet( urlPatterns = {"/api/*"} )
2  public class Example extends HttpServlet {
3      protected void doGet( HttpServletRequest request, HttpServletResponse response) throws Exception {
4          response.setContentType("text/html");
5          PrintWriter out = response.getWriter();
6          out.println("Hello World!");
7      }
8  }
```

Одной аннотации достаточно, чтобы объявить сервлет и замапить его на обслуживание всех запросов, которые будут приходить на URL, заданный шаблоном `/api/*`.

8.2 Настройка мапинга сервлета

У аннотации `@WebServlet` много параметров, которые позволяют очень гибко сконфигурировать сервлет, которые она описывает. Давай ознакомимся с основными из них:

	Атрибут	Описание
1	<code>name</code>	Уникальное имя сервлета (как в <code>web.xml</code>)
2	<code>displayName</code>	Читаемое имя сервлета для человека
3	<code>description</code>	Описание сервлета

4	<code>value</code>	Задает url для мапинга
5	<code>urlPatterns</code>	Задает список url-ов для мапинга (используется вместо value)
6	<code>initParams</code>	Позволяет задать стартовые параметры сервлета
7	<code>asyncSupported</code>	Указывает, что сервлет может работать асинхронно (HTTP/2)
8	<code>loadOnStartup</code>	Порядковый номер для управления приоритетом старта сервлетов
9	<code>smallIcon</code>	Задает маленькую иконку сервлета
10	<code>largeIcon</code>	Задает большую иконку сервлета

Интересных моментов тут несколько.

Во-первых, обрати внимание, что `value` и `urlPatters` — взаимозаменяемые атрибуты. Они оба позволяют задать список урлов, на которые должен замапиться сервлет.

Во-вторых, параметр `asyncSupported` указывает, будет ли сервлет корректно обрабатывать асинхронные запросы по протоколу HTTP/2.

И третий важный атрибут — это `initParams`, он позволяет задать список параметров, которые будут помещены в ServletContext.

Пример:

```
1  @WebServlet(  
2      urlPatterns = {"/sendFile", "/uploadFile"},  
3      loadOnStartup = 1,  
4      asyncSupported = true,  
5      initParams = {  
6          @WebInitParam(name = "saveDir", value = "c:/uploaded"),  
7          @WebInitParam(name = "allowedTypes", value = "jpg,gif,png")  
8      }  
9  )  
10 public class ImageUploadServlet extends HttpServlet {  
11  
12     public void doGet(HttpServletRequest request, HttpServletResponse response) throws Exception {  
13         String saveDir = getInitParameter("saveDir");  
14         String fileTypes = getInitParameter("allowedTypes");  
15  
16         PrintWriter writer = response.getWriter();  
17  
18         writer.println("saveDir = " + saveDir);  
19         writer.println("fileTypes = " + fileTypes);  
20     }  
21 }
```

8.3 Настройка маппинга фильтров

Как уже говорилось выше, фильтры — это служебная разновидность сервлетов, и настраиваются они подобно сервлетам. Вот какие атрибуты есть у аннотации `@WebFilter`:

	Атрибут	Описание
1	<code>filterName</code>	Уникальное имя фильтра (как в web.xml)
2	<code>displayName</code>	Читаемое имя фильтра для человека

3	description	Описание фильтра
4	value / urlPatterns	Задаёт список url-ов для мапинга
5	dispatcherTypes	Задаёт список типов DispatcherType
6	servletNames	Задаёт список сервлетов, к которым применяется
7	initParams	Позволяет задать стартовые параметры фильтра
8	asyncSupported	Указывает, что фильтр может работать асинхронно (HTTP/2)
9	smallIcon	Задаёт маленькую иконку фильтра
10	largeIcon	Задаёт большую иконку фильтра

Пример фильтра, который **перехватывает все запросы к определенным сервлетам**:

```
1 @WebFilter(servletNames = {"MyOwnServlet", "UploadServlet"})
2 public class MyFilter implements Filter {
3     // implements Filter's methods here...
4 }
```

Пример фильтра, которому **задано несколько параметров**:

```
1 @WebFilter(
2     urlPatterns = "/uploadFilter",
3     initParams = @WebInitParam(name = "fileTypes", value = "doc;xls;zip;text;jpg;png;gif")
4 )
5 public class UploadFilter implements Filter {
6     // implements Filter's methods here...
7 }
```

Пример фильтра, которому **заданы типы для RequestDispatcher**:

```
1 @WebFilter(
2     urlPatterns = "/admin",
3     dispatcherTypes = {DispatcherType.REQUEST, DispatcherType.FORWARD}
4 )
5 public class MyFilter implements Filter {
6     // implements Filter's methods here...
7 }
```

Более детально о назначении фильтров и сервлетов ты узнаешь в следующих лекциях.



Введите текст комментария



Руслан Шмаков Уровень 81 EXPERT 3 октября 2022, 08:35

...

Два раза прочитал, нифига не понял. Видимо, надо на практике самому руками пробовать

Ответить

− +1 +



Igor Уровень 85 EXPERT 8 сентября 2022, 19:51

...

В общем, про web.xml можно забыть и сосредоточиться на аннотациях :)

Ответить

− +4 +

Дмитрий Рыбин Уровень 41

2 октября 2022, 16:11

...

и забыть)

Ответить

− +1 +

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

