JavaTp开发使用文档

**框架开发人：lxs**

**联系方式：1227033198@qq.com**

## **框架说明**

### 框架开发目的：

* + 1. **本框架纯属个人爱好所开发，一是为了巩固学习的知识，二是为了做一款简单的框架来体验一下框架开发的过程，三是为了在即将到来的实习面试准备一个项目，争取更大的竞争力**
    2. **本框架致力于开发一个简单容易上手的微小型的框架，主要使用场景为需求不复杂的微小型网站**

### 说明

* + 1. **本框架为一站式的开发框架，个人将其简化为两层(java中一般分为三层或者更多)，**
    2. **本框架更多的是对基本的java代码的封装，所以框架还提供了原始的java对象，方便使用者灵活使用本框架，以满足更复杂的需求以及个人喜好**

## 框架使用

### Web层：

#### 控制器使用

* + - 1. **需要在Web.xml中配置Filter过滤器**

**filter-class为固定名称：**com.java.tp.JavaTpFilter

* + - 1. **需要在init-param中配置控制器包所在位置**

**param-name为controllerPackge**

**param-value为控制器所在包**

<filter>

<filter-name>f</filter-name>

<filter-class>com.java.tp.JavaTpFilter</filter-class>

<init-param>

<param-name>controllerPackge</param-name>

<param-value>com.test.controller</param-value>

</init-param>

</filter>

<filter-mapping>

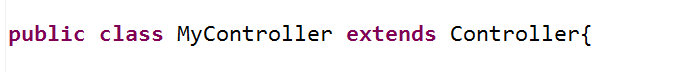
<filter-name>f</filter-name>

<url-pattern>/\*</url-pattern>

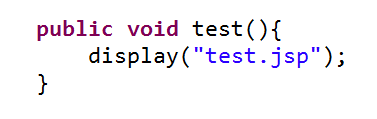
</filter-mapping>

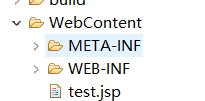
* + - 1. **控制器定义：**

**定义一个java类使它继承com.java.tp.Controller**



* + - 1. **控制器方法定义：** 
         1. **方法使用public void 方法名(){}格式书写业务逻辑**





* + - 1. **控制器访问URL**
         1. [**http://localhost:8080/项目名称/控制器名称/**](http://localhost:8080/项目名称/控制器名称/)**方法名名?key=value&key2=value...........**

**http://localhost:8080/JavaTp/MyController/test?id=1**



* + - 1. **内置方法、变量说明：**
         1. **void redirect(String url) 实现url跳转**
         2. **void display(String url) 转发至相关html或者jsp页面**
         3. **void alert(String message) 向客户浏览器弹出提示信息**
         4. **void alert(String message,String url) 向客户浏览器弹出提示信息，并跳转到指定页面**
         5. **void jsonReturn(Map<String, String> map) 返回json信息**
         6. **void HashMap<String, Object> getParam(HttpServletRequest request) 获取表单信息**
         7. **request: 可在控制器直接使用原生request**
         8. **response 可在控制器直接使用原生 response**
         9. **void assign(String key,Object value) 将数据放到request域**
         10. **void setSession(String key,Object value) 设置session**
         11. **void destorySession()销毁session**

### 数据库访问

#### 设置数据库配置文件db.properties

配置项：

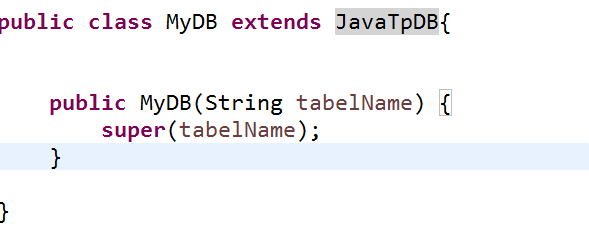
|  |
| --- |
| driverName=驱动名称  url=数据库url  user=用户名  password=数据库密码  connectionSize=数据库连接个数(默认20) |

|  |
| --- |
| driverName=com.mysql.jdbc.Driver  url=jdbc:mysql://localhost:3306/testdb?characterEncoding=utf-8&&useSSL=false&&useUnicode=true  user=root  password=root  connectionSize=20 |
|  |

#### 数据库使用:

1. 创建一个类继承com.java.tp.JavaTpDB
2. 创建构造函数
3. 使用数据库类

MyDB myDB=new MyDB("test\_news");



1. 数据库类相关方法

**void add(Map<String, Object>map) throws SQLException**

添加数据，map String为数据库字段，Objec为字段对应值

**void del(Map<String, Object>map) throws SQLException**

删除数据，map String为数据库字段，Objec为字段对应值

需要包含主键

**void update(Map<String, Object>map) throws SQLException**

更新数据，map String为数据库字段，Objec为字段对应值

需要包含主键

**Map<String, Object> load(Map<String, Object>map) throws SQLException**

通过条件查询单条数据，map String为数据库字段，Objec为字段对应值，传入的为查询的where条件

需要包含主键

**List<Map<String, Object>> query() throws SQLException**

**查询数据列表此方法可以与下面的方法一起使用**

JavaTpDB field(String ...strings)

设置字段

JavaTpDB where(String key,String operation,String value)

设置where条件，可以多次调用where设置多个条件

JavaTpDB having(String key,String operation,String value)

设置having条件

JavaTpDB OderByDesc(String field)

设置排序条件为降序

JavaTpDB OderByAsc(String field)

设置排序条件为升序

JavaTpDB limit(int begin)

JavaTpDB limit(int begin,int end)

以上两个设置分页

JavaTpDB group(String field)

设置分组条件

Field where having OderByDesc OderByAsc limit group支持连缀调用

public static Connection getConnection()

可以通过类名直接调用获取一个connection，进行底层的数据库操作

protected static void close(ResultSet rs,PreparedStatement ps,Connection con)

关闭数据库连接

protected static void close(PreparedStatement ps,Connection con)

关闭数据库连接

protected static void close(Connection con)

关闭数据库连接

