# ThinkPhp知识点

## 1.查询时用between.

 $where1['create\_time'] = ['between', strtotime($data['start\_time']) . ',' . strtotime($data['end\_time'])];  
 ​  
 $where1['status']='1';  
 // $where1['id']='1';  
 $where1['app\_id']=$merchant\_id;  
 $money=Db::name('order')->where(array\_merge($where1, ['id'=>['<',$value['id']]]))->order('id desc')->sum('money');

## 2.查询时获取单条记录的某一个值用value

 Db::name('order')->where(['id'=>'1'])->value('name');

## 3.composer 安装

Composer create-project topthink/think tp5.1 5.1.\* --prefer-dist

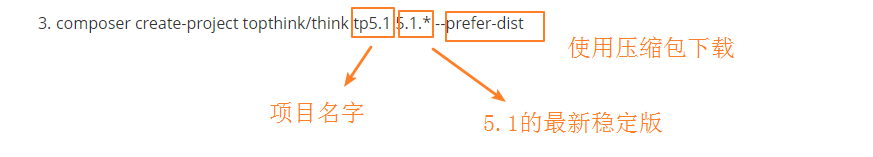
Composer create-project topthink/think tp5.1 5.1.\* --prefer-dist

* [topthink/framework](https://packagist.org/packages/topthink/framework): 5.1.\*

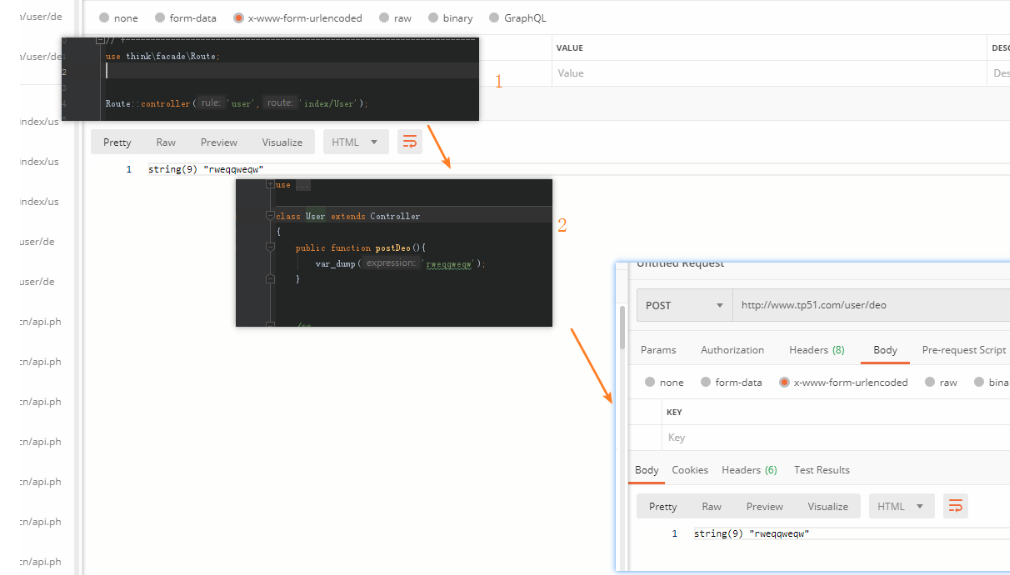
Composer create-project topthink/think xcx 5.1.\* --prefer-dist

Composer create-project topthink/think packageTp5 5.1.\* --prefer-dist

Composer create-project topthink/think tp\_demo 5.1.\* --prefer-dist



## 4.TP5.0之后,可以使用控制器路由



## 5.使用$request接收参数

当控制器继承了Controller之后,可以直接使用$this->request->post();获取所以post的参数.

## 6.中间件

作用:过滤和拦截http请求.

