Laravel 01

1. Laravel框架定义

Laravel是一套简洁、优雅的PHP开发框架，可以将开发者从杂乱的代码中解放出来，通过简单的表达式语法开发出很给力的Web应用，Laravel拥有高质量的文档，丰富的扩展包，被称为“巨匠级PHP开发框架”。

1. laravel框架常用的网站

1.http://laravelacademy.org/ laravel学院

2.http://doc.laravel-china.org/ laravel中国

3.http://www.golaravel.com/ laravel社区

三、安装方式

1. 利用composer安装

Composer create-project laravel/laravel your-project-name --prefer-dist “5.5.\*”

1. 一键安装包

http://www.golaravel.com/download/

1. Github 项目托管平台

<https://github.com/laravel/laravel>

四、Composer

1.下载 <https://getcomposer.org/download/>

2.安装 如果不出问题就是下一步傻瓜式操作，出问题见composer安装讲义

五、Laravel框架主要目录结构

App 应用程序核心代码所在目录

Bootstrap 自动加载文件及缓存目录

Config 配置文件目录

Database 数据库填充、迁移目录，可以理解为数据库存放目录

Public 网站根目录，包含入库文件，及其他静态资源目录

Resources 视图、语言目录

Routes 路由目录

Storage 模板、文件、日志缓存目录

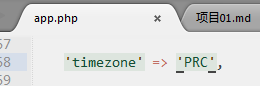
Tests 自动化测试文件目录

Vendor composer依赖包

六、配置要求

1.保证storage目录和bootstrap/cache目录有写入权限（默认开启）

2.修改时区 config/app.php “timezone”=>”PRC”



3.laravel框架程序秘钥

1. 通过composer安装自带秘钥
2. 如果没有可以用php artisan key:generate命令在laravel框架内生成，然后放到.env文件里。
3. 克隆框架没有秘钥（版本库中更新项目，无.env文件，可以复制一份，然后更新生成秘钥）
4. 没有秘钥会报错，**不能确保你的用户会话和其他加密数据的安全！**
5. **Artisan 其实是包含php文件的工具匠，里面有很多的php文件，在命令行下用php命令实现php文件，实现功能：例如 创建控制器、请求、中间件、生成程序秘钥**

IMG_256

4.读取和设置配置

读取 config::get(‘app.属性名’);

设置 config::set(‘app.属性名’，‘属性值’);

5.获取环境配置

Env(‘DB\_HOST’)

七、路由

1.目录

Routes

---| console.php 定义所有基于闭包的控制台命令的地方

---| channels.php 用来注册你的应用支持的所有的事件广播渠道的地方

---| api.php 无状态路由

---| web.php 主要路由文件

2.基本路由

<1.>路由格式

Route::get('路由', function () {

return 'Hello World';

});

<2.>get请求方式

Route::get($uri, $callback);

<3.>post请求方式

Route::post($uri, $callback);

<4.>put请求方式

Route::put($uri, $callback);

<5.>delete请求方式

Route::delete($uri, $callback);

<6.>视图路由

Route::view('/welcome', 'welcome', ['name' => 'Taylor']);

<7.>重定向路由

Route::redirect('/here', '/there', 301);

<8.>其他路由

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

//路由匹配get、post请求方式

});

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

//路由匹配任何一种请求方式

});

备注：post表单请求需要 csrf保护，像put、delete请求需要传递请求方法

3.传参

<1.>必填参数

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

return 'User '.$id;

});

<2.>可选参数

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

return $name;

});

<3.>多参数

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

//

});

<4.>参数限制

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

//限制用户名

})->where('name', '[A-Za-z]+');

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

//限制数字

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

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

//限制多参数

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

4.路由组

Route::group([], function(){

Route::get('/404', function(){

abort(404,'not found!!!');

});

});

5.路由别名

Route::get('user/profile', function () {

//第一种别名方式

})->name('profile');

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

//第二种别名方式

}]);

\*\*\*\*使用

$url = route('profile');

// 生成重定向...

return redirect()->route('profile');

6.前缀

Route::group(['prefix' => 'admin'], function () {

Route::get('users', function () {

// 匹配 "/admin/users" URL

});

});

7.路由命名空间

Route::group(['namespace' => 'User'], function(){

// 控制器在 "App\Http\Controllers\Admin\User" 命名空间下

});

八、练习作业

1.练习操作composer

2.利用form表单，熟练操作各种路由请求方式