## 命名空间

tp5里，命名空间对应了文件所在的目录，app命名空间通常代表了文件的起始目录为application，而think命名空间则代表了文件的起始目录为thinkphp/library/think,后面的命名空间则表示从起始目录开始的子目录

引入的核心类，都是以小写think开头的。

## url

模块名官方规定小写

默认情况下，URL地址中的控制器和操作名是不区分大小写的

5.0不支持普通url访问

## 路由

route.php里两种定义方式：

|  |
| --- |
| return [  //name可以不传值；  'test/[:name]' =>'index/index/test',  ]  或者  \think\Route::rule('test/[:name]','index/index/test'); |

以上路由只要以test开始，就能匹配到改路由；

若需要完整匹配，即test后面只能传一个参数，使用类似正则的$符。

|  |
| --- |
| return [  // 路由参数name为可选  'hello/[:name]$' => 'index/hello',  ];  http://tp5.com/hello/thinkphp // 正确匹配  http://tp5.com/hello/thinkphp/val/value // 不会匹配 |

闭包定义：

|  |
| --- |
| return [  // 定义闭包  'hello/[:name]' => function ($name) {  return 'Hello,' . $name . '!';  },  ];  或：  return [  // 定义闭包  'hello/[:name]' => function ($name) {  return 'Hello,' . $name . '!';  },  ]; |

路由规则：

|  |
| --- |
|  |

[‘控制器/方法’，[‘method’ =>’get/post’],[ 所有参数对应的正则]]

路由分组：

|  |
| --- |
| 都是以blog开头的，可以分组： |

全局变量规则

|  |
| --- |
| return [  '\_\_pattern\_\_' => [  'name' => '\w+',  ],  ] |

当定义了全局变量规则时，后面的带有name参数的路由，都可以不再单独设置参数规则；

若之后又设置了局部变量规则，局部规则会覆盖全局规则。

|  |
| --- |
| return [  '\_\_pattern\_\_' => [  'name' => '\w+',  ],  'blog/:name' =>['blog/read', ['method' => 'get'], ['name' => '\w{5}']],  ] |

会以下面规则为准。

## 生成URL（路由地址）

路由 定义完了之后，使用Url类生成实际的url地址（路由地址）；对于上面的路由

public function archive(*$year*, *$month*)  
{  
 return Url::build('blog/archive',['year'=>*$year*,'month'=>*$month*]);  
}

build（‘完整的实际地址’，[参数与参数值的数组] or ‘query字符串’）

或用全局帮助函数url；

url('blog/archive',['year'=>$year,'month'=>$month]);

若配置了url\_html\_suffix，会带返回带后缀的url。

## 模型类

模型类命名采用驼峰法

## RESTful

