

Introduction to Servlet

What is Servlet
 Servlet Terminology
 Servlet API
 Servlet Interface
 GenericServlet
 HttpServlet
 Servlet Life Cycle
 Servlet Example
 How servlet works?
 War File
 welcome-file-list
 Load on startup

Servlet with IDE

servlet in Eclipse
 servlet in Myeclipse
 servlet in Netbeans

ServletRequest**Servlet Collaboration**

RequestDispatcher
 sendRedirect

ServletConfig**ServletContext****Attribute in Servlet****Session Tracking**

Session Techniques
 1) Cookies in Servlet
 Cookies: Login & Logout
 2) Hidden Form Field
 3) URL Rewriting
 4) HttpSession
 Session: Login & Logout

Event and Listener**Servlet Filter**

What is Filter
 Authentication Filter
 FilterConfig
 Useful examples

Servlet Miscellaneous

ServletInputStream
 ServletOutputStream
 Annotation Servlet
 SingleThreadModel
 SSI

Development

Registration Example
 Fetching records
 Improving Performance
 Uploading file
 Downloading file
 Servlet Sending Email
 Write data to PDF
 Login Example
 Writing Image

ServletContext Interface

An object of ServletContext is created by the web container at time of deploying the project. This object can be used to get configuration information from web.xml file. There is only one ServletContext object per web application.

If any information is shared to many servlet, it is better to provide it from the web.xml file using the **<context-param>** element.

[ServletContext Interface](#)**[Usage of ServletContext Interface](#)****[Methods of ServletContext interface](#)****[How to get the object of ServletContext](#)****[Syntax to provide the initialization parameter in Context scope](#)****[Example of ServletContext to get initialization parameter](#)****[Example of ServletContext to get all the initialization parameter](#)**

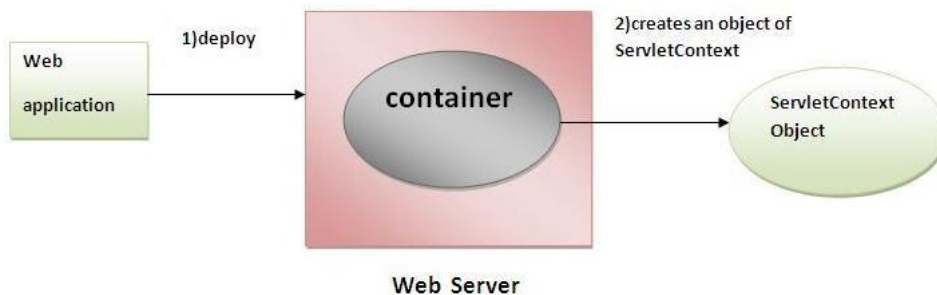
Advantage of ServletContext

Easy to maintain if any information is shared to all the servlet, it is better to make it available for all the servlet. We provide this information from the web.xml file, so if the information is changed, we don't need to modify the servlet. Thus it removes maintenance problem.

Usage of ServletContext Interface

There can be a lot of usage of ServletContext object. Some of them are as follows:

1. The object of ServletContext provides an interface between the container and servlet.
2. The ServletContext object can be used to get configuration information from the web.xml file.
3. The ServletContext object can be used to set, get or remove attribute from the web.xml file.
4. The ServletContext object can be used to provide inter-application communication.



Commonly used methods of ServletContext interface

There is given some commonly used methods of ServletContext interface.

1. **public String getInitParameter(String name):**Returns the parameter value for the specified parameter name.
2. **public Enumeration getInitParameterNames():**Returns the names of the context's initialization parameters.
3. **public void setAttribute(String name, Object object):**sets the given object in the application scope.
4. **public Object getAttribute(String name):**Returns the attribute for the specified name.
5. **public Enumeration getInitParameterNames():**Returns the names of the context's initialization parameters as an Enumeration of String objects.
6. **public void removeAttribute(String name):**Removes the attribute with the given name from the servlet context.

How to get the object of ServletContext interface

Servlet Quiz

Servlet Quiz(Part-1)

Servlet Quiz(Part-2)

Servlet Quiz(Part-3)

Interview Questions

1. **getServletContext()** method of ServletConfig interface returns the object of ServletContext.
2. **getServletContext()** method of GenericServlet class returns the object of ServletContext.

