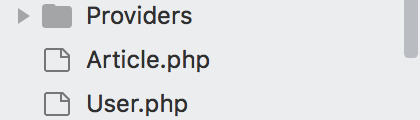
2018-08-30 Laravel总结学习，对象操作数据库，终端操作

后台对于数据库的操作是避免不了的，但是如何快速的进行数据库管理是很值得学习跟研究的，现在面向对象编程感觉仍然是很方便的，所以，数据的展示最好也要用对象来操作，而不是面向过程编程然后再对得到的数据进行解析然后操作。所谓的面向对象其实就是对数据进行严格结构化操作，化零为整，统一调度。

可以用$user = DB::select("select \* from user”);，这样的方法进行取数据，但是能不能再简化并且准确的操作数据呢？答案是可以的，那就是创建对应数据库model对象。

在终端命令行里输入创建model指令：

php artisan make:model Article ，

这样便在Laravel目录下创建了一个Model对象

我们需要在终端中输入如下命令行进行数据库的映射管理：

php artisan thinker

好了，下面便可执行创建模型及保存到数据库。

Article对象创建便可直接对数据库表中的Articles表进行操作。但是假如我们不在数据库中创建对应的表或者表中的字段与创建出来Article模型对象字段不完全一致会保存成功吗？

首先我们先建立一个Article模型：

$article=new App\Article;

对model进行赋值：

$article->title='My first title’;

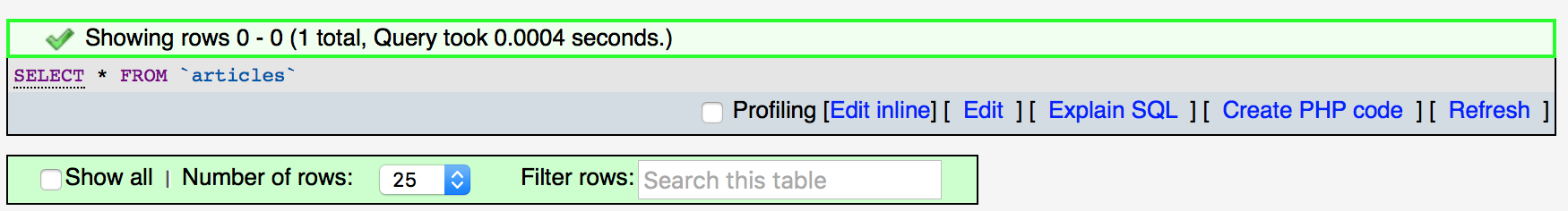
$article->content=‘content';

$article->published\_at=Carbon\Carbon::now();

然后进行保存操作：

$article->save();

出错了，看一下原因，上面之行了两次save()操作，第一次失败的原因是数据库中不存在articles数据表，根据提示去创建相应的表：

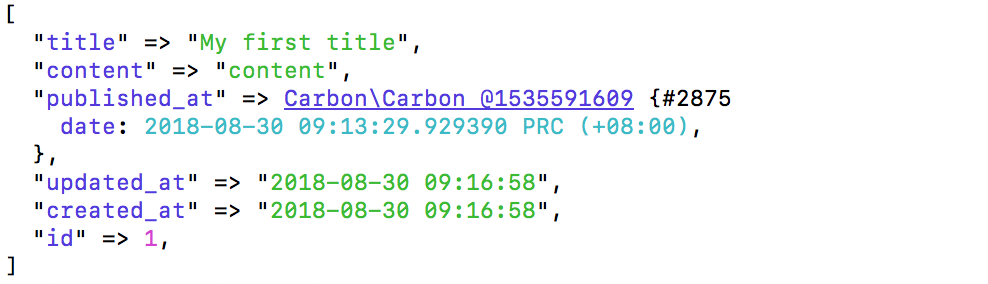


表格创建完毕，再次执行save();还是报错，看一下原因，原来是表中的字段与$article对象属性不完全一致导致的，把表里的字段写一致再执行save()操作一下试试：

可以看到，保存成功了，再看一下数据库：

可以看到数据库保存成功了。

保存成功后可以查看数据了，在终端输入如下命令行：

回车，查看数据源：

返回了从数据库中查询的数据了并且返回的是全部数据。

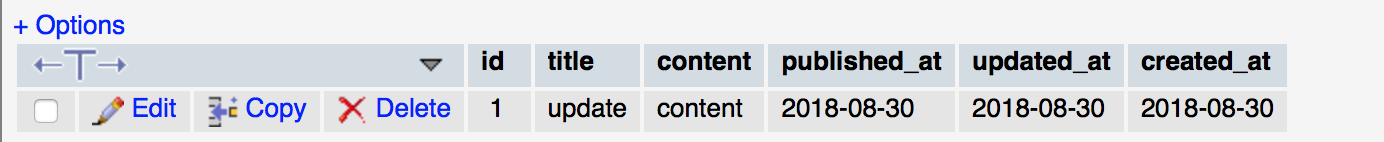
下面操作一下对数据库修改的操作，首先先搜索出一条数据：

$article=APP\Article::find(1);

$article->titile=‘uptate’;

$article->save();

现在在再去数据库中去看一下数据是不是变了，

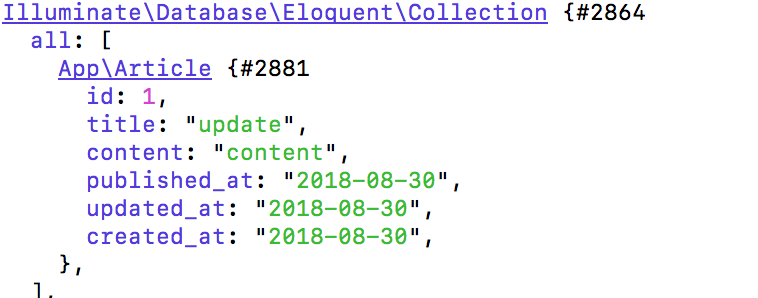


可以看见之前title字段变为了修改后的’update’，说明这里操作成功了。

查找第一条数据很明显不能满足后台开发的需求，where查询语句的运用显然是必要的，在命令行里输入命令行进行搜索：

$section=APP\Article::where(‘title’,’=’,’update’)->get();

回车搜索，返回结果如下：



到此终端命令行进行数据库进行对象操作进基本完成了简单的入门操作，下面的就是把上面的命令运用到php工程里去。php真的很灵活，到灵活的原因便是它本身的约束很少，所以扩展性极高，对于扩展思维广度而言很值得学习！