**Difference between ServletConfig and ServletContext in Java Servlet**

**ServletConfig** and **ServletContext**, both are objects created at the time of [servlet](https://www.geeksforgeeks.org/introduction-java-servlets/) initialization and used to provide some initial parameters or configuration information to the servlet. But, the difference lies in the fact that information shared by ServletConfig is for a specific servlet, while information shared by ServletContext is available for all servlets in the web application. **ServletConfig:**

* ServletConfig is an object containing some initial parameters or configuration information created by Servlet Container and passed to the servlet during initialization.
* ServletConfig is for a particular servlet, that means one should store servlet specific information in web.xml and retrieve them using this object.
* **Example:** Suppose, one is building a job portal and desires to share different email ids (which may get change over time) to recruiter and job applicant. So, he decides to write two servlets one for handling recruiter’s request and another one for the job applicant. **Where to store email-ids?** Put email-id as a name-value pair for different servlet inside web.xml which can further be retrieved using getServletConfig().getInitParameter(“name”) in the servlet.

**ServletContext:**

* ServletContext is the object created by Servlet Container to share initial parameters or configuration information to the whole application.
* **Example:** Suppose, the name of one’s job portal is “NewWebsite.tg”. Showing the website name at the top of webpages delivered by different servlets, one needs to store the website name in every servlet inviting redundancy. Since the information shared by ServletContext can be accessed by every Servlet, it is better to go with ServletContext and retrieve the website name using getServletContext.getInitParameter(“Name”) whenever required.

**Implementation of examples of ServletConfig and ServletContext is shown below.**

web.xml

* Servlet for recruiter
* Servlet for applicant

|  |
| --- |
| * Servlet for recruiter   **import** java.io.\*;  **import** javax.servlet.\*;  **import** javax.servlet.http.\*;    **public** **class** Recruiter **extends** HttpServlet {    **protected** **void** doGet(HttpServletRequest request,                           HttpServletResponse response)  **throws** ServletException, IOException      {          String email              = getServletConfig()                    .getInitParameter("Email");          String website              = getServletContext()                    .getInitParameter("Website-name");          PrintWriter out = response.getWriter();          out.println("<center><h1>" + website                      + "</h1></center><br><p>Contact us:"                      + email);      }  }   * Servlet for applicant   import java.io.\*;  import javax.servlet.ServletException;  import javax.servlet.http.\*;  public class Applicant extends HttpServlet {  protected void doGet(HttpServletRequest request,  HttpServletResponse response)  throws ServletException, IOException  {  String email  = getServletConfig()  .getInitParameter("Email");  String website  = getServletContext()  .getInitParameter("Website-name");  PrintWriter out = response.getWriter();  out.println("<center><h1>" + website  + "</h1></center><br><p>Contact us:"  + email);  }  } |

* **Output**: Deploy the app and open urls on localhost:
  1. **/servlet1**:[](https://media.geeksforgeeks.org/wp-content/uploads/20190724210033/recruiter.jpg)
  2. **/servlet2**:[](https://media.geeksforgeeks.org/wp-content/uploads/20190724210108/applicant.jpg)
* As shown above, different servlets get different email-id with same parameter name “Email” because of different values attached to it in the web.xml. Each servlet has got their own <init-param> inside <servlet> containing email-id.
* Both servlets got the access of ServletContext parameter “Website-name”.Since, <context-param> is outside of <servlet> in web.xml, hence it is independent of servlet and accessible from whole app.

**Below is the table of comparison between the two:**

| **ServletConfig** | **ServletContext** |
| --- | --- |
| [ServletConfig](https://www.geeksforgeeks.org/servletconfig-in-servlet/#:~:text=ServletConfig%20is%20an%20object%20containing,retrieve%20them%20using%20this%20object.) is servlet specific | [ServletContext](https://www.geeksforgeeks.org/servlet-context-event-and-context-listener/#:~:text=ServletContextEvent%20class%20provides%20alerts%2Fnotifications,context%20and%20acts%20on%20them.) is for whole application |
| Parameters of servletConfig are present as name-value pair in <init-param> inside <servlet>. | Parameters of servletContext are present as name-value pair in <context-param> which is outside of <servlet> and inside <web-app> |
| ServletConfig object is obtained by getServletConfig() method. | ServletContext object is obtained by getServletContext() method. |
| Each servlet has got its own ServletConfig object. | ServletContext object is only one and used by different servlets of the application. |
| Use ServletConfig when only one servlet needs information shared by it. | Use ServletContext when whole application needs information shared by it |