**今天完成的事情：**

**第一部分：JAVA基础知识**

**学习预览：**

* Mybatis的MySQL动态语句

**第二部分：修真任务**

**学习预览：**

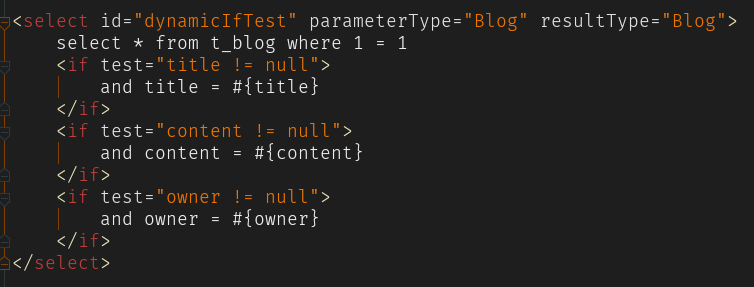
**第一部分**

1. **Mybatis的MySQL动态语句**

Mybatis的动态MySQL是基于OGNL（Object-Graph Navigation Language）表达式来完成的，动态MySQL可以帮助实现一些逻辑操作。

1. **if**

if就是简单的条件判断，利用if语句我们可以实现某些简单的条件选择

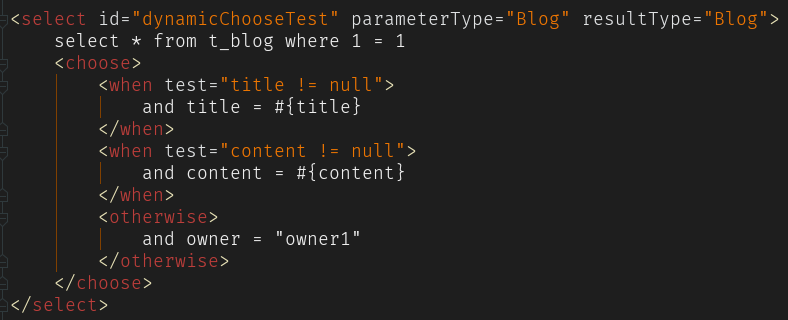


**语句分析：**

* 如果提供了title参数，那么就要满足title=#{title}
* 如果提供了Content和Owner的时候，也需要满足相应的条件，之后就是返回满足这些条件的所有Blog

1. **choose**

choose元素的作用就相当于JAVA中的switch语句，通常都是与when和otherwise搭配



**逻辑分析：**

* when元素表示当when中的条件满足的时候就输出其中的内容，按照条件的顺序执行
* 当when中有条件满足的时候，就会跳出choose（即所有的when和otherwise条件中，只有一个会输出）
* 当所有条件都不满足的时候，就输出otherwise中的内容

**语句分析：**

* 当title!=null的时候就输出and titlte = #{title}，不再往下判断条件
* 当title为空且content!=null的时候就输出and content = #{content}
* 当所有条件都不满足的时候就输出otherwise中的内容