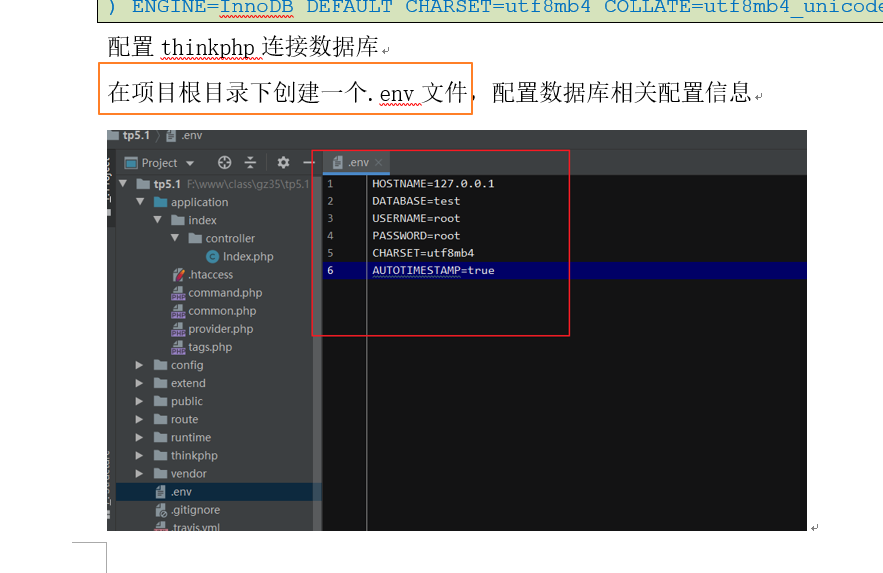
使用步骤:1.定义中间件(php think make:middleware Check).2.根据业务需求编辑中间件的内容.3.注册中间件(一般在路由中注册即可)

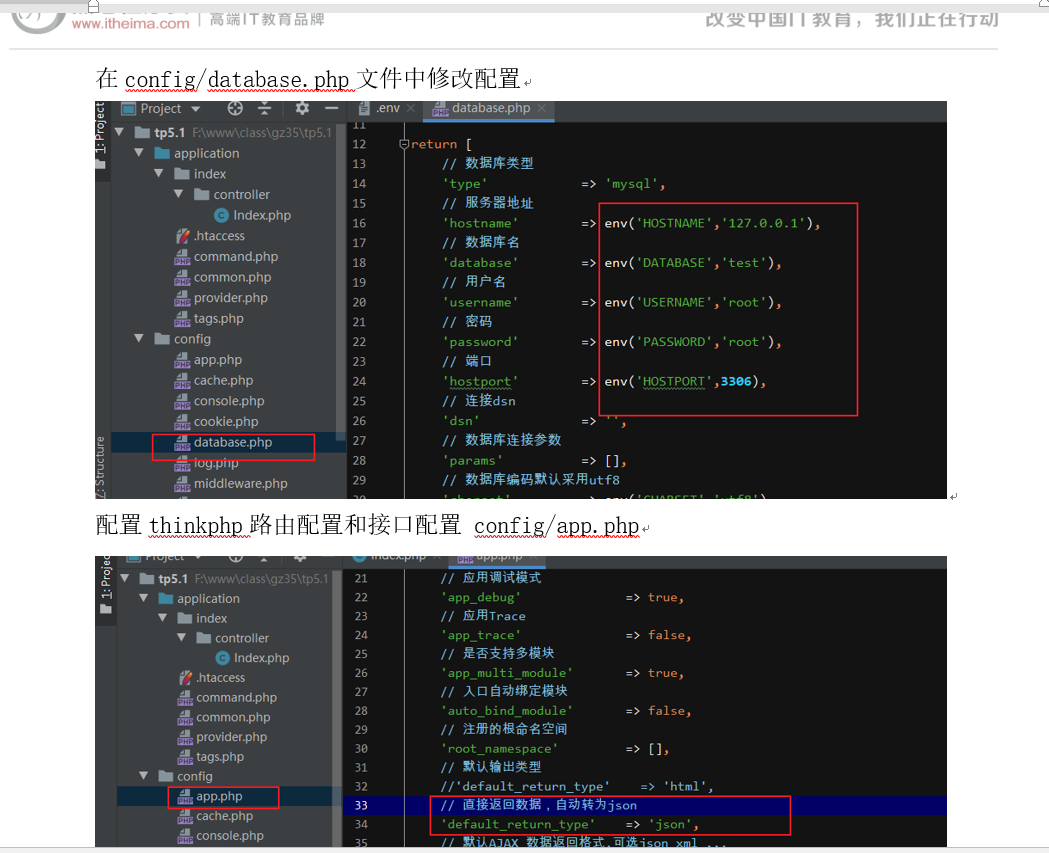


## 7.开始的准备工作

 绑定API模块  
     define('BIND\_MODULE', 'api');  
 设置跨域请求头header("Access-Control-Allow-Origin: \*");  
 // 如果需要自定义请求头时，直接把请求头添加到下面的参数里面即可  
 header("Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept, Authorization");  
 // 前端VUE框架自定义Header头请求API时，会发起一次预请求，thinkphp默认是拒绝OPTIONS类型的请求的  
 // 判断是否是OPTIONS请求，是的话直接exit结束，直接让浏览器返回200  
 if($\_SERVER['REQUEST\_METHOD'] == 'OPTIONS')  
 {      
     exit;  
 }

