

select

查询语句是 MyBatis 中最常用的元素之一——光能把数据存到数据库中价值并不大，还要能重新取出来才有用，多数应用也都是查询比修改要频繁。MyBatis 的基本原则之一是：在每个插入、更新或删除操作之间，通常会执行多个查询操作。因此，MyBatis 在查询和结果映射做了相当多的改进。一个简单查询的 `select` 元素是非常简单的。比如：

```
<select id="selectPerson" parameterType="int" resultType="hashmap">
    SELECT * FROM PERSON WHERE ID = #{id}
</select>
```

这个语句名为 `selectPerson`，接受一个 `int`（或 `Integer`）类型的参数，并返回一个 `HashMap` 类型的对象，其中的键是列名，值便是结果行中的对应值。

注意参数符号：

```
#{id}
```

这就告诉 MyBatis 创建一个预处理语句（`PreparedStatement`）参数，在 JDBC 中，这样的参数在 SQL 中会由一个“?”来标识，并被传递到一个新的预处理语句中，就像这样：

```
// 近似的 JDBC 代码，非 MyBatis 代码...

String selectPerson = "SELECT * FROM PERSON WHERE ID=?";

PreparedStatement ps = conn.prepareStatement(selectPerson);

ps.setInt(1, id);
```

当然，使用 JDBC 就意味着使用更多的代码，以便提取结果并将它们映射到对象实例中，而这正是 MyBatis 的拿手好戏。参数和结果映射的细节会在后面单独的小节中说明。

`select` 元素允许你配置很多属性来配置每条语句的行为细节。

```
<select
    id="selectPerson"
    parameterType="int"
    parameterMap="deprecated"
    resultType="hashmap"
```

```

resultMap="personResultMap"

flushCache="false"

useCache="true"

timeout="10"

fetchSize="256"

statementType="PREPARED"

resultSetType="FORWARD_ONLY">
    
```

Select 元素的属性

属性	描述
id	在命名空间中唯一的标识符，可以被用来引用这条语句。
parameterType	将会传入这条语句的参数的类全限定名或别名。这个属性是可选的，因为 MyBatis 可以通过类型处理器（TypeHandler）推断出具体传入语句的参数，默认值为未设置（unset）。
parameterMap	用于引用外部 parameterMap 的属性，目前已被废弃。请使用行内参数映射和 parameterType 属性。
resultType	期望从这条语句中返回结果的类全限定名或别名。注意，如果返回的是集合，那应该设置为集合包含的类型，而不是集合本身的类型。 resultType 和 resultMap 之间只能同时使用一个。
resultMap	对外部 resultMap 的命名引用。结果映射是 MyBatis 最强大的特性，如果你对其理解透彻，许多复杂的映射问题都能迎刃而解。 resultType 和 resultMap 之间只能同时使用一个。
flushCache	将其设置为 true 后，只要语句被调用，都会导致本地缓存和二级缓存被清空，默认值： false 。
useCache	将其设置为 true 后，将会导致本条语句的结果被二级缓存缓存起来，默认值：对 select 元素为 true 。
timeout	这个设置是在抛出异常之前，驱动程序等待数据库返回请求结果的秒数。默认值为未设置（unset）（依赖数据库驱动）。

fetchSize	这是一个给驱动的建议值，尝试让驱动程序每次批量返回的结果行数等于这个设置值。默认值为未设置（unset）（依赖驱动）。
statementType	可选 STATEMENT, PREPARED 或 CALLABLE。这会让 MyBatis 分别使用 Statement, PreparedStatement 或 CallableStatement，默认值：PREPARED。
resultSetType	FORWARD_ONLY, SCROLL_SENSITIVE, SCROLL_INSENSITIVE 或 DEFAULT（等价于 unset）中的一个，默认值为 unset（依赖数据库驱动）。
databaseId	如果配置了数据库厂商标识（databaseIdProvider），MyBatis 会加载所有不带 databaseId 或匹配当前 databaseId 的语句；如果带和不带的语句都有，则不带的会被忽略。
resultOrdered	这个设置仅针对嵌套结果 select 语句：如果为 true，将会假设包含了嵌套结果集或是分组，当返回一个主结果行时，就不会产生对前面结果集的引用。这就使得在获取嵌套结果集的时候不至于内存不够用。默认值：false。
resultSets	这个设置仅适用于多结果集的情况。它将列出语句执行后返回的结果集并赋予每个结果集一个名称，多个名称之间以逗号分隔。