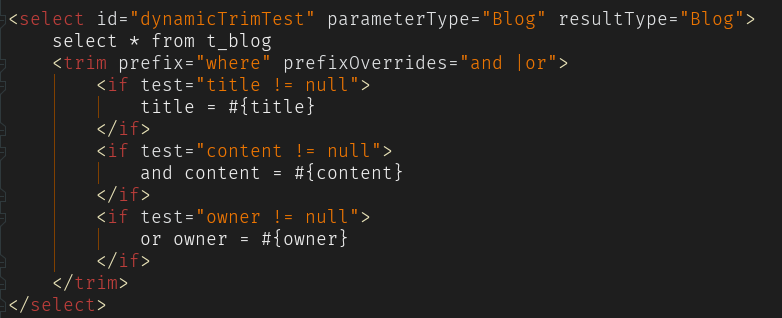
1. **trim**

trim元素的主要功能：

* 可以在自己包含的内容前加上某些前缀或后缀（prefix和suffix）
* 可以把包含内容的首部或尾部的某些内容覆盖

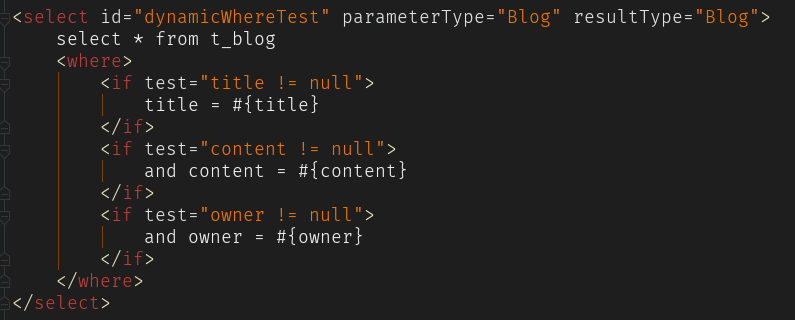
（prefixOverrides和suffixOverrides）

可以利用trim来代替where的功能



1. **where**

where语句的作用主要是简化SQL语句中where中的条件判断



**语句分析：**

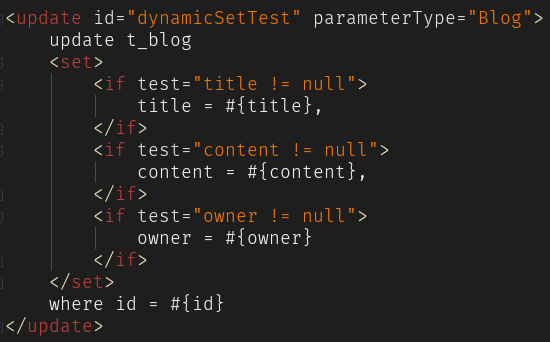
* 如果title=null， 而content != null
* 那么输出的整个语句会是select \* from t\_blog where content = #{content}
* 而不是select \* from t\_blog where and content = #{content}
* MyBatis会智能的把首个and 或 or 给忽略

1. **set**

set元素主要是用在更新操作的时候，在包含的语句前输出一个set，然后：

* 如果包含的语句是以逗号结束的话将会把该逗号忽略
* 如果set包含的内容为空的话则会出错

有了set元素我们就可以动态的更新那些修改了的字段



1. **for each**

foreach的主要用在构建in条件中，它可以在SQL语句中进行迭代一个集合

**foreach元素的属性主要有：**

* **item**

item表示集合中每一个元素进行迭代时的别名

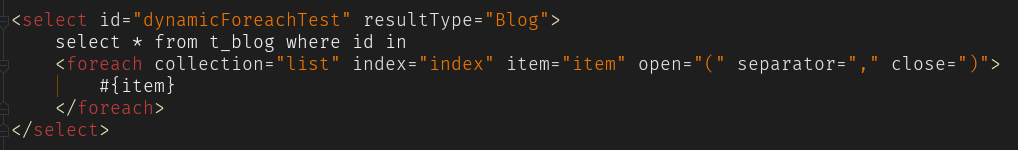
* **index**

index指定一个名字，用于表示在迭代过程中，每次迭代到的位置

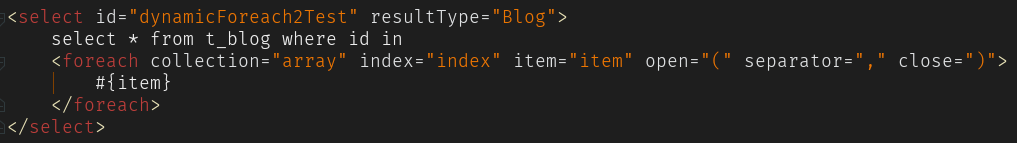
* **collection**

**该属性是必须指定的，但在不同情况下属性的值是不一样的**：

1. 如果传入的是单参数且参数类型是一个List的时候，collection属性值为list



1. 如果传入的是单参数且参数类型是一个array数组的时候，collection的属性值为array



1. 如果传入的是多参数，我们就需要把它们封装成一个Map了，当然单参数也可以封装成map



* **open**

open表示该语句以什么开始

* **separator**

separator表示在每次进行迭代之间以什么符号作为分隔符

* **close**

close表示以什么结束

**明天计划完成的事情：**

学习Junit的单元测试方法，用批量插入的方法往数据库插入用户数据，顺便完成批量修改操作

**遇到的问题：**

MySQL的批量插入修改方法很多样，也有很多优化方法，今天没有学习完，明天继续

**收获：**

学习了MySQL批量操作的方法，以及动态MySQL的方法