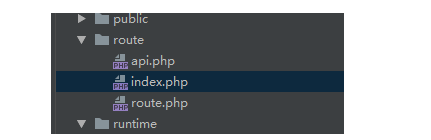
新建env文件







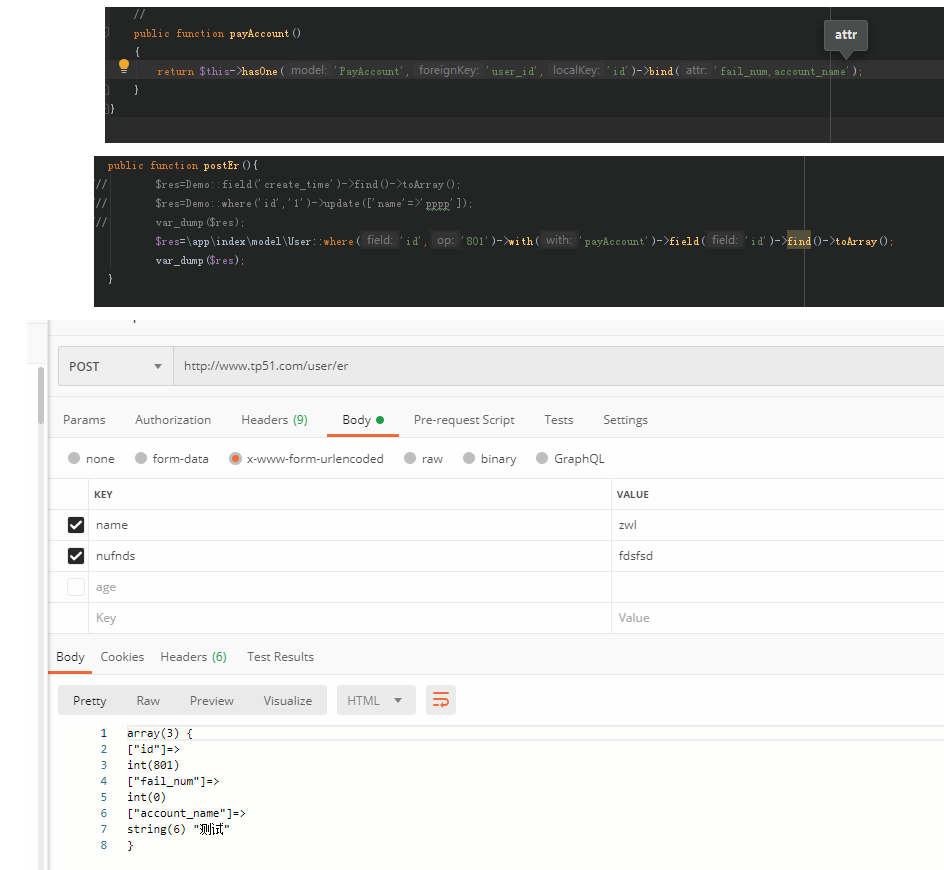
## 8.只要文件在route文件夹中,都可以被自动加载到路由文件,不需要在重新引入了



## 9.关联查询

一对一才可以使用bind().直接绑定字段.

