# Laravel笔记

1. 安装
   1. composer安装

composer create-project --prefer-dist laravel/laravel blog "5.2.\*"

1. 配置
   1. 使用config()获取你的设置值，设置值可以用[点]语法来获取：文件名.选项名称；

$value = config(‘app.timezone’);// show “UTC”

若要在运行期间修改设置，需要给config()传一个数组；

config([‘app.timezone’ => ‘Amercia/Chicago’]);

* 1. 维护模式

当你的应用处于维护模式时，所有传递至应用程序的请求都会显示出一个自定义视图。

启用维护模式：php artisan down

关闭维护模式：php artisan up

维护模式的默认模板在：resources/views/errors/503.blade.php

* 1. 数据库迁移

Laravel 的数据库迁移提供了一个简单的方式，使用流畅、一目了然的 PHP 代码来定义数据表的结构与修改。你无需再告诉团队成员要手动增加字段至他们本机的数据库中，他们只需要运行你提交到版本控制的迁移，即可应用数据库修改。

使用Artisan命令：

php artisan make:migration create\_tasks\_table --create=tasks//创建数据库迁移文件，里面等于是建表语句，默认放在database/migrations 目录中。

php artisan migrate //运行迁移，等于执行上面的迁移文件

* 1. Eloquent 模型

每个 Eloqunet 模型会直接对应一张数据表。

使用make:model命令来生成tasks数据表的模型Task（是个空模型，放在app目录下）：

php artisan make:model Task //默认创建的Task模型对应tasks数据表

1. 路由
   1. 响应多个HTTP动作的路由

Route::match(['get', 'post'], '/', function () {

//

});

Route::any('foo', function () {

//

});

* 1. 路由参数
     1. 必选参数

Route::get('posts/{post}/comments/{comment}', function ($postId, $commentId) {

//

});

路由参数都放在花括号内，参数不能含‘-’；

* + 1. 可选参数

Route::get('user/{name?}', function ($name = 'John') {

return $name;

});

在参数后面加‘？’表示可选；

* + 1. 正则限制参数

//使用where（）方法，前面是参数名，后面是正则；若同时限制多个参数，采用关联数组方式。

Route::get('user/{id}', function ($id) {

//

})

->where('id', '[0-9]+');

Route::get('user/{id}/{name}', function ($id, $name) {

//

})

->where(['id' => '[0-9]+', 'name' => '[a-z]+']);

* + 1. 全局限制

若希望路由参数一直遵循正则表达式，在RouteServiceProvider 的 boot 方法里定义这些模式：

public function boot(Router $router)

{

$router->pattern('id', '[0-9]+');

parent::boot($router);

}

模式一旦被定义，便会自动应用到所有使用该参数名称的路由上

* + 1. 命名路由

------//为这个路由指定一个名字；

Route::get('user/profile', ['as' => 'profile', function () {

//

}]);

------//也可以为控制器的方法指定路由名字：

Route::get('user/profile', [

'as' => 'profile', 'uses' => 'UserController@showProfile'

]);

------//还可以在路由定义后链式调用name()方法

Route::get('user/profile', 'UserController@showProfile')->name('profile');

一旦你在指定的路由中分配了名称，则可通过 route 函数来使用路由名称生成 URLs 或重定向：

$url = route('profile');

$redirect = redirect()->route('profile');

若路由中带参数，可以用 route(‘profile’,[‘id’=>12]); 直接将参数加入到url中；

-------//在路由群组的属性数组中指定 as 关键字，会为群组中所有路由设置相同的前缀

Route::group(['as' => 'admin\_'], function () {

Route::get('dashboard', ['as' => 'dashboard', function () {

// 路由名称为「admin\_dashboard」

return route(‘admin\_dashboard’);

}]);

});