MiniDao 集成 Spring 文档

JEECG 开源社区

www.jeecg.org

2016/07/19

目　录

一、MiniDao-PE简介

MiniDao-PE 是一种持久化解决方案，类似 mybatis 的持久层解决方案，可以轻松集成入 Hibernate 工程，事务统一管理，解决了 Hibernate 工程想支类 mybaits 的功能问题。

具有以下特征:

- O/R mapping 不用设置 xml，零配置便于维护
- 不需要了解 JDBC 的知识
- SQL 语句和 java 代码的分离
- 接口和实现分离，不用写持久层代码，用户只需写接口，以及某些接口方法对应的 sql 它会通过 AOP 自动生成实现类
- 支持自动事务处理和手动事务处理
- 支持与 hibernate 轻量级无缝集成
- SQL 支持脚本语言

1. Minidao代码案例讲解

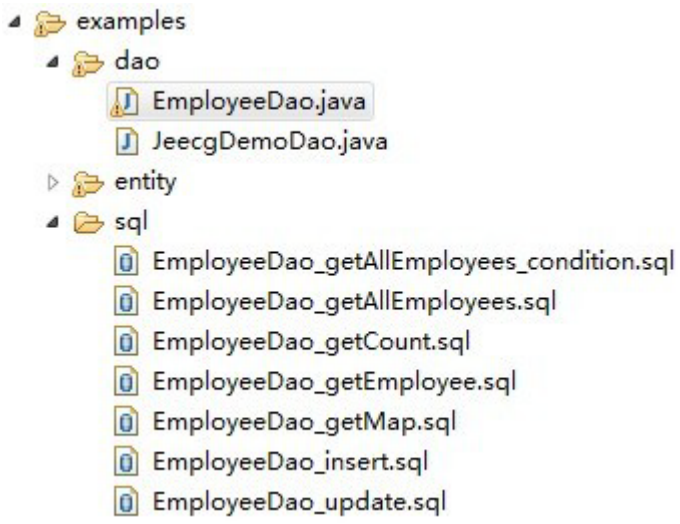
（1）接口定义[EmployeeDao.java]

```
@MiniDao
public interface EmployeeDao {

    /**
     * 返回List<Map>类型，全部数据
     * @param employee
     * @return
     */
    @Arguments({ "employee"})
    List<Map<String,Object>> getAll(Employee employee);

    /**
     * 查询返回Java对象
     * @param empno
     * @return
     */
    @Arguments("empno")
    Employee getEmployee(String empno);

    /**
     * 支持多个参数，查看返回Map
     * @param empno
     * @param name
     * @return
     */
    @Arguments({ "empno", "name"})
    Map<String,Object> getMap(String empno, String name);

    Map<String,Object> getMap2(@Param("empno") String empno,@Param("name")String name);

    /**
     * 修改数据
     * @param employee
     * @return
     */
```

（2）SQL文件[EmployeeDao_getAllEmployees.sql]

```
SELECT * FROM employee where 1=1
<#if employee.age ?exists>
and age = :employee.age
</#if>
<#if employee.name ?exists>
and name = :employee.name
</#if>
<#if employee.empno ?exists>
and empno = :employee.empno
</#if>
```

（3） minidao 接口定义和 SQL 文件对应目录



（4）MiniDao的spring配置

```
        <!-- 注册 MiniDao 接口 -->
<!-- MiniDao动态代理类 -->
    <bean id="miniDaoHandler"
class="org.jeecgframework.minidao.factory.MiniDaoBeanScannerConfigurer">
        <!-- 是使用什么字母做关键字Map的关键字 默认值origin 即和sql保持一致,lower小写(推荐),upper 大写 -->
        <property name="keyType" value="lower"></property>
        <!-- 格式化sql -->
        <property name="formatSql" value="false"></property>
        <!-- 输出sql -->
```

```
        <property name="showSql" value="false"></property>
        <!-- 数据库类型 -->
        <property name="dbType" value="mysql"></property>
        <!-- dao地址,配置符合spring方式 -->
        <property name="basePackage" value="org.jeecgframework.web,com.jeecg"></property>
        <!-- 使用的注解,默认是Minidao,推荐 Repository-->
        <property name="annotation"
value="org.springframework.stereotype.Repository"></property>
    </bean>
```

（5）测试代码

```
public class Client {
public static void main(String args[]) {
    BeanFactory factory = new ClassPathXmlApplicationContext(
            "applicationContext.xml");
    EmployeeDao employeeDao = (EmployeeDao) factory.getBean("employeeDao");
    Employee employee = new Employee();
    List<Map> list =  employeeDao.getAllEmployees(employee);
    for(Map mp:list){
        System.out.println(mp.get("id"));
        System.out.println(mp.get("name"));
        System.out.println(mp.get("empno"));
        System.out.println(mp.get("age"));
        System.out.println(mp.get("birthday"));
        System.out.println(mp.get("salary"));
    }
}
}
```

2．MiniDao与spring集成

（1）新建MiniDao的spring配置文件

文件名：spring-minidao.xml（可以自定义），只要让 spring 扫描到这个文件即可。

扫描方法一：

在 web.xml 中的 spring 监听器中扫描规则中包含 spring-minidao.xml。

```
<context-param>
  <param-name>contextConfigLocation</param-name>
  <param-value>classpath:spring-*.xml</param-value>
  </context-param>
```

扫描方法二：

在 spring 的配置文件中引入 **spring-minidao.xml。**

```
<import resource="classpath*:spring-minidao.xml" />
```

（2）MiniDao配置文件

```
<!-- MiniDao动态代理类 -->
    <bean id="miniDaoHandler" class="org.jeecgframework.minidao.factory.MiniDaoBeanScannerConfigurer">
        <!-- 是使用什么字母做关键字Map的关键字 默认值origin 即和sql保持一致,lower小写(推荐),upper 大写 -->
        <property name="keyType" value="lower"></property>
        <!-- 格式化sql -->
        <property name="formatSql" value="false"></property>
        <!-- 输出sql -->
        <property name="showSql" value="false"></property>
        <!-- 数据库类型 -->
        <property name="dbType" value="mysql"></property>
        <!-- dao地址,配置符合spring方式 -->
        <property name="basePackage" value="org.jeecgframework.web,com.jeecg"></property>
        <!-- 使用的注解,默认是Minidao,推荐 Repository-->
        <property name="annotation" value="org.springframework.stereotype.Repository"></property>
    </bean>
```

（3）MiniDao依赖maven引入

```
<dependency>
    <groupId>org.jeecgframework</groupId>
    <artifactId>minidao-pe</artifactId>
    <version>1.6-SNAPSHOT</version>
    <exclusions>
</dependency>
```

二、技术交流

- 作者： 张代浩
- 技术论坛：www.jeecg.org
- 邮箱： jeecg@sina.com
- 交流群：325978980，143858350
- 源码下载：http://git.oschina.net/jeecg/minidao-pe