# Laravel

目录机构-------解析

**App -----**

**Http -----控制器**

**Exceptions----异常处理代码**

**Console ----- 控制台信息**

**User.php ------ 模型类**

**Config ------- 配置文件**

**Database -------模型的定义，配置**

Public -------静态资源文件（入口文件）

**Resources -----视图**

**Assets ------未处理的css，js文件**

**Lang ------ 语言包**

**Views -----试图实现（.php文件内html）**

**Routes ------路由**

**Web.php 访问路由控制**

**Storage --------缓存**

**Tests -------单元测试**

**Routes（路由详情）**

一．请求方式

1. Get：Route::get('/about', function () {

return 'about';

});

1. post：Route::post('/about', function () {

return 'about';

});

1. delete：Route:: delete ('/about', function () {

return 'about';

});

1. patch：Route::patch('/about', function () {

return 'about';

});

二．路由传参

1. 必填参数

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

return 'id=>'.$id;

});

2. 选填参数

Get：Route::get('/about/{id?}', function ($id=’默认的参数’) {

return 'id=>'.$id;

});

3. 参数验证 id必须是数字

Route::*get*('/about/{id}', **function** ($id='默认的') {  
 **return** 'id=>'.$id;  
})->where('id','[0-9]+');

/\* 基础控制器如何使用！ \*/

# Controllers

1. 路由绑定控制器：

Route::get('member/info','MemberController@info');

Route::get('member/{id}',[

'uses'=>'MemberController@info',

'as'=>'cs/{id}'

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

1. 多模块控制器

绑定方法：

Route::get('/admin','Admin\IndexController@index');

1. 资源控制器（用于注册实例类控制器路由增加删除修改）

命令生生成资源控制器：

php artisan make:controller UserController –resource

资源管理器提供命名规范：

1. index ---资源列表
2. create ---添加资源表单页面
3. store -----添加功能的实现
4. show -----查看单个实例
5. edit -----修改实例页面
6. update -----修改功能的实现
7. destore -----删除功能的实现

路由注册

Route::resource('/user','Admin\UserController');

视图传参：

$nam=’’;

$list=’’;

return view('\Admin\User\index',compact('list','name'));

视图{{$name}}绑定参数

1. 静态页面 控制器：

命令生静态页面 控制器：

php artisan make:controller StaticPagesController

# 视图

1. 模板继承

@extends('layouts.default') 继承 layouts文件夹下的default模板（只要名称。不要模板后缀）

# 微博测试项目

1. 安装依赖（自带依赖 ）

bootstrap-sass —— Bootstrap NPM 扩展包；

jquery —— jQuery NPM 扩展包；

laravel-mix —— 由 Laravel 官方提供的静态资源管理工具；

vue —— VUE.js 前端框架；

1. 安装命令:
   1. yarn install --no-bin-links
   2. yarn add cross-env
   3. npm run watch-poll （监听resources/assets/sass/app.scss的变化，将其编译到public/css下）