

```
<fmt:formatNumber value="9876543.21" type="percent"/><br/>
<fmt:setLocale value="ru-RU"/>
<jsp:text>Русская валюта - </jsp:text>
<fmt:formatNumber value="9876543.21" type="currency"/><br/>
<fmt:setLocale value="en-EN"/>
<jsp:text>Английская валюта - </jsp:text>
<fmt:formatNumber value="9876543.21" type="currency"/><br/>
<jsp:text>Французская валюта - </jsp:text>
<fmt:setLocale value="fr-FR"/>
<fmt:formatNumber value="9876543.21" type="currency"/><br/>
</body></html>
</jsp:root>
```

В результате работы документа в браузер будет выведено:

Вывод формата числа 9876543.21:

Обычный формат - 9 876 543,21

Процентный формат - 987 654 321%

Русская валюта - 9 876 543,21 руб.

Английская валюта - £9,876,543.21

Французская валюта - 9 876 543,21 €

JSTL sql

Используется для выполнения запросов SQL непосредственно из JSP и обработки результатов запроса в JSP.

```
<%@taglib uri="http://java.sun.com/jstl/sql" prefix="sql"%>
```

— для обычной страницы JSP;

```
<jsp:root version="1.2" xmlns:sql=
```

```
"http://java.sun.com/jstl/sql">...</jsp:root> — для JSP-документа.
```

Теги:

<sql:dateParam> — определяет параметры даты для **<sql:query>** либо **<sql:update>**;

<sql:param> — определяет параметры **<sql:query>** либо **<sql:update>**;

<sql:query> — выполняет запрос к БД;

<sql:setDataSource> — устанавливает data source для **<sql:query>**, **<sql:update>**, и **<sql:transaction>** тегов;

<sql:transaction> — объединяет внутренние теги **<sql:query>** и **<sql:update>** в одну транзакцию;

<sql:update> — выполняет преобразование БД.

В промышленном программировании данная библиотека не используется из-за прямого доступа из JSP в СУБД, что является явным нарушением шаблона MVC.

JSTL xml

Используется для обработки данных XML в JSP-документе.

```
<% @taglib uri="http://java.sun.com/jstl/xml" prefix="x" %>
```

— для обычной JSP-страницы,

```
<jsp:root version="1.2" xmlns:x=
```

```
"http://java.sun.com/jstl/xml">...</jsp:root> — для XML формата JSP.
```

Теги:

<x:forEach> – XML-версия тега **<c:choose>**;
<x:choose> – XML-версия тега **<c:forEach>**;
<x:if> – XML-версия тега **<c:if>**;
<x:otherwise> – XML-версия тега **<c:otherwise>**;
<x:out> – XML-версия тега **<c:out>**;
<x:param> – XML-версия тега **<c:param>**, определяющая параметры для другого тега **<x:transform>**;
<x:parse> – разбор XML-документа;
<x:set> – XML-версия тега **<c:set>**;
<x:transform> – трансформация XML-документа;
<x:when> – XML-версия тега **<c:when>**;
<x:choose> – XML-версия тега **<c:choose>**;
<x:forEach> – XML-версия тега **<c:forEach>**.

Включение ресурсов

В реальных проектах JSP-страницы часто состоят из статических элементов. Для этого используется директива **include**, а файл, содержащий необходимый статичный элемент, сохраняется с расширением **.jspx**, что означает «фрагмент JSP». При необходимости включения содержимого в JSP-страницу каждый раз, когда та получает запрос, используется стандартное действие **jsp:include**. В этом случае включаемые сегменты имеют доступ к объектам **request**, **session** и **application** исходной страницы и ко всем атрибутам, которые имеют эти объекты. Если использовать директиву **include**, то изменения включаемого сегмента отразятся только после изменения исходной страницы (контейнер JSP перекомпилирует исходную страницу). Для включения содержимого в JSP-документ также используется стандартное действие **jsp:include**. При этом не обязательно, чтобы включаемый JSP-фрагмент был правильным XML-документом. Главное, чтобы он возвращал текст в виде правильного XML и не нарушал структуру исходного JSP-документа.

```

<!--пример #15 : включение в код статического содержимого : incl_title.jspx -->
<jsp:root xmlns:jsp="http://java.sun.com/JSP/Page"
          xmlns:c="http://java.sun.com/jsp/jstl/core"
          version="2.0">
  <jsp:directive.page contentType=
    "text/html; charset=UTF-8" />
  <html><head>
    <jsp:directive.include file="\WEB-INF\jspf\title.jspf" />
  </head>
  <body>
    <h1>JSP-страница, использующая директиву include</h1>
    <h3>Директива include используется для включения статиче-
      ского содержимого, например заголовка страницы.</h3>
  </body></html>
</jsp:root>

```