JFinal 2.0及之后的版本都自带的常用工具类DreamLu jfinal2015-09-22 2091

JFinal中自带了不少好用的工具类，方便我们使用，下面列举几个最常用的。

一、StrKit字符串处理

// 首字母变小写

StrKit.firstCharToLowerCase("ABC"); //aBC

// 首字母变大写

StrKit.firstCharToUpperCase("abc"); //Abc

// 字符串为 null 或者为 "" 时返回 true

StrKit.isBlank(String str);

// 字符串不为 null 而且不为 "" 时返回 true

StrKit.notBlank(String str);

// 字符串数组不为 null 而且不为 "" 时返回 true

StrKit.notBlank(String... str);

// 对象数组不为 null

StrKit.notNull(Object... paras);

// 为一个数据表项自动生成一个id (uuid: 通用唯一识别码 )

StrKit.getRandomUUID();

二、HashKit加密算法，默认UTF-8

// MD5加密算法加密

HashKit.md5(srcStr)

// SHA1加密

HashKit.sha1(srcStr)

// SHA256 散列

HashKit.sha256(srcStr)

// SHA384 散列

HashKit.sha384(srcStr)

// SHA512 散列

HashKit.sha512(srcStr)

三、JsonKit字符串处理

// 对象转化成json字符串

JsonKit.toJson(Object value);

// 对象转化成json字符串并指定深度，兼容2.1及之前的版本

JsonKit.toJson(Object value, int depth);

// 解析json字符串到某个model对象（这里是Blog对象）

JsonKit.parse(jsonString, Blog.class);

四、PathKit路径获取，列举了2个

// 获取WebRoot的路径

PathKit.getWebRootPath();

// 获取classes路径

PathKit.getRootClassPath();

五、PropKit配置文件读取

// PropKit 工具类用来操作外部配置文件

//一般在JFinalConfig配置常量中，加载主配置文件

PropKit.use("config.txt");

// 默认从主配置文件，在JFinal web项目中会从WebConfig中的配置文件中读取

loadPropertyFile(String fileName);

// 其他则默认从第一个use的配置文件中读取

PropKit.get(String key);

// 使用非主配置文件，use之后会缓存该配置文件

Prop prop = PropKit.use(String fileName);

// 从prop中获取参数

prop.get(String key);

// 使用配置文件，并卸载该文件的缓存

Prop propLess = PropKit.useless(String fileName);

六、HttpKit，http请求工具类

/\*\*

\* Send GET request

\*/

HttpKit.get(String url);

HttpKit.get(String url, Map<String, String> queryParas);

HttpKit.get(String url, Map<String, String> queryParas, Map<String, String> headers);

/\*\*

\* Send POST request

\*/

HttpKit.post(String url, String data);

HttpKit.post(String url, String data, Map<String, String> headers);

HttpKit.post(String url, Map<String, String> queryParas, String data);

HttpKit.post(String url, Map<String, String> queryParas, String data, Map<String, String> headers);

/\*\*

\* 从request中读取post过来的数据

\*/

HttpKit.readIncommingRequestData(HttpServletRequest request);

七、Ret返回值封装

1、创建Ret对象返回JSON

// 创建ret对象

Ret ret = Ret.create("userList", UserList);

// 放入键值对

ret.put("status", 200);

ret.put("message", "success");

// 传回JSON串

renderJson(ret.getData());

2、返回请求结果字串

// 请求失败返回值封装

ret.fail("msg", "请求失败");

// 请求成功返回值封装

ret.ok("msg", "请求成功");

// 直接返回请求结果JSON串

renderJson(ret);

八、FileKit文件工具类

//删除对应文件对象，若是文件，则直接删除，若不是文件，则删除所在文件夹

FileKit.delete(file);

九、HandlerKit 处理流工具类

//返回并渲染404页面

HandlerKit.renderError404(request, response, true);

//自定义view渲染404页面

HandlerKit.renderError404(view, request, response, true);

十、JavaKeyword JAVA关键字集合

//集合中添加关键字

JavaKeyword.me.addKeyword(keyword);

//集合中删除关键字

JavaKeyword.me.removeKeyword(keyword);

//检查集合中是否包含关键字

JavaKeyword.me.contains(keyword);

十一、LogKit 日志输出

/

//输出日志信息到控制台

LogKit.info("message");

//输出警告日志到控制台

LogKit.warn("warn to you.");

//输出错误日志到控制台

LogKit.error("you have an error.!");

十二、Prop 配置文件读取

//配置文件Prop对象，编码参数不加时默认编码UTF-8

Prop prop = new Prop("config.txt", "UTF-8");

//用Prop对象获取相关的值

prop.get("userName");

//根据key获取int型参数值，为空默认返回0

getInt(key, 0);

//判断是否配置文件有包含配置key

containsKey(key);