使用field即可过滤相应的字段



## 10.渲染模板时,直接用view即可.不用$this->fetch()了,太麻烦了



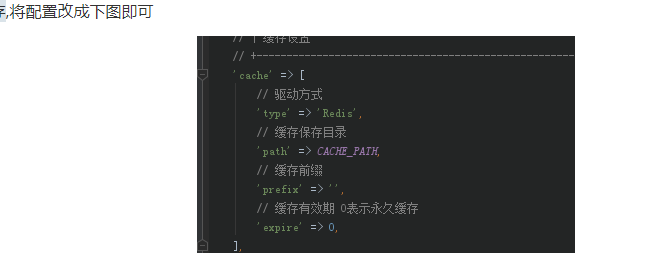
## 11.验证器



## 12.使用token防御csrf攻击



## 13.TP框架中使用redis缓存



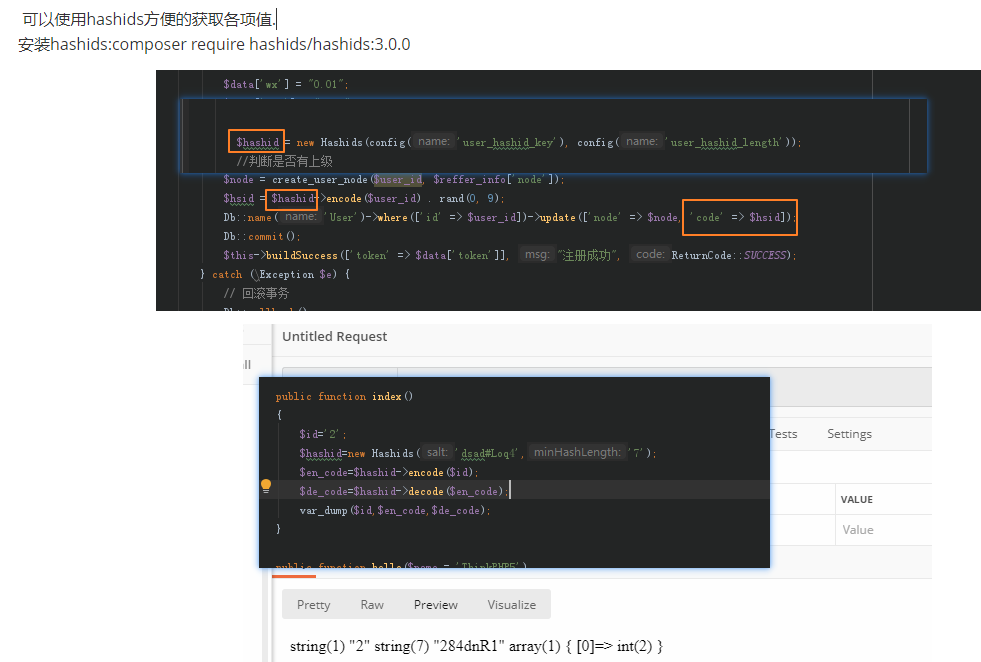
## 14.获取新增数据的id方法

 $user\_id = Db::name('User')->insertGetId($data)

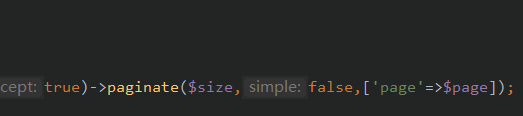
## 15.用id生成唯一标识码（hashids）

经常获取到id之后,一般用新增的id来生成改用户的唯一标识码或者改用户的推荐码.因为id具有唯一性.

可以使用hashids方便的获取各项值. 安装hashids:composer require hashids/hashids:3.0.0



## 使用分页的方法



Size是每页显示的数量，page显示的页面

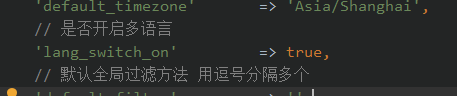
会返回跟多分析的数据。自己取即可

## 使用whereor查询数据

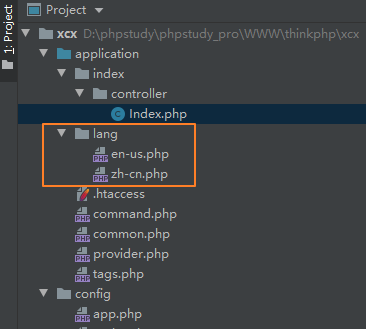
$data = LotteryRecord::*getInstance*()->where(function ($query) use ($where1){  
 $query->where($where1);  
})->whereOr(function ($query) use ($where2){  
 $query->where($where2);  
})->field('id,status,value,mix,day,expire\_time')->paginate($size,false,['page'=>$page]);

