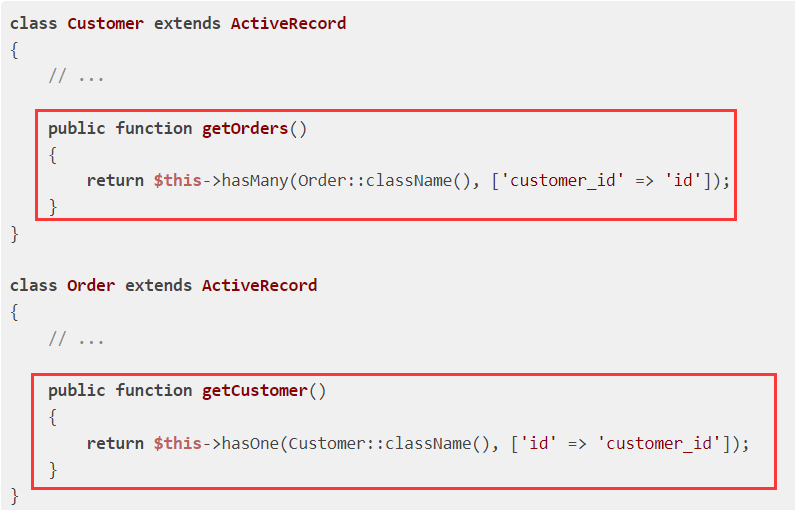
关于活动记录的常规用法不做赘述。

# 使用关联数据

除了处理单个数据库表之外，Active Record 还可以将相关数据集中进来， 使其可以通过原始数据轻松访问。 例如，客户数据与订单数据相关 因为一个客户可能已经存放了一个或多个订单。这种关系通过适当的声明， 你可以使用 $customer->orders 表达式访问客户的订单信息 这表达式将返回包含 Order Active Record 实例的客户订单信息的数组。

## 声明关联关系

你必须先在 Active Record 类中定义关联关系，才能使用 Active Record 的关联数据。 简单地为每个需要定义关联关系声明一个 关联方法 即可，如下所示，

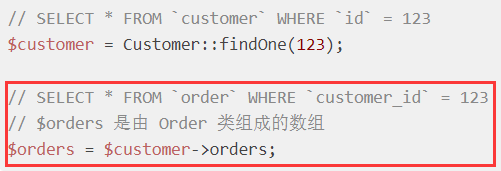


上述的代码中，我们为 Customer 类声明了一个 orders 关联， 和为 Order 声明了一个 customer 关联。

每个关联方法必须这样命名：getXyz。然后我们通过 xyz（首字母小写）调用这个关联名。 请注意关联名是大小写敏感的。

## 访问关联数据

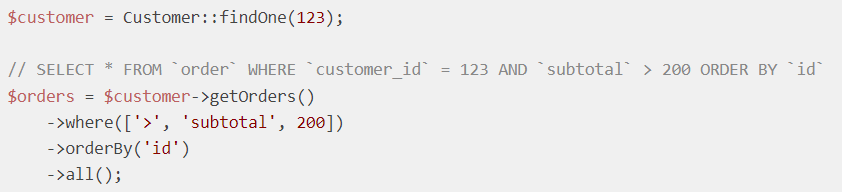
定义了关联关系后，你就可以通过关联名访问相应的关联数据了。就像 访问一个由关联方法定义的对象一样，具体概念请查看 [属性](https://www.yiichina.com/doc/guide/2.0/concept-properties)。 因此，现在我们可以称它为 关联属性 了。



当你第一次访问关联属性时，将执行 SQL 语句获取数据，如 上面的例子所示。如果再次访问相同的属性，将返回先前的结果，而不会重新执行 SQL 语句。要强制重新执行 SQL 语句，你应该先 unset 这个关联属性， 如：unset（$ customer-> orders）。

## 动态关联查询

由于关联方法返回 [yii\db\ActiveQuery](https://www.yiichina.com/doc/api/2.0/yii-db-activequery) 的实例，因此你可以在执行 DB 查询之前， 使用查询构建方法进一步构建此查询。例如，

