<https://learnku.com/docs/laravel-core-concept/5.5>

# 依赖注入、控制翻转、反射

## 1.反射

根据类名返回该类的任何信息，比如该类有什么方法，参数，变量等等

// 获取User的reflectionClass对象

$reflector = new reflectionClass(User::class);

// 拿到User的构造函数

$constructor = $reflector->getConstructor();

// 拿到User的构造函数的所有依赖参数

$dependencies = $constructor->getParameters();

// 创建user对象

$user = $reflector->newInstance();

// 创建user对象，需要传递参数的

$user = $reflector->newInstanceArgs($dependencies = []);

# ioc容器理解

# Closure匿名函数、闭包

匿名函数：没有名字的函数，最经常用作回调函数的参数值

## 1.匿名函数赋值

$a=function($name){

echo '1231'.$name;

};

//调用闭包函数

$a('123');

### 在函数中定义闭包函数并调用

//在函数中定义闭包函数并调用

function test(){

$fun=function($str){

echo $str;

};

$fun('aaaa');

}

test();//输出aaaa

### 在函数中把匿名函数返回，并调用它

// 在函数中把匿名函数返回，并调用它

function test(){

return function($str){

echo $str;

};

}

$fun=test();

$fun('aaaa');

### 匿名函数当做参数传递，并且调用它

// 把匿名函数当做参数传递，并且调用它

function test($fun){

$fun('aaaa');

}

$printStr=function($str){

echo $str;

};

test($printStr);//输出aaaa

## 2.use关键字用于连接闭包和外界变量

use关键字用于连接闭包和外界变量

function getMoney() {

$rmb = 1;

$dollar = 8;

$func = function() use ( $rmb ) {

echo $rmb;

echo $dollar;

};

$func();

}

getMoney();

//输出：1

dollar没有在use关键字中声明，在这个匿名函数里也就不能获取到它

**use引用的变量是一个副本，在匿名函数中修改引用的变量，原来的外界变量是不变的**

function getMoney() {

$rmb = 1;

$func = function() use ( $rmb ) {

echo $rmb.'<br>';

//把$rmb的值加1

$rmb++;

};

$func();

echo $rmb;

}

getMoney();

//输出：

//1

//1

可以用完全引用来改变外界变量

function getMoney() {

$rmb = 1;

$func = function() use ( &$rmb ) {

echo $rmb.'<br>';

//把$rmb的值加1

$rmb++;

};

$func();

echo $rmb;

}

getMoney();

//输出：

//1

//2

## 3.闭包绑定

通过bindTo将闭包内部状态绑定到对象上，

class A {

private static $sfoo = 1;

private $ifoo = 2;

}

$cl1 = static function() {

return A::$sfoo;

};

$cl2 = function() {

//匿名函数中莫名奇妙的有个this,这个this关键词

//就是说明这个匿名函数是需要绑定在类中的。

return $this->ifoo;

};

$bcl1 = Closure::bind($cl1, null, 'A');

$bcl2 = Closure::bind($cl2, new A(), 'A');

echo $bcl1(), "\n";//输出 1

echo $bcl2(), "\n";//输出 2

<https://www.cnblogs.com/phpper/p/6993093.html>

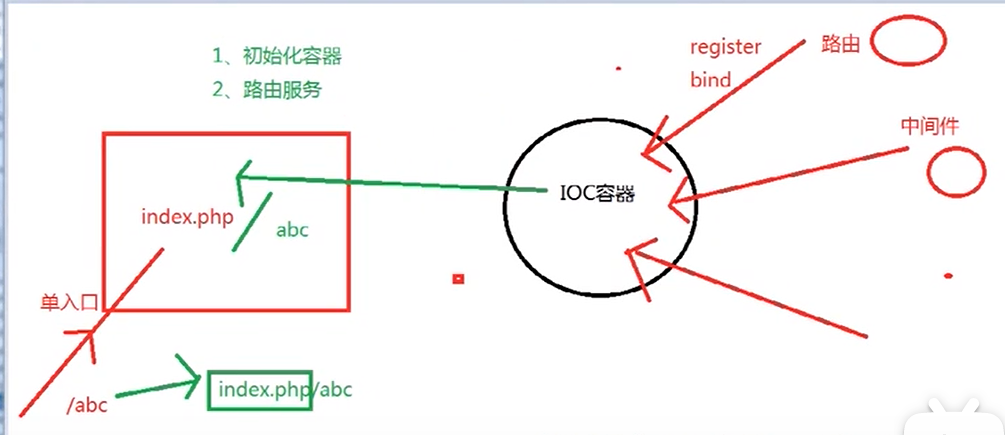
**源码位置：vendor/laravel/illuminate**

# 系统运行方式

系统组件（路由、中间件、service）

应用组件

组件注册在ioc容器中

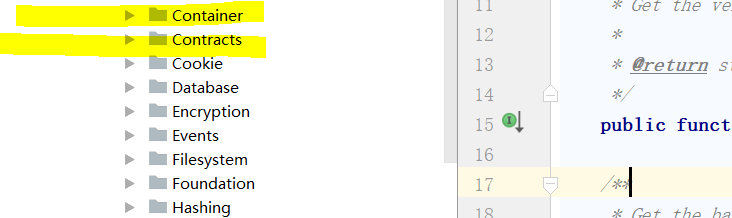


运行步骤：

入口文件Index.php进入

1. 初始化容器
2. 路由服务

# Laravel启动流程



两个文件区别

## Bind

## Make

Ioc容器源码解析