

极客学院
jikexueyuan.com

MyBatis动态SQL

MyBatis动态SQL— 课程概要

- if、choose标记
- where、set和trim标记
- foreach标记

if、choose标记

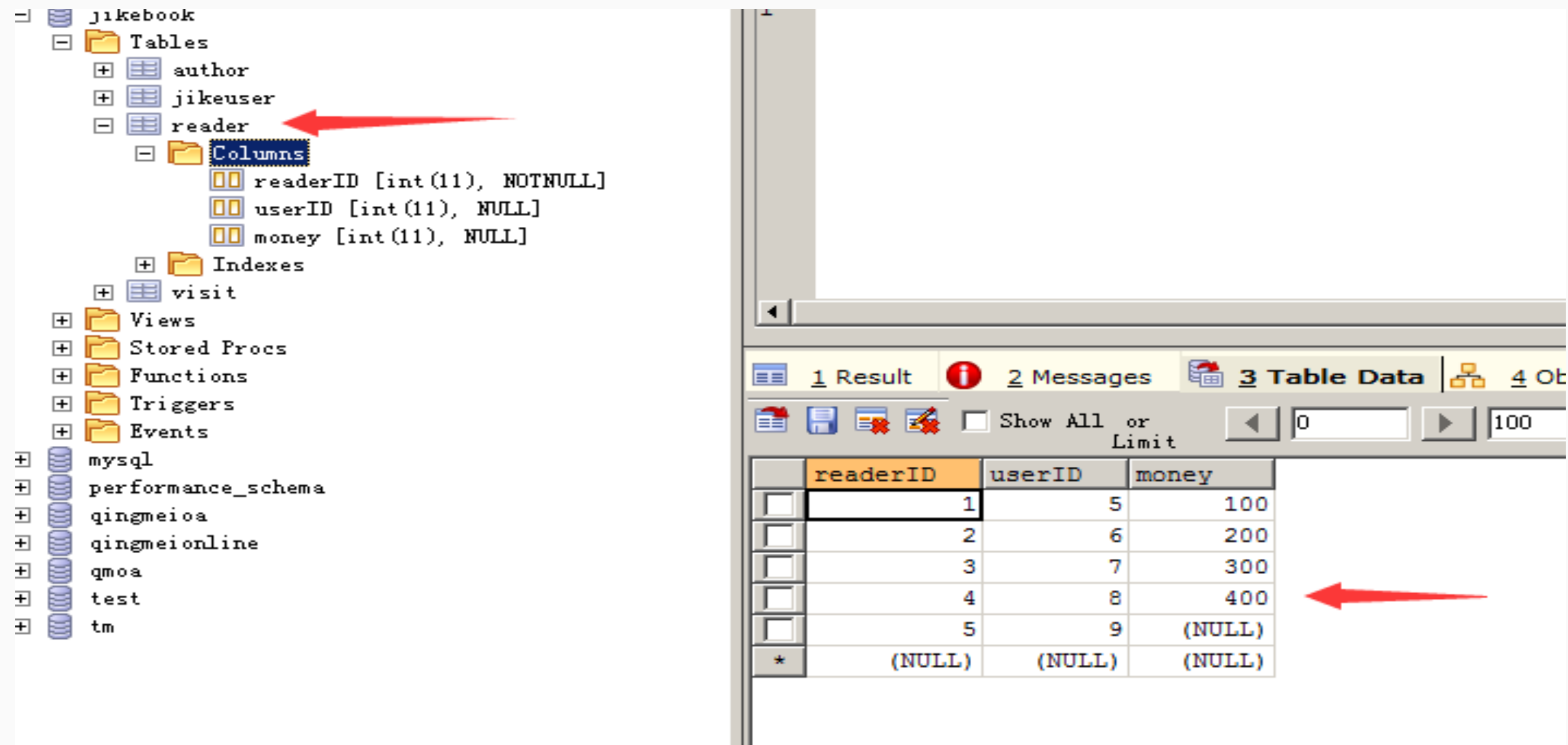
if、choose标记

本课时包含以下知识点：

- if标记
- choose标记

if、choose标记 — if标记

- 数据库reader表
- 判断金币大于某值的会员号，如果没有金币值就是全体会员



The screenshot displays a MySQL database interface. On the left, a tree view shows the database structure, with the 'reader' table under the 'jikebook' database selected. A red arrow points to the 'reader' table. The table's columns are listed as follows:

