

15. 构造器的 join 查询

学习要点：

1. join 查询

本节课我们来开始学习数据库的构造器查询中 join 查询。

一. join 查询

1. 使用 join 实现内联接的多表查询，比如三张表进行 inner join 查询；

```
$users = DB::table('users')
->join('books', 'users.id', '=', 'books.user_id')
->join('profiles', 'users.id', '=', 'profiles.user_id')
->select('users.id', 'users.username', 'users.email',
        'books.title', 'profiles.hobby')

->get();
```

2. 也可以使用 leftjoin 左连接或 rightjoin 右连接实现多表查询；

```
$users = DB::table('users')
->leftJoin('books', 'users.id', '=', 'books.user_id')
->rightJoin('profiles', 'users.id', '=', 'profiles.user_id')
->get();
```

3. 使用 crossjoin 交叉连接查询，会生成笛卡尔积，再用 distinct()取消重复；

```
$users = DB::table('users')
->crossJoin('books')
->select('username', 'email')
->distinct()
->get();
```

4. 如果你想要实现闭包查询，和 where 类似，只不过要用 on 和 orOn 方法；

```
$users = DB::table('users')
->join('books', function ($join) {
    //支持 orOn 连缀
    $join->on('users.id', '=', 'books.user_id');
})->toSql();
```

PS: on()方法后面如果想要再增加筛选条件，可以追加 where()；

5. 使用 joinSub 实现子连接查询，将对应的内容合并在一起输出；

```
//子连接查询
$query = DB::table('books')->selectRaw('user_id,title');
$users = DB::table('users')->joinSub($query, 'books', function ($join) {
```

```
$join->on('users.id', '=', 'books.user_id');  
})->get();
```

6. 使用 `union()` 或 `unionAll()` 方法实现两个查询的合并操作；

```
//union 取消重复, unionAll 不取消重复  
$query = DB::table('users');  
$users = DB::table('users')  
    ->union($query)  
    ->get();
```