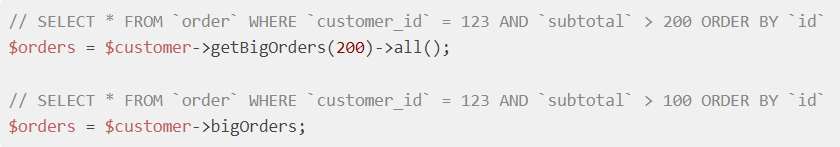


与访问关联属性不同，每次通过关联方法执行动态关联查询时， 都会执行 SQL 语句，即使你之前执行过相同的动态关联查询。

有时你可能需要给你的关联声明传递参数，以便您能更方便地执行 动态关系查询。例如，您可以声明一个 bigOrders 关联如下，



然后你就可以执行以下关联查询：



## 中间关联表

在数据库建模中，当两个关联表之间的对应关系是多对多时， 通常会引入一个[连接表](https://en.wikipedia.org/wiki/Junction_table)。

例如，order 表 和 item 表可以通过名为 order\_item 的连接表相关联。一个 order 将关联多个 order items， 而一个 order item 也会关联到多个 orders。

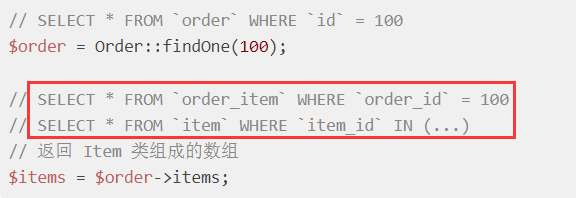
当声明这种表关联后，您可以调用 [via()](https://www.yiichina.com/doc/api/2.0/yii-db-activerelationtrait" \l "via()-detail) 或 [viaTable()](https://www.yiichina.com/doc/api/2.0/yii-db-activequery" \l "viaTable()-detail) 指明连接表。[via()](https://www.yiichina.com/doc/api/2.0/yii-db-activerelationtrait" \l "via()-detail) 和 [viaTable()](https://www.yiichina.com/doc/api/2.0/yii-db-activequery" \l "viaTable()-detail) 之间的区别是 前者是根据现有的关联名称来指定连接表，而后者直接使用 连接表。例如，



或者,



使用连接表声明的关联和正常声明的关联是等同的，例如，



## 通过多个表来连接关联声明

通过使用 [via()](https://www.yiichina.com/doc/api/2.0/yii-db-activerelationtrait" \l "via()-detail) 方法，它还可以通过多个表来定义关联声明。

再考虑考虑上面的例子，我们有 Customer，Order 和 Item 类。 我们可以添加一个关联关系到 Customer 类，这个关联可以列出了 Customer（客户） 的订单下放置的所有 Item（商品）， 这个关联命名为 getPurchasedItems()，关联声明如下代码示例所示：



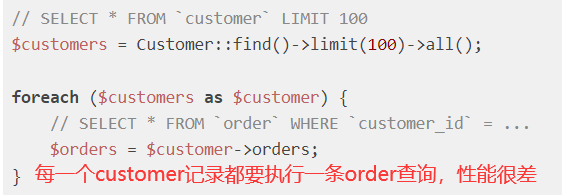
## 延迟加载和即时加载

在 [访问关联数据](https://www.yiichina.com/doc/guide/2.0/db-active-record" \l "accessing-relational-data) 中，我们解释说可以像问正常的对象属性那样 访问 Active Record 实例的关联属性。SQL 语句仅在 你第一次访问关联属性时执行。我们称这种关联数据访问方法为 延迟加载。

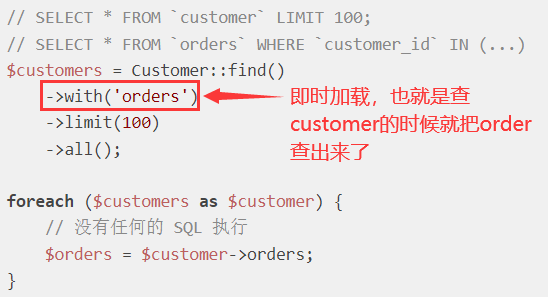
例如，



延迟加载使用非常方便。但是，当你需要访问相同的具有多个 Active Record 实例的关联属性时， 可能会遇到性能问题。请思考一下以下代码示例。 有多少 SQL 语句会被执行？