Column	Type	Null
readerID	int(11)	NOTNULL
userID	int(11)	NULL
money	int(11)	NULL

On the right, the 'Table Data' tab is active, showing the data for the 'reader' table. A red arrow points to the 'money' column. The data is as follows:

readerID	userID	money
1	5	100
2	6	200
3	7	300
4	8	400
5	9	(NULL)
*	(NULL)	(NULL)

if、choose标记 — if标记

1、建立select

```
<select id="selectReaderMoney" resultType="jikeReader"
    parameterType="jikeReader">
    select * from reader
        where 1=1
        <if test="money!=null">
            and money>#{money}
        </if>
</select>
```

if、choose标记 — if标记

2、测试

```
jikeReader oneReader=new jikeReader();  
oneReader.setMoney(200); //去除此行就是全体读者  
List<jikeReader>  
ap=session.selectList("selectReaderMoney",oneReader);
```

if、choose标记 — **choose**标记

1、建立select，判断用户名，如果不存在就判断id，最后判断密码不为空

```
select * from jikeuser where 1=1
<choose>
  <when test="userName!=null">
    and userName like #{userName}
  </when>
  .....
  <otherwise>
    and password is not null
  </otherwise>
</choose>
```


if、choose标记 — **choose**标记

2、测试

```
JiKeUser oneUser=new JiKeUser();  
oneUser.setUserName("%j%");  
oneUser.setId(5);  
List<JiKeUser> ap=session.selectList("selectJiKeUserChoose",oneUser);  
.....
```

where、set和trim标记

where、set和trim标记

本课时包含以下知识点：

- where标记
- set标记
- trim标记

where、set和trim标记 — **where**标记

1、where 标记=智能条件

2、改写以前查询语句

```
select * from jikeuser
<where>
  <if test="userName!=null">
    userName like #{userName}
  </if>
  <if test="id!=null">
    and id =#{id}
  </if>
</where>
```

where、set和trim标记 — **set**标记

- set 智能赋值，用JiKeUser表赋值
- 建立update

```
update JiKeUser
```

```
<set>
```

```
    <if test="userName!= null">userName=#{userName},</if>
```

```
    <if test="password != null">password=#{password},</if>
```

```
</set>
```

```
where id=#{id}
```

where、set和trim标记 — trim标记

- trim标记=格式化标记
- 可以与其它标记组合完成where与set标记功能

prefix	前缀增加
suffix	后缀增加
prefixOverrides	自动判断前置
suffixOverrides	自动判断后置

where、set和trim标记 — trim标记

- 代替set

```
<trim prefix="SET"    suffix="WHERE id = #{id}"
      suffixOverrides=", ">
    <if test="userName != null and userName != '' ">
        userName = #{userName},
    </if>
    .....
</trim>
```

Where、set和trim标记 — trim标记

- 代替where

```
select * from jikeuser
    <trim prefix="where"    prefixOverrides="and|or">
    <if test="userName!=null">
        and userName like #{userName}
    </if>
    <if test="id!=null">
        and id =#{id}
    </if>
</trim>
```


foreach标记

foreach标记

本课时包含以下知识点：

- 循环查询
- 循环赋值

foreach标记 — 循环查询

```
<foreach item="item" index="index" collection="list"
  open="(" separator="," close=")">
  #{item}
</foreach>
```

属性	描述
item	每一次迭代结果
collection	循环集合或指定类型
separator	元素之间的分隔符，可选。
open	开始符号，可选。
close	关闭符号，可选。
index	list和数组的序号可选。

foreach标记 — 循环查询

1、建立Select，从指定id集合中查询出会员

```
select * from jikeuser
<where>
    id in
    <foreach item="item" index="index" collection="list"
        open="(" separator="," close=")">
        #{item}
    </foreach>
</where>
```

foreach标记 — 循环查询

2、测试

```
ArrayList<Integer> ides=new ArrayList();  
ides.add(2);  
.....  
List<JiKeUser> ap=session.selectList("selectJiKeUserForeach", ides);
```

foreach标记 — 循环赋值

1、insert中的foreach,使用集合一次增加多个用户

```
insert into jikeUser (userName, password) values
<foreach item="item" index="key" collection="list"
    open="" separator="," close="">({item.userName},
    {item.password})
</foreach>
```

foreach标记 — 循环赋值

2、测试

```
ArrayList<JiKeUser> jkuList=new ArrayList();  
.....  
session.insert("insertJiKeUserForeach",jkuList);  
session.commit();
```

MyBatis动态SQL

这本套课程中我们学习了MyBatis动态SQL。你应当掌握了以下知识：

- MyBatis动态SQL标记 <if>
- MyBatis动态SQL标记 <choose>
- MyBatis动态SQL标记 <when>
- MyBatis动态SQL标记 <set>
- MyBatis动态SQL标记<trim>
- MyBatis动态SQL标记 <foreach>

你可以使用这些技巧来开发以MyBatis做为底层数据访问的相关工程，如果想继续提高，你可以继续在极客学院学习Java课程。

极客学院

jikexueyuan.com

中国最大的IT职业在线教育平台

