一对多关联映射

第一步：创建pojo—->Orders.java

package com.yb.entity;

public class Orderdetail {

private int id;

private int ordersId;

private int itemsId;

private int itemsNum;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public int getOrdersId() {

return ordersId;

}

public void setOrdersId(int ordersId) {

this.ordersId = ordersId;

}

public int getItemsId() {

return itemsId;

}

public void setItemsId(int itemsId) {

this.itemsId = itemsId;

}

public int getItemsNum() {

return itemsNum;

}

public void setItemsNum(int itemsNum) {

this.itemsNum = itemsNum;

}

}

第二步：配置OrdersMapperCustom.xml

<resultMap type="com.yb.entity.Orders" id="OrdersUserResultMap">

<id column="id" property="id"/>

<result column="user\_id" property="userId"/>

<result column="number" property="number"/>

<result column="createtime" property="createtime"/>

<result column="note" property="note"/>

<association property="user" javaType="com.yb.entity.User">

<id column="user\_id" property="id"/>

<result column="username" property="username"/>

<result column="sex" property="sex"/>

<result column="address" property="address"/>

</association>

</resultMap>-->

<resultMap type="com.yb.entity.Orders" id="OrdersAndOrderDetailResultMap"

extends="OrdersUserResultMap"><!--继承上面的resultMap-->

<!-- <collection property="orderdetails" ofType="com.yb.entity.Orderdetail">

<id column="orderdetail\_id" property="id"/>

<result column="items\_id" property="itemsId"/>

<result column="items\_num" property="itemsNum"/>

<result column="orders\_id" property="ordersId"/>

</collection>

</resultMap>

<select id="findOrdersAndOrderDetailResultMap" resultMap="OrdersAndOrderDetailResultMap">

SELECT

orders.\* ,

USER.username,

USER.sex,

USER.address,

orderdetail.id orderdetail\_id,

orderdetail.orders\_id,

orderdetail.items\_id,

orderdetail.items\_num

FROM

orders,

USER,

orderdetail

WHERE

orders.user\_id=user.id AND orderdetail.orders\_id=orders.id

</select>

第三步：在接口OrdersMapperCustom.java中添加

public List<Orders>findOrdersAndOrderDetailResultMap()；

第四步：测试

@Test

public void testFindOrdersAndOrderDetailResultMap() throws Exception {

InputStream inputStream = Resources.getResourceAsStream("mybatis-config.xml");

SqlSessionFactory sqlSessionFactory = new SqlSessionFactoryBuilder().build(inputStream);

//得到会话sqlSession

SqlSession sqlSession=sqlSessionFactory.openSession();

//创建代理对象

OrderCustomMapper mapper = sqlSession.getMapper(OrderCustomMapper.class);

//调用mapper的方法,返回映射在pojo 中Orders(里面补充了Orderdetail orderDetails)

List<Orders>list=mapper.findOrdersAndOrderDetailResultMap();

System.out.println(list);

sqlSession.close();

}