## 使用多语言

1. 打开多语言切换配置



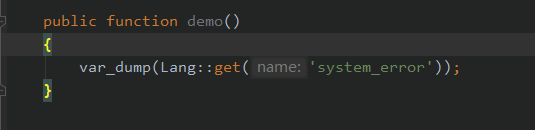
1. 在application目录下新建lang文件



会被自动加载到

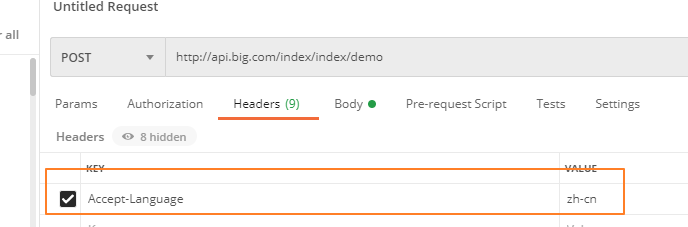
1. 然后在控制器的接口中使用Lang类，



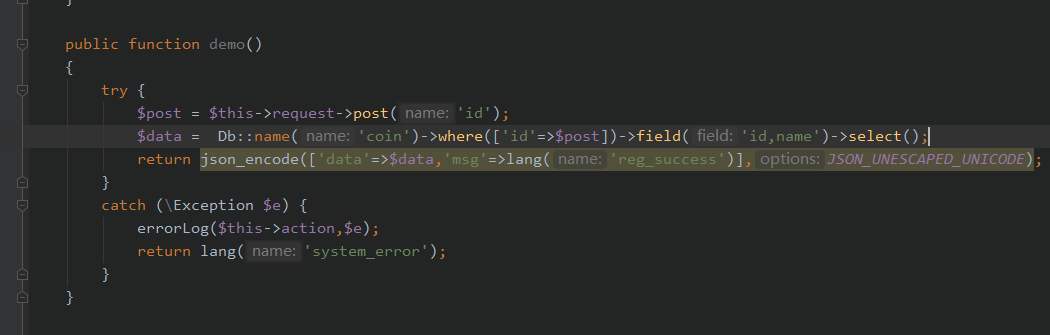


注意，是facade下面的Lang，才可以这样用。

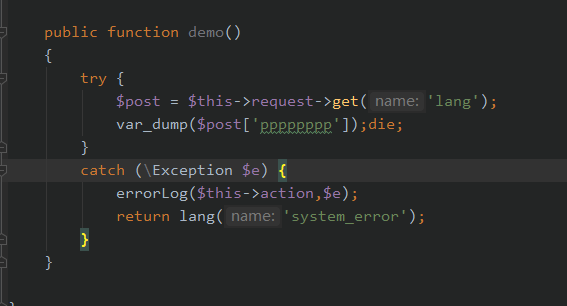
1. 请求的时候用header头的形式，带上Accept-language参数和参数值



控制器会根据lang参数自动识别，返回对应的提示信息。



## 写接口时要用下图的形式去编写



protected $action;  
public function \_\_construct()  
{  
 parent::*\_\_construct*();  
 $this->action = $this->request->action();  
}

Action是报错的方法。

即使错误了，也会抛出异常，可以自定义异常信息。还要记录日志。使用

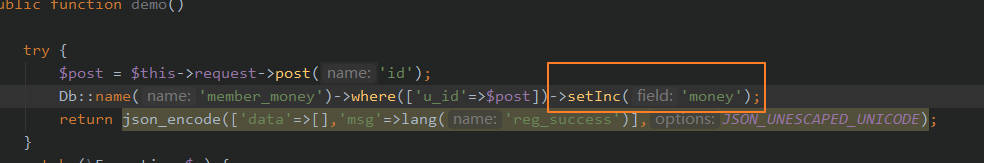
use think\facade\Log;  
*/\*\*  
 \** ***@param*** *$action  
 \** ***@param*** *$error  
 \*/*function errorLog($action,$error)  
{  
 $e = $error->getMessage().'in'.$error->getFile().$error->getLine();  
 Log::alert($action."\n".'============error=============start'."\n".$e."\n".'============error=============end');  
}

Tp自带的log类，记录对应的日志。

## ThinkPhp的生命周期

1. 经过入口文件。
2. 加载框架的引导文件。
3. 注册自动加载机制。
4. 注册错误和异常机制
5. 应用初始化
6. uRL访问检测
7. 路由监测
8. 分发请求
9. 响应输出
10. 应用结束

