

Oracle资格认证:简单解决复杂的OracleIAS问题Oracle认证考试
PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/557/2021_2022_Oracle_E8_B5_84_E6_c102_557051.htm 做了一个小的系统辅助功能，可以周期性访问某个URL、执行某个SQL语句 or 执行某个系统命令。执行SQL语句和系统命令比较简单，这里不再详述，主要说一下访问某个URL。实际上JDK自身已有工具类用于创建HTTP请求，类名是：java.net.HttpURLConnection，但考虑到基础类通常比较粗糙，很多情况要自己考虑和处理，就转头去Google了下，发现果然有开源的工具包可以使用，几个工具包中以HttpClient较为常用，而且是apache的东东，于是决定采用HttpClient。从apache上down了包commons-httpclient-3.1.jar和commons-codec-1.3.jar两个包，后者是HttpClient依赖的包。帮助写的很好，即便是像我这样英文很烂，也能很快上手。

```
public boolean visitURL(String url)
{
    // Commons HttpClient 3.1
    HttpClient client = new
    HttpClient().GetMethod method = new GetMethod(url). //
    Provide custom retry handler is
    necessary
    method.getParams().setParameter(HttpMethodParams.RE
    TRY_HANDLER, new DefaultHttpClientRetryHandler(3,
    false)).boolean rs = false.
    try {
        // Execute the method.
        int statusCode =
        client.executeMethod(method).
        if (statusCode !=
        HttpStatus.SC_OK) {
            logger.error("Method failed: "
            method.getStatusLine()).
        }
        else {
            rs = true.
        }
    } catch (HttpException e) {
        logger.error("Fatal protocol violation: "
        e.getMessage()).
    } catch
    (IOException e) {
        logger.error("Fatal transport error: "
```

e.getMessage()).} finally {
// Release the
connection.method.releaseConnection().}return rs.} 本机Tomcat
下run一下，工作正常，随即丢到服务器（Oracle IAS环境）
上测试，程序应该出乎意料的报了个错。 09/03/16 19:03:43
java.lang.NoClassDefFoundError09/03/16 19:03:43 at
org.apache.commons.httpclient.HttpMethodBase.writeRequestLine
(HttpMethodBase.java:2015)09/03/16 19:03:43 at
org.apache.commons.httpclient.HttpMethodBase.writeRequest(Http
pMethodBase.java:1864)09/03/16 19:03:43 at
org.apache.commons.httpclient.HttpMethodBase.execute(HttpMet
hodBase.java:975)09/03/16 19:03:43 at
org.apache.commons.httpclient.HttpMethodDirector.executeWith
Retry(HttpMethodDirector.java:368)09/03/16 19:03:43 at
org.apache.commons.httpclient.HttpMethodDirector.executeMeth
od(HttpMethodDirector.java:164)09/03/16 19:03:43 at
org.apache.commons.httpclient.HttpClient.executeMethod(HttpCli
ent.java:437)09/03/16 19:03:43 at
org.apache.commons.httpclient.HttpClient.executeMethod(HttpCli
ent.java:324)09/03/16 19:03:43 at
com.zbht.util.TimerTaskManager.runURLTask(TimerTaskManager
.java:237)09/03/16 19:03:43 at
_system._timer__task._test._jspService(_test.java:182)09/03/16
19:03:43 at
com.orionserver.http.OrionHttpJspPage.service(OrionHttpJspPage.
java:59)09/03/16 19:03:43 at
oracle.jsp.runtimev2.JspPageTable.service(JspPageTable.java:462)09

/03/16 19:03:43 at
oracle.jsp.runtimev2.JspServlet.internalService(JspServlet.java:594)0
9/03/16 19:03:43 at
oracle.jsp.runtimev2.JspServlet.service(JspServlet.java:518)09/03/16
19:03:43 at
javax.servlet.http.HttpServlet.service(HttpServlet.java:856)09/03/16
19:03:43 at
com.evermind.server.http.ServletRequestDispatcher.invoke(Servlet
RequestDispatcher.java:713)09/03/16 19:03:43 at
com.evermind.server.http.ServletRequestDispatcher.forwardInternal
(ServletRequestDispatcher.java:370)09/03/16 19:03:43 at
com.evermind.server.http.HttpRequestHandler.doProcessRequest(
HttpRequestHandler.java:871)09/03/16 19:03:43 at
com.evermind.server.http.HttpRequestHandler.processRequest(Htt
pRequestHandler.java:453)09/03/16 19:03:43 at
com.evermind.server.http.AJPRequestHandler.run(AJPRequestHan
dler.java:302)09/03/16 19:03:43 at
com.evermind.server.http.AJPRequestHandler.run(AJPRequestHan
dler.java:190)09/03/16 19:03:43 at
oracle.oc4j.network.ServerSocketReadHandler\$SafeRunnable.run(S
erverSocketReadHandler.java:260)09/03/16 19:03:43 at
com.evermind.util.ReleasableResourcePooledExecutor\$MyWorker.
run(ReleasableResourcePooledExecutor.java:303)09/03/16 19:03:43
at java.lang.Thread.run(Thread.java:595) 错误信息看上去比较低
级：NoClassDefFoundError，类没找到，迅速了检查了一下本
机和服务器上的jar包是否相同，“一模一样”！这就奇怪了

。检查本机的开发环境，只添加了这两个jar，其他的都没有动过，又检查服务器的运行环境，一样没有变化。于是删掉本机开发环境下的这两个jar，问题浮出来了，类中对httpclient的7、8个引用中只有1个提示未找到指定的类，看来Oracle自己的某个包中已经包含某个较低版本的httpclient，jar包冲突的问题是件让人沮丧的事情，尝试解决这种问题会所耗费的时间也许是其他方法的N倍，无心恋战。其实此处要进行的操作很简单，就是访问指定的URL，根据返回的内容检查是否成功，HttpClient是完整模拟浏览器，考虑了很多种问题，使用起来反倒是复杂了，决定转用JDK的基础类：`java.net.HttpURLConnection` 事情出奇的顺利，空间里找到了之前写的一个方法，正好解决这个问题，以下是代码清单：

```
private boolean visitURL(String strUrl, String successFlag)
{
    boolean rs = false;
    HttpURLConnection jconn = null;
    ByteArrayOutputStream byteArrayOutputStream = new
        ByteArrayOutputStream().
    try {
        URL url = new URL(strUrl);
        jconn = (HttpURLConnection)
            url.openConnection();
        jconn.setDoOutput(true);
        jconn.setDoInput(true);
        jconn.connect();
        InputStream in = jconn.getInputStream();
        byte[] buf = new byte[4096];
        int bytesRead;
        while ((bytesRead = in.read(buf)) != -1)
            {
                byteArrayOutputStream.write(buf, 0, bytesRead);
            }
        String strRead = new
            String(byteArrayOutputStream.toByteArray());
        logger.debug(strRead);
        strRead =
            StringUtil.NVL(strRead).if(strRead.indexOf(successFlag) != -1)
    }
    catch (Exception e)
    {
        e.printStackTrace();
    }
    return rs;
}
```

```
{logger.info("Visit URL gt. success !").rs = true.}} catch  
(MalformedURLException e) {e.printStackTrace().} catch  
(IOException e) {e.printStackTrace().} finally {jconn.disconnect().  
try {byteArrayOutputStream.close().} catch (IOException e)  
{e.printStackTrace().}}return rs.}
```

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com