

## 10. 数据库配置入门

学习要点：

### 1. 配置数据库

本节课我们来开始学习数据库的配置方法，以及连接数据库。

#### 一. 配置数据库

1. 框架支持原生、查询构造器和 Eloquent ORM(关系型对象映射器)来操作数据库；
2. 数据库的配置在 `config/database.php`，如果是本地可以直接配置 `.env` 文件；
3. 我们通过 `.env` 文件配置数据库连接的相关信息，以提供给 `database` 读取；

```
DB_CONNECTION=mysql                                // .env
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=grade
DB_USERNAME=root
DB_PASSWORD=123456

'mysql' => [                                         // database.php
    'driver' => 'mysql',
    ...
]
```

4. 我们可以直接创建一个新的控制器 `DataController` 来测试数据库部分；
5. 数据库有一个专用类 `DB`，可以用它来实现原生查询和构造器查询；

```
//使用 DB 类的 select()方法执行原生 SQL
$user = DB::select('select * from laravel_user');
return $user;
```

6. 查询构造器主要通过 `DB` 类的各种数据库操作方法来实现，比如选定一条；

```
//这里省去了 laravel_，需要在 database.php 配置
$user = DB::table('user')->find(19);
return [$user];
```

7. 由于火狐浏览器自动将 JSON 显示的很美化，而 `find()`只返回对象；

```
return response()->json($user);                    //这种写法去掉了外部的数组下标 0
```

8. 使用 Eloquent ORM 模型来操作数据库，使用命令在 `Http` 目录下创建模型；

```
php artisan make:model Http/Models/User            //默认在 app 目录
```

```
//使用 Eloquent ORM 构建
$user = User::all();
return $user;
```

9. 上面使用模型来操作数据后，报错提示数据表是复数：users；
10. 而我们真实的数据库表为：laravel\_user，为何会这样？？？
11. 前缀可以在 database.php 修改添加：laravel\_，最终变为：laravel\_users；
12. 由于模型编码规范要求数据表是复数，这里的复数并不是单纯加 s；
13. 可能会加 es，可能会加 ies，也可能是 child 编程 children 之类的；
14. 可以使用字符串助手：Str::plural()来判断英文单词的复数情况：

```
return Str::plural('bus');           //buses
return Str::plural('user');           //users
return Str::plural('child');          //children
```

15. 你可以根据规范去更改数据表名称，或者强制使用现有的数据表名；  
`protected $table = 'user';`