Atitit web系统返回操作结果规范法 Mvc框架自定义

Mvc框架自定义

返回不了std out，好像可以，

返回不了 log文件内容，太杂乱。。

专门使用一个output map记录返回结果吧，规章。。。

同时可以吧结果放入记录操作日志文件目录里面去，序列化json

/bookmarksHtmlEverythingIndexPrj/src/aOPtool/clruser.java

@Path("/api416")

**public** String get(HttpServletRequest req,HttpServletResponse resp) **throws** IOException, Exception {

String paramString=req.getParameter("param");

String[] args = StringUtils.*splitByWholeSeparator*(paramString, " ");

// clruser.closeEcho=true;

clruser.*main*(args);

MvcUtil.*outputHtml*(resp, FileUtils.*readFileToString*(**new** File(clruser.*outFile*)));

**return** "thing";

}

**package** aOPtool;

**import** java.io.File;

**import** java.io.IOException;

**import** java.util.List;

**import** java.util.Map;

**import** java.util.function.Consumer;

**import** javax.servlet.Servlet;

**import** javax.servlet.ServletConfig;

**import** javax.servlet.ServletException;

**import** javax.servlet.ServletRequest;

**import** javax.servlet.ServletResponse;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** org.apache.catalina.Context;

**import** org.apache.catalina.LifecycleException;

**import** org.apache.catalina.startup.Tomcat;

**import** org.apache.log4j.Logger;

**import** com.alibaba.fastjson.JSON;

**import** com.attilax.mvc.MethodObj;

**import** com.attilax.mvc.MvcServlet;

**import** com.attilax.mvc.MvcUtil;

**import** com.attilax.util.TomcatV4t416;

**import** com.google.common.collect.Lists;

// http://localhost:888/api416?param=唐唐云学堂 删除 18821766710 056060

// aOPtool.clrusrTomcatStart 5201

**public** **class** clrusrTomcatStart {

**final** **static** Logger ***logger*** = Logger.*getLogger*(preSvr\_adminPubScrpt2publishtool2.**class**);

// port basedir

**public** **static** **void** main(String[] args) **throws** ServletException, LifecycleException {

// get url out mapper

List list = Lists.*newArrayList*();

list.add(clruser.**class**);

Map<String, MethodObj> url\_method\_maps = MvcUtil.*get\_url\_out\_mapper*(list);

***logger***.info(url\_method\_maps);

// javax.servlet.ServletContext

// javax.servlet.ServletContext.getVirtualServerName

// 获得当前项目路径get cur prj path

// new File("webroot").getAbsolutePath(); //

// 项目中web目录名称，以前版本为WebRoot、webapp、webapps，现在为WebContent

// System.out.println(new File("").getAbsolutePath());

System.***out***.println(JSON.*toJSON*(args));

**int** port = Integer.*parseInt*(args[0].trim());

String contextPath = "/";

String baseDir = **new** File("").getAbsolutePath();// pathx.webAppPath\_jensyegeor();

System.***out***.println("app basedir:" + baseDir);

**try** {

baseDir=args[1].trim();

} **catch** (Exception e) {

// e.printStackTrace();

}

System.***out***.println("app basedir from args:" + baseDir);

TomcatV4t416 tomcatV3T33aTomcatV3T33 = **new** TomcatV4t416();

tomcatV3T33aTomcatV3T33.startTomcat(port, contextPath, baseDir,**new** Consumer<Map>() {

@Override

**public** **void** accept(Map mp) {

Context Context1=(Context) mp.get("Context");

Tomcat Tomcat1=(Tomcat) mp.get("Tomcat");

Tomcat1.*addServlet*(Context1, "servletName", **new** MvcServlet(url\_method\_maps));

Context1.addServletMappingDecoded("/\*","servletName");//配置servlet映射路径

}

});

// tomcatV3T33aTomcatV3T33.tomcat.ad

tomcatV3T33aTomcatV3T33.tomcat.getServer().await();

String heartbeat\_str = "--tomcat run. heartbeat\_str";

// Global.heartbeatRecycle(heartbeat\_str);

}

}