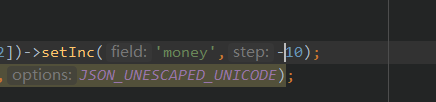
## 使用setInc方法可以方便的操作金额。



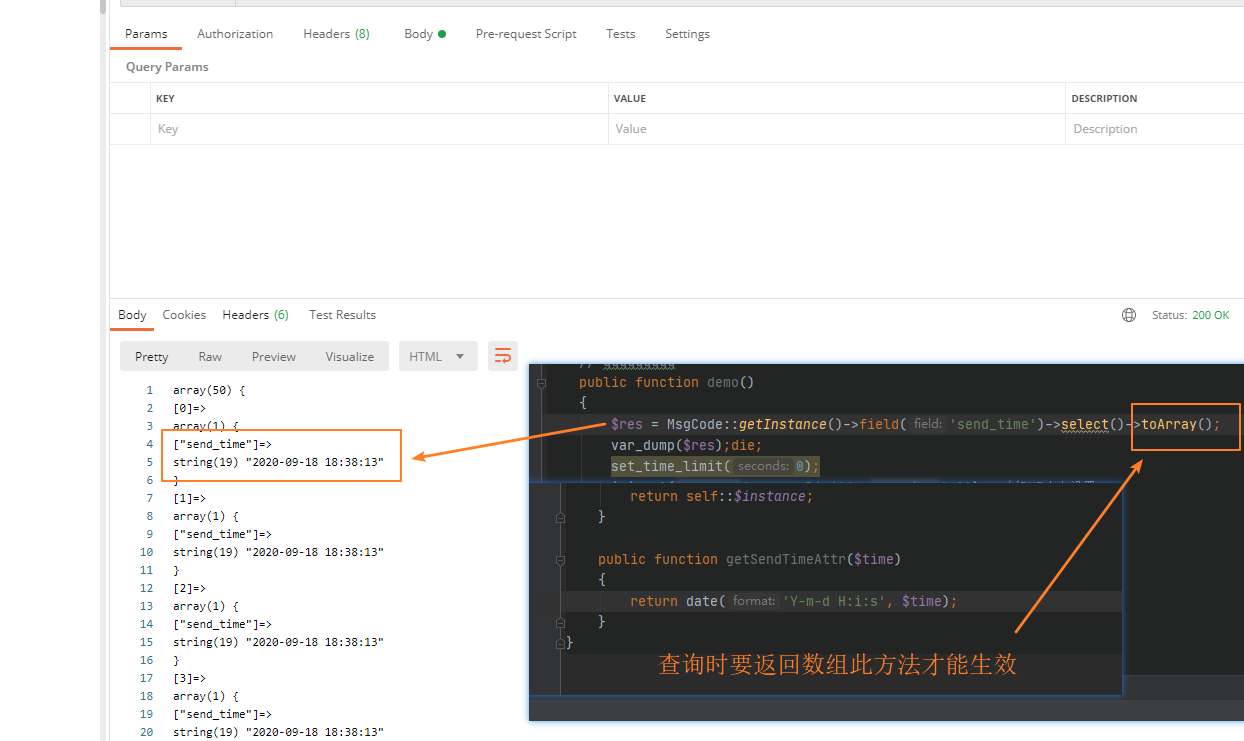
只有一个参数就加一，如果有两个参数，就加第二个参数值。



用负数也可以。



## 自动转换时间



## 23find（）和select（）方法区别

当find（）找到的数据为空时，会返回null。

当select找到的数据为空时，会返回空数组或空对象。空的对象去判断的结果是true，空数组去判断的结果是false

$object = 空对象。

If ($object) {

echo ‘ok’；

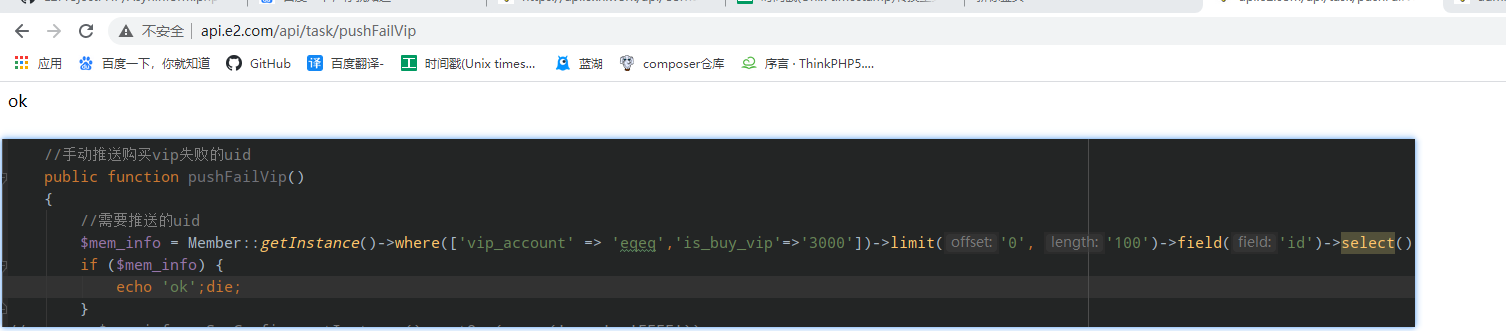
}

Else {

Echo ‘no’；

