Number of ways to create a servlet

THREE Ways

1. Implements Servlet
2. Extends Generic Servlet
3. Extends HTTP Servlet 🡪 Recommended in real time

Implement Servlet

Servlet is an interface defined in javax.servlet

It has 5 different abstract methods.

1. init()
2. service()
3. destroy()
4. getServletConfig()
5. getServletInfo()

In order to create a servlet using this approach,

Step 1: implement our servlet class from Servlet

Step 2: override all 5 abstract methods

Example:

**package** com.servlet.approach1;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** javax.servlet.Servlet;

**import** javax.servlet.ServletConfig;

**import** javax.servlet.ServletException;

**import** javax.servlet.ServletRequest;

**import** javax.servlet.ServletResponse;

**public** **class** ImplementsServlet **implements** Servlet{

**public** **void** destroy() {

System.***out***.println("Destroy Method");

}

**public** ServletConfig getServletConfig() {

// **TODO** Auto-generated method stub

**return** **null**;

}

**public** String getServletInfo() {

// **TODO** Auto-generated method stub

**return** **this**.getClass().getName();

}

**public** **void** init(ServletConfig arg0) **throws** ServletException {

System.***out***.println("Web App is loading into Servlet container");

}

**public** **void** service(ServletRequest arg0, ServletResponse arg1) **throws** ServletException, IOException {

System.***out***.println("Servlet is ready to process the user request");

/\* How to display the message in Browser OR

\* How to generate a dynamic web page from servlet \*/

// Step 1: get the reference of PrintWriter from ServletResponse

PrintWriter pw = arg1.getWriter();

// Step 2: provide the information to render on UI

pw.println("Hi.....");

}

}

Approach – 2 Extends Generic Servlet

Generic servlet is predefined class in javax.servlet

This class implements

javax.servlet.Servlet,

javax.servlet.ServletConfig

we have the predefined implementations for all the FOUR methods available in servlet interface (1. Init, 2. Destroy 3. getServletConfig 4. getServletInfo)

but Service() method is left undefined. Because the user has to provide the business logic or servlet definition inside the service method

**package** com.servlet.approach2;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** javax.servlet.GenericServlet;

**import** javax.servlet.ServletException;

**import** javax.servlet.ServletRequest;

**import** javax.servlet.ServletResponse;

/\*\*

\* Servlet implementation class ExtendsGenericServlet

\*/

**public** **class** ExtendsGenericServlet **extends** GenericServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

\* **@see** GenericServlet#GenericServlet()

\*/

**public** ExtendsGenericServlet() {

**super**();

// **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** Servlet#service(ServletRequest request, ServletResponse response)

\*/

**public** **void** service(ServletRequest request, ServletResponse response) **throws** ServletException, IOException {

/\* How to display the message in Browser OR

\* How to generate a dynamic web page from servlet \*/

// Step 1: get the reference of PrintWriter from ServletResponse

PrintWriter pw = response.getWriter();

// Step 2: provide the information to render on UI

pw.println("<h1> Defined from Extends Generic Servlet </h1>");

}

}

Other methods available in Generic Servlet:

1. GenericServlet() 🡪 Constructor
2. init(ServletConfig)🡪 executes during server startup
3. service(ServletRequest, ServletResponse)

🡪 used to provide the definition of servlet or Business Logic

1. destroy() 🡪 executes during shutting down the server
2. getServletConfig()
3. getServletInfo()
4. getInitParameter(String)
5. getInitParameterNames()
6. getServletContext()
7. getServletName()
8. init()
9. log(String)
10. log(String, Throwable)

From the above list, the methods from 2 to 6 are inherited from Servlet interface

The methods from 7 to 10 are inherited from Servlet Config interface

Approach – 3 : extends from HttpServlet