Syntax of getServletContext() method

```
public ServletContext getServletContext()
```

Example of getServletContext() method

```
//We can get the ServletContext object from ServletConfig object
ServletContext application=getServletConfig().getServletContext();

//Another convenient way to get the ServletContext object
ServletContext application=getServletContext();
```

Syntax to provide the initialization parameter in Context scope

The **context-param** element, subelement of web-app, is used to define the initialization parameter in the application scope. The param-name and param-value are the sub-elements of the context-param. The param-name element defines parameter name and param-value defines its value.

```
<web-app>
.....

<context-param>
  <param-name>parametername</param-name>
  <param-value>parametervalue</param-value>
</context-param>
.....
</web-app>
```

Example of ServletContext to get the initialization parameter

In this example, we are getting the initialization parameter from the web.xml file and printing the value of the initialization parameter. Notice that the object of ServletContext represents the application scope. So if we change the value of the parameter from the web.xml file, all the servlet classes will get the changed value. So we don't need to modify the servlet. So it is better to have the common information for most of the servlets in the web.xml file by context-param element. Let's see the simple example:

DemoServlet.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class DemoServlet extends HttpServlet{
    public void doGet(HttpServletRequest req,HttpServletResponse res)
    throws ServletException,IOException
    {
        res.setContentType("text/html");
        PrintWriter pw=res.getWriter();

        //creating ServletContext object
        ServletContext context=getServletContext();

        //Getting the value of the initialization parameter and printing it
        String driverName=context.getInitParameter("dname");
        pw.println("driver name is="+driverName);

        pw.close();
    }
}
```

web.xml

```
<web-app>

<servlet>
<servlet-name>sonoojaiswal</servlet-name>
<servlet-class>DemoServlet</servlet-class>
</servlet>

<context-param>
<param-name>dname</param-name>
<param-value>sun.jdbc.odbc.JdbcOdbcDriver</param-value>
</context-param>

<servlet-mapping>
<servlet-name>sonoojaiswal</servlet-name>
<url-pattern>/context</url-pattern>
</servlet-mapping>

</web-app>
```

Example of ServletContext to get all the initialization parameters

In this example, we are getting all the initialization parameter from the web.xml file. For getting all the parameters, we have used the `getInitParameterNames()` method in the servlet class.

DemoServlet.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class DemoServlet extends HttpServlet{
    public void doGet(HttpServletRequest req,HttpServletResponse res)
    throws ServletException,IOException
    {
        res.setContentType("text/html");
        PrintWriter out=res.getWriter();

        ServletContext context=getServletContext();
        Enumeration<String> e=context.getInitParameterNames();

        String str="";
        while(e.hasMoreElements()){
            str=e.nextElement();
            out.print("<br> "+context.getInitParameter(str));
        }
    }
}
```

web.xml

```
<web-app>

<servlet>
<servlet-name>sonoojaiswal</servlet-name>
<servlet-class>DemoServlet</servlet-class>
</servlet>

<context-param>
<param-name>dname</param-name>
<param-value>sun.jdbc.odbc.JdbcOdbcDriver</param-value>
</context-param>
```

```

<context-param>
<param-name>username</param-name>
<param-value>system</param-value>
</context-param>

<context-param>
<param-name>password</param-name>
<param-value>oracle</param-value>
</context-param>

<servlet-mapping>
<servlet-name>sonoojaiswal</servlet-name>
<url-pattern>/context</url-pattern>
</servlet-mapping>

</web-app>

```

Next Topic **Attribute**


<<prev

next>>


1

Tweet 199

+1


javatpoint.com
 Like

123,818 people like javatpoint.com.



Facebook social plugin

Like the www.javatpoint.com on facebook / subscribe to get latest updates



[Blog](#)



Tutorials

Core Java Tutorial
 Servlet Tutorial
 JSP Tutorial
 Mail API Tutorial
 Design Pattern Tutorial
 Struts 2.0 Tutorial
 Spring 3.0 Tutorial
 Hibernate Tutorial
 Android Tutorial
 JavaScript Tutorial
 SQL Tutorial
 C Tutorial
 AJAX Tutorial
 JUnit Tutorial

Forum

Core Java Ques.
 Servlet Ques.
 JSP Ques.
 Struts Ques.
 Spring Ques.
 Hibernate Ques.
 Android Ques.

SSS IT PVT LTD

Development
 Training
 SEO

Interview Ques.

Core Java
 Servlet
 JSP

Projects

Free Java Projects

Quizzes

Core Java quiz
 Servlet quiz
 JSP quiz

CONTACT US

Tel. : 0120-4256464
 Mob. : +91 9990449935
 Email : enquiry@sssit.org
 Address: Plot - 6, GT Road,
 Near MMX Mall,
 Mohan Nagar, Ghaziabad,
 201007, UP, India

© 2011-2013 Javatpoint.
 All Rights Reserved.



JAXB Tutorial
Maven Tutorial

Consultancy

Struts2 quiz
Android quiz
OCJP/SCWCD quiz