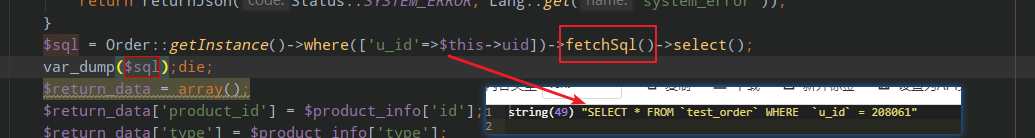
}

输出ok



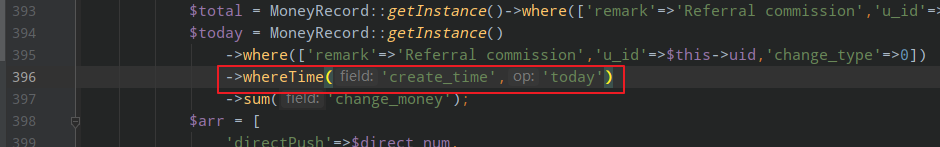
注意此细节。

## 24获取某条模型语句的原生sql。可以使用fetchSql方法，进行获取。比如



## 25使用时间查询时可以用whereTime方便的查询





## 26Find\_in\_set。查询某个用户的团队人数，

如果某个字段记录了所有上级的信息。可以用find\_in\_set函数进行查询。比如where（“find\_in\_set({$this->uid},p\_uids)”）->select();

代码的意思是是，p\_uids中是否存在$this->uid。如果某个用户的p\_uids字段中，有this-》uid，说明某个用户是this-》uid的下级。也就是团队人数。

