

Chapter 7. Building JSP Pages Using the Expression Language (EL)

[Prev](#)

Part I. Exam Objectives

[Next](#)

Chapter 7. Building JSP Pages Using the Expression Language (EL)

Given a scenario, write EL code that accesses the following implicit variables including: `pageScope`, `requestScope`, `sessionScope`, and `applicationScope`, `param` and `paramValues`, `header` and `headerValues`, `cookie`, `initParam` and `pageContext`.

There are several implicit objects that are available to EL expressions used in JSP pages. These objects are always available under these names:

- `pageContext` - the `PageContext` object. Provides an API to access various objects including:
 - `context` - the context for the JSP page's servlet and any Web components contained in the same application.
 - `session` - the session object for the client.
 - `request` - the request triggering the execution of the JSP page.
- `pageScope` - a `java.util.Map` that maps page-scoped attribute names to their values.
- `requestScope` - a `java.util.Map` that maps request-scoped attribute names to their values.
- `sessionScope` - a `java.util.Map` that maps session-scoped attribute names to their values.
- `applicationScope` - a `java.util.Map` that maps application-scoped attribute names to their values.
- `param` - a `java.util.Map` that maps parameter names to a single `String` parameter value (obtained by calling `ServletRequest.getParameter(String name)`).
- `paramValues` - a `java.util.Map` that maps parameter names to a `String[]` of all values for that parameter (obtained by calling `ServletRequest.getParameterValues(String name)`).
- `header` - a `java.util.Map` that maps header names to a single `String` header value (obtained by calling `HttpServletRequest.getHeader(String name)`).
- `headerValues` - a `java.util.Map` that maps header names to a `String[]` of all values for that header.
- `cookie` - a `java.util.Map` that maps cookie names to a single `Cookie` object. Cookies are retrieved according to the semantics of `HttpServletRequest.getCookies()`. If the same name is shared by multiple cookies, an implementation must use the **FIRST** one encountered in the array of `Cookie` objects returned by the `getCookies()` method. However, users of the `cookie` implicit object must be aware that the ordering of cookies is currently unspecified in the servlet specification.
- `initParam` - a `java.util.Map` that maps context initialization parameter names to their `String` parameter value (obtained by calling `ServletContext.getInitParameter(String name)`).

Examples:

The request's URI (obtained from `HttpServletRequest`):

```
${pageContext.request.requestURI}
```

The value of the `numberOfItems` property of the session-scoped attribute named `cart`:

```
${sessionScope.cart.numberOfItems}
```

The context path:

```
${pageContext.request.contextPath}
```

The session-scoped attribute named 'profile' (null if not found):

```
${sessionScope.profile}
```

The String value of the productId parameter, or null if not found:

```
${param.productId}
```

The value of the productId request parameter:

```
${param["productId"]}
```

The String[] containing all values of the productId parameter, or null if not found:

```
${paramValues.productId}
```

A collection's members can be accessed using square brackets as shown by retrieval of the userName parameter from the param object. Members of an array or List can be accessed if the value in square brackets can be coerced to an int.

```
<html>
  <head><title>Customer Profile for ${param["userName"]}</title></head>
  <body>
    ...
  </body>
</html>
```

Maps can be accessed using the dot operator OR square brackets. For example, `${param.userName}` is EQUIVALENT to `${param["userName"]}`.

The host HTTP attribute:

```
${header["host"]}
```

Here is an example of accessing a page-scoped object that is called pageColor:

```
<body bgcolor="${pageScope.pageColor}">
```

it is equivalent to:

```
<body bgcolor="${pageScope['pageColor']}">
```

[Prev](#)

Given a specific design goal for including a JSP segment in another page, write the JSP code that uses the most appropriate inclusion mechanism (the include directive or the `jsp:include` standard action).



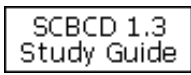
ICAD
Study Guide



SCJP 5.0
Study Guide

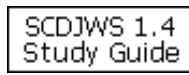


Magnet
Mocker

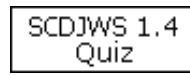
[Up](#)[Home](#)

SCBCD 1.3
Study Guide

IBM Test 287
Study Guide



SCDJWS 1.4
Study Guide



SCDJWS 1.4
Quiz

[Next](#)

Given a scenario, write EL code that uses the following operators: property access (the `.` operator), collection access (the `[]` operator).

webhosting provided by [ixvar inc](#)