**7.会话跟踪**

* HttpServletRequest应用
  + HttpServletRequest简介
    - Web客户端发送给Web服务器的HTTP请求消息可以分为三个部分：请求行、请求消息头、消息正文（也叫实体内容）。ServletAPI中定义了一个ServletRequest接口来作为获取客户端请求信息的基本接口，Servlet程序通过调用ServletRequest对象的方法可以获知客户端的请求信息，以及客户端和服务器的网络环境信息。
    - 公共接口类HttpServletRequest继承自ServletRequest
  + 获得请求行的相关信息
    - HTTP请求消息的请求行可以分为三个部分：请求方式、资源路径和HTTP协议版本，如下所示：GET /RequestURI?param1=a¶m2=b HTTP/1.1
    - getMethod()返回HTTP请求消息中的请求方式(如GET、POST)
    - getRequestURI()返回请求行中的资源名部分
    - getQueryString()返回请求行中的参数部分
  + 获取网络连接信息
    - ServletRequest接口中定义了若干获取客户端和服务器的网络连接信息和访问特性的方法。
    - getRemoteAddr()返回发出请求的客户机的IP地址getRemoteHost()返回发出请求的客户机的的完整主机名getRemotePort()返回发出请求的客户机所使用的网络接口的端口号 getLocalAddr()返回Web服务器上接收当前请求的网络接口的IP地址getLocalName()返回Web服务器上接收当前请求的网络接口的IP地址所对应的主机名getLocalPort()返回Web服务器上接收当前请求的网络接口的端口号getServerName()返回当前请求所指向的主机名getServerPort()返回当前请求所连接的服务器端口号getScheme()返回请求的协议名
  + 获取请求头信息
    - ServletRequest和HttpServletRequest接口中定义了若干用于读取HTTP请求消息中的头字段的方法，其中的getHeader方法是一个通用的方法，可用于读取所有的头字段。
    - getHeader()getHeaders()getHeaderNames()getIntHeader()getDateHeader()getContentType()getContentLength()getCharacterEncoding()
  + 获取请求参数
    - ServletRequest和HttpServletRequest接口中定义了getParameter等方法，这些方法能读取HTTP请求消息的请求行中的URL地址后的附加信息。
    - getParameter()返回某个指定名称的参数的值getParameterValues()返回同一个参数名所对应的所有参数值getParameterNames()返回一个包含请求消息中的所有参数名的Enumeration对象getParameterMap()将请求消息中的所有参数名和值装入进一个Map对象中返回
* HttpServletResponse应用
  + HttpServletResponse简介
    - 代表了对客户端的Http响应。它允许Servlet设置内容长度和回应的MIME类型，并且提供输出流ServletOutputStream sendRedirect(java.lang.String location)：把响应发送到另一页面或者Servlet进行处理setContentType(java.lang.String type)：设置响应的MIME类型setCharacterEncoding(java.lang.String charset)：设置响应的字符编码类型
  + 中文乱码
    - 要求页面提交方式使用post方法提交，html页面的编码格式统一为“gb2312”
    - HttpServletResponse的setCharacterEncoding方法用来设置响应的字符编码类型，其常与HttpServletRequest的setCharacterEncoding方法一起使用，用来解决中文乱码问题
* Servlet中跳转与转发
  + Forward和sendRedirect异同
    - forward只能将请求转发给同一个Web应用中的组件；sendRedirect可以定向到应用程序外的其他资源。
    - forward方法重定向后URL不会改变；sendRedirect方法的URL会改变。
    - forward方法在服务器端内部将请求转发给另一个资源，浏览器只知道发出请求并得到相应结果，并不知在服务器内部发生的转发行为。sendRedirect对浏览器的请求直接作出响应，响应的结果告诉浏览器重新发出对另外一个URL的访问请求
    - forward方法的调用者与被调用者之间共享相同的request、response对象；HttpservletResponse.sendRedirect方法调用者和被调用者使用各自的request、response对象。
    - forward适用于一次请求响应过程由Web程序内部的多个资源来协同完成，需要在同一个Web程序内部资源之间跳转，使用HttpServletRequest.setAttribute方法将预处理结果传递给下一个资源。sendRedirect告诉浏览器重新发出对另外一个URL的访问请求，此时HttpServletRequest.setAttribute方法就不能将预处理结果传递给下一个资源。
* Servlet上下文环境
  + ServletContext接口简介
    - ServletContext是Servlet与Servlet容器之间直接通信的接口。
      * 每个 Web 应用都有唯一的ServletContext对象，可以把ServletContext对象形象地理解为Web应用的总管家，同一个Web应用中的所有Servlet对象都共享一个总管家，Servlet对象们可通过这个总管家来访问容器中的各种资源
      * ServletContext以web的已知路径为根路径。比如，假定一个servlet上下文位于http://www.mycorp.com/catalog。以/catalog请求路径开头的所有请求，已知为上下文路径
      * ServletContext，所有用户共用一个。要放必须的、重要的、所有用户需要共享的、线程又是安全的一些信息
* ServletContext接口方法
  + 用于在Web应用范围内存取共享数据的方法
    - setAttribute(String name, java.lang.Object object)getAttribute(String name)getAttributeNames()removeAttribute(String name)
  + 访问当前Web应用的资源
    - getContextPath()getInitParameter(String name)getInitParameterNames()getServletContextName()getRequestDispatcher(String path)
  + 访问Servlet容器中的其他Web应用
    - getContext(String uripath)
  + 访问Servlet容器的相关信息
    - getMajorVersion()：返回Servlet容器支持的Java Servlet API的主版本号。getMinorVersion()：返回Servlet容器支持的Java Servlet API的次版本号。getServerInfo()：返回Servlet容器的名字和版本。
  + 访问服务器端的文件系统资源。
    - getRealPath(String path)getResource(String path)getResourceAsStream(String path)getMimeType(String file)
  + 输出日志。
    - log(String msg)：向Servlet的日志文件中写日志。
    - log(String message, java.lang.Throwable throwable)：向Servlet的日志文件中写错误日志，以及异常的堆栈信息