1. 日志记录

Log类中记录日志的方法一般有两种。

分别是record和write。

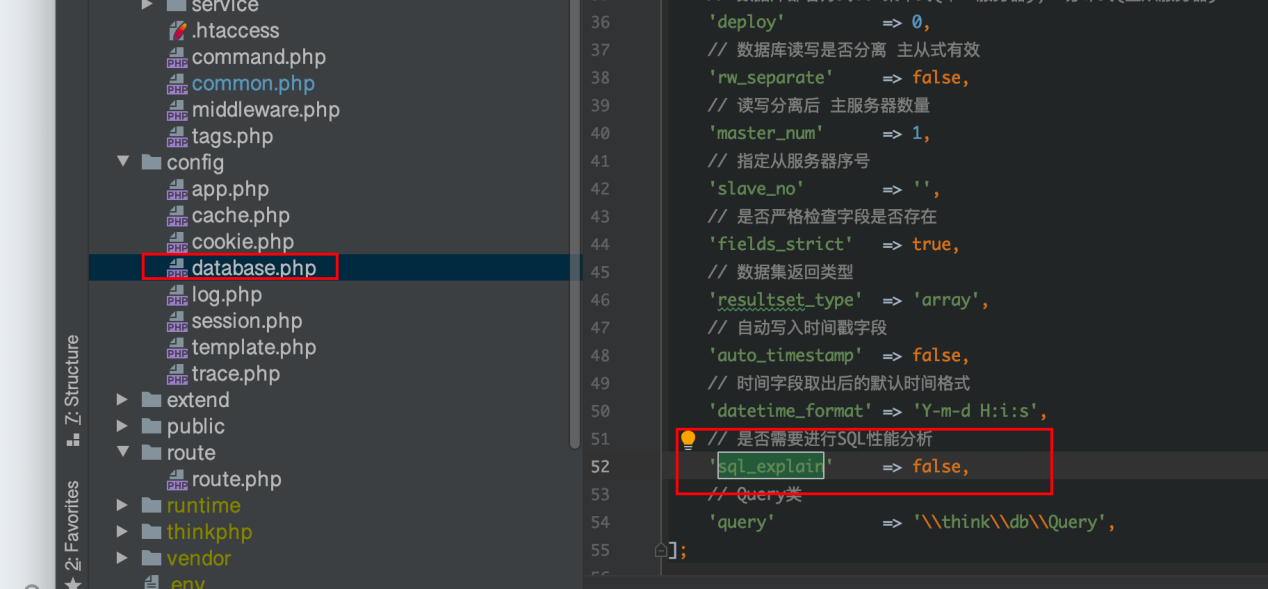
record不是实时写入日志信息的。write是实时写入日志的。



，写入日志时，可以自定义日志的记录文件，用init初始化，然后用path参数设置。

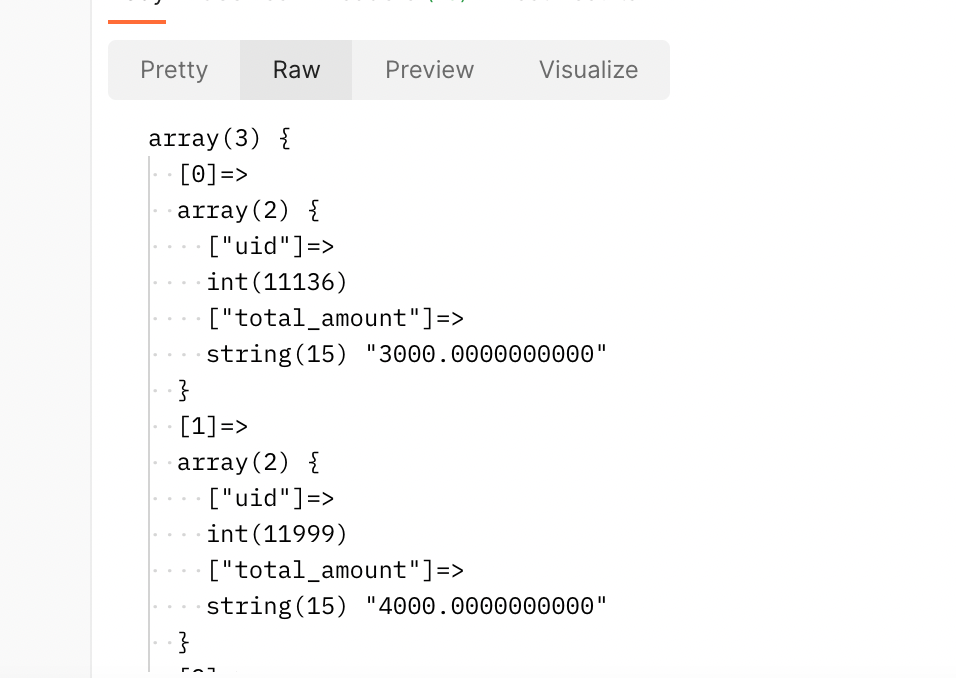
Log::init(['type' => 'File', 'path' => '../runtime/errorlogs/']);

## 27.如果接口响应慢的话，可以打开配置sql\_explain，分析接口的每一条sql运行的具体情况。



## 28使用group by报错解决方法

28.Db::name('recharge\_order')->field('any\_value(uid) AS uid,SUM(any\_value(amount)) AS total\_amount')->group('uid')->select();



## 29.tp5.1版本分页要像下面这样子分页

{$page|raw}

如果是tp5.0版本直接即可

{$page}

## 30错误日志，最好开启数据库调试模式，日志中才会记录sql语句。

tp5.1记录这样子记录

*/\*\*  
 \** ***@param*** *$action  
 \** ***@param*** *$error  
 \*/*function errorLog($action, $error)  
{  
 $log = Log::init(['path' => *ROOT\_PATH*.'runtime/errorlogs/']);  
 $log->record($action . "\n" . '============error=============start' . "\n" . $error . "\n" . '============error=============end');  
}

Tp5.0时要这样子记录。

*/\*\*  
 \** ***@param*** *$action  
 \** ***@param*** *$error  
 \*/*function errorLog($action, $error)  
{  
 Log::*init*(['type' => 'File', 'path' => *ROOT\_PATH* . 'runtime/errorlogs' . *DS*]);  
 Log::*write*($action . "\n" . '============error=============start' . "\n" . $error . "\n" . '============error=============end');  
}

## 钩子函数一般和行为搭配执行。

中间件不指定方法名的话，默认执行run方法。

$res = Hook::*exec*('app\\service\\UserServer','getVipProportion',$level);

## 使用这个name方法，即可实现分表，按月，按星期，按日分表都可以的。



## 33.主从配置

要配置下图中的选项。





读取主库的情况