为了解决上述的性能问题，你可以使用所谓的 即时加载，如下所示，



你可以即时加载一个或多个关联。 你甚至可以即时加载 嵌套关联 。

嵌套关联是一种 在相关的 Active Record 类中声明的关联。例如，Customer 通过 orders 关联属性 与 Order 相关联， Order 与 Item 通过 items 关联属性相关联。 当查询 Customer 时，您可以即时加载 通过嵌套关联符 orders.items 关联的 items。

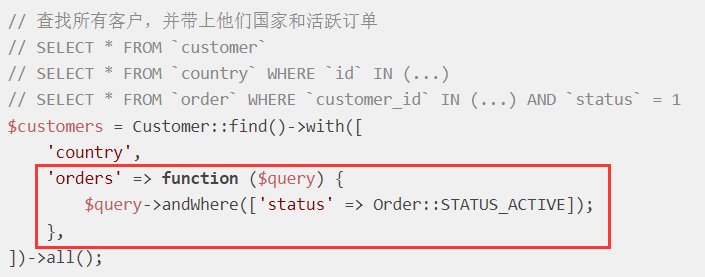
### with()语法

以下代码展示了 [with()](https://www.yiichina.com/doc/api/2.0/yii-db-activequerytrait" \l "with()-detail) 的各种用法。我们假设 Customer 类 有两个关联 orders 和 country，而 Order 类有一个关联 items。



### 自定义with()

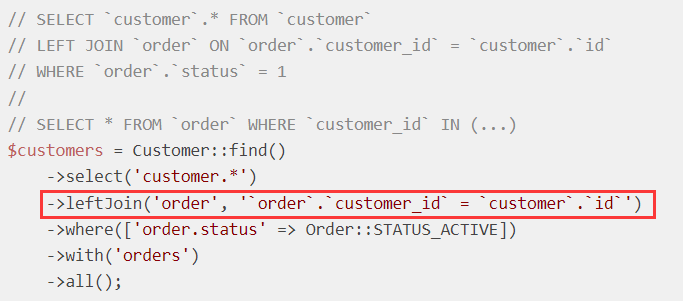
当即时加载一个关联，你可以通过匿名函数自定义相应的关联查询。 例如，



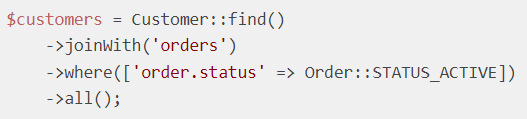
## 关联关系的 JOIN 查询

到目前为止，我们所介绍的关联查询，仅仅是使用主表列 去查询主表数据。实际应用中，我们经常需要在关联表中使用这些列。

例如， 我们可能要取出至少有一个活跃订单的客户。为了解决这个问题，我们可以 构建一个 join 查询，如下所示：



但是，更好的方法是通过调用 [yii\db\ActiveQuery::joinWith()](https://www.yiichina.com/doc/api/2.0/yii-db-activequery" \l "joinWith()-detail) 来利用已存在的关联声明：



两种方法都执行相同的 SQL 语句集。然而，后一种方法更干净、简洁。

默认的，[joinWith()](https://www.yiichina.com/doc/api/2.0/yii-db-activequery" \l "joinWith()-detail) 会使用 LEFT JOIN 去连接主表和关联表。 你可以通过 $joinType 参数指定不同的连接类型（比如 RIGHT JOIN）。 如果你想要的连接类型是 INNER JOIN，你可以直接用 [innerJoinWith()](https://www.yiichina.com/doc/api/2.0/yii-db-activequery" \l "innerJoinWith()-detail) 方法代替。

调用 [joinWith()](https://www.yiichina.com/doc/api/2.0/yii-db-activequery" \l "joinWith()-detail) 方法会默认 [即时加载](https://www.yiichina.com/doc/guide/2.0/db-active-record" \l "lazy-eager-loading) 相应的关联数据。 如果你不需要那些关联数据，你可以指定它的第二个参数 $eagerLoading 为 false。

### 关联表别名

 从 2.0.7 版本起，Yii 为此提供了一个快捷方法。您现在可以定义和使用关联表的别名，如下所示：

