## 第5章 Servlet技术

### 一、选择题

1．HttpServlet的子类要从HTTP请求中获得请求参数，应该调用哪个方法。（ ）

A．java B．javac C．javadoc D．javap

2．Servlet容器在启动Web应用时创建哪些对象（多选）（ ）。

A．ServletRequest对象 B． ServletContext对象

C．Filter对象 D． 所有的Servlet对象

3．以下那些对象由Servlet容器创建（多选）（ ）。

A．Servlet对象 B．ServletRequest对象

C．ServletConfig对象 D．ServletResponse对象

4．当响应正文为那种数据类型时，浏览器会让用户下载响应正文（ ）。

A．text/html B．text/plain

C．application/msword D．image/jpeg

5． 当Servlet容器销毁一个Servlet时，会销毁哪些对象（多选）（ ）。

A．Servlet对象 B．与 Servlet对象关联的ServletConfig对象

C．ServletContext对象 D．ServletRequest对象和ServletResponse对象

6．仔细查看下面的这段web配置：

<servlet>

<servlet-name>SimpleServlet</servlet-name>

<servlet-class>com.lyq.SimpleServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>SimpleServlet</servlet-name>

<url-pattern>/SimpleServlet</url-pattern>

</servlet-mapping>

JSP中使用哪个URL可是访问此配置中的Servlet ( )。

A．Servlet B．SimpleServlet

C．/SimpleServlet D．com.lyq.SimpleServlet

7． Servlet首先执行的方法是( )。

A．init() B．doGet() C．destroy() D．doPost()

8． Filter接口响应请求时运行的方法是( )。

A．doFilter() B．destroy() C．service() D．init()

9． Servlet的service()方法是作用是( )。

A．释放资源 B．处理客户端的请求

C．完成初始化工作 D．获取Servlet对象的配置信息

10． 开发Servlet程序通常继承哪个接口？( )

A．Servlet B．ServletConfig C．GenericServlet D．HttpServlet

### 二、填空题

1． Servlet的生命周期的三个阶段分别为\_\_\_\_\_\_\_\_\_\_\_\_\_阶段、\_\_\_\_\_\_\_\_\_\_\_\_阶段和\_\_\_\_\_\_\_\_\_\_\_\_\_阶段。

2．假设Form表单的提交方式为GET，那么在Servlet中调用的是\_\_\_\_\_\_\_\_\_\_\_\_\_方法。

3．Servlet在调用doGet()或doPose()方法之前，首先调用\_\_\_\_\_\_\_\_\_\_\_\_\_方法。

4．Filter接口的doFilter()方法的参数类型包括\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

5．HttpServlet类仍然是一个抽象类，实现了service()方法，并针对HTTP 1.1中定义的7种请求类型提供了相应的方法：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

6．通过<filter>标签声明一个过滤器对象。在此标签下包含3个常用子元素，分别为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

7．阅读下列代码：

public class MyServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

response.setCharacterEncoding("GBK");

PrintWriter out = response.getWriter();

out.println("<HTML>");

out.println(" <HEAD><TITLE>Servlet实例</TITLE></HEAD>");

out.println(" <BODY>");

out.print(" Servlet实例： ");

out.print(this.getClass());

out.println(" </BODY>");

out.println("</HTML>");

out.flush();

out.close();

}

}

请过浏览器将请求交给MyServlet处理，返回给浏览器的结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

8．创建一个Servlet监听：

public class MyFilter implements Filter{

public void destroy() {

System.out.println("执行destroy()");

}

public void doFilter(ServletRequest arg0, ServletResponse arg1,

FilterChain arg2) throws IOException, ServletException {

System.out.println("执行doFilter()");

arg2.doFilter(arg0, arg1);

}

public void init(FilterConfig arg0) throws ServletException {

System.out.println("执行init()");

}

}

当浏览该项目首页的时候，监听在控制台输出的内容为：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

9．Servlet过滤器是客户端与目标资源间的中间层组件，用于\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

10．仔细查看下边的过滤器代码片段：

public void doFilter(ServletRequest request, ServletResponse response,

FilterChain chain) throws IOException, ServletException {

chain.doFilter(request, response);

}

其中chain.doFilter(request, response);代码的作用是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。