做条件判断的,如果我们不使用这个标签,我们肯定会在代码中判断如查询的元素是否为 空,传入的元素是否为空,而这时我们直接使用这个标签,就减少了代码的书写 概念 🖯 动态 SQL 通常要做的事情是有条件地包含 where 子句的一部分 <select id="findActiveBlogWithTitleLike" resultType="Blog"> SELECT * FROM BLOG WHERE state = 'ACTIVE' <if test="title != null"> AND title like #{title} </if> </select> 例子 ⊝ 如果没有传入"title",那么所有处于"ACTIVE"状态的BLOG都会返回;反 之若传入了 "title" , 那么就会把模糊查找 "title" 内容的BLOG结果返回 对于这类标签,就是采用多个选项中找一个,就像单项选择题,但是你不会都 选择,只会从中选择1个来作为条件。就有点类似于switch。。case。 <select id="findActiveBlogLike" resultType="Blog"> SELECT * FROM BLOG WHERE state = 'ACTIVE' <choose> <when test="title != null"> AND title like #{title} </when> <when test="author!= null and author.name!= null"> choose (when, otherwise) AND author name like #{author.name} ○20、动态 SQL </when> <otherwise> AND featured = 1 微信公众号java-mindmap </otherwise> </choose> 例子 😑 </select> 概念 如果使用第一个if语句的话,就会发现没有写where标签就会报错。而这类标签通常是搭配条件标签使用的。 trim (where, set) where一般可以用1=1来解决 毫无疑问这个标签肯定是用于循环了,用于遍历,如果我们传入的参数是一个数组或者集 合类,那么这个标签可以循环遍历。一般我们都是使用sql中的in语句时才使用。 概念 <select id="selectPostIn" resultType="domain.blog.Post"> SELECT * FROM POST P WHERE ID in <foreach item="item" index="index" collection="list" open="(" separator="," close=")"> #{item} foreach </foreach> 例子 </select> 你可以将任何可迭代对象(如列表、集合等)和任何的字典或者数组对象传递给foreach作为集合参数 注意 当使用可迭代对象或者数组时,index是当前迭代的次数,item的值是本次迭代获取的元素。