

手摸手教你 如何成为一名优秀的工程师



自我介绍

我是周梦康，后端研发工程师。

常用语言 `php`、`java`，《深入 PHP 内核》作者之一。

博客	https://mengkang.net
github	https://github.com/zhoumengkang
微博	http://weibo.com/zmkang
微信	zhoumengkang
邮箱	zhoumengkang@php.net

几个二维码



本次直播的回放地址



微博互动问答



微信征集 PHP 内核 Bug



来自灵魂深处的拷问（这不是大纲）

你有注意自己代码的语义表达吗？

项目上线有检查代码语法错误吗？

代码的注释你认真写了吗有坑吗？

写代码时有注意分段分层复用吗？

你的代码在别人看来会是黑洞吗？

添加查询语句考虑数据库性能吗？

你的代码上线运行时真的安全吗？

我把上面这些字对这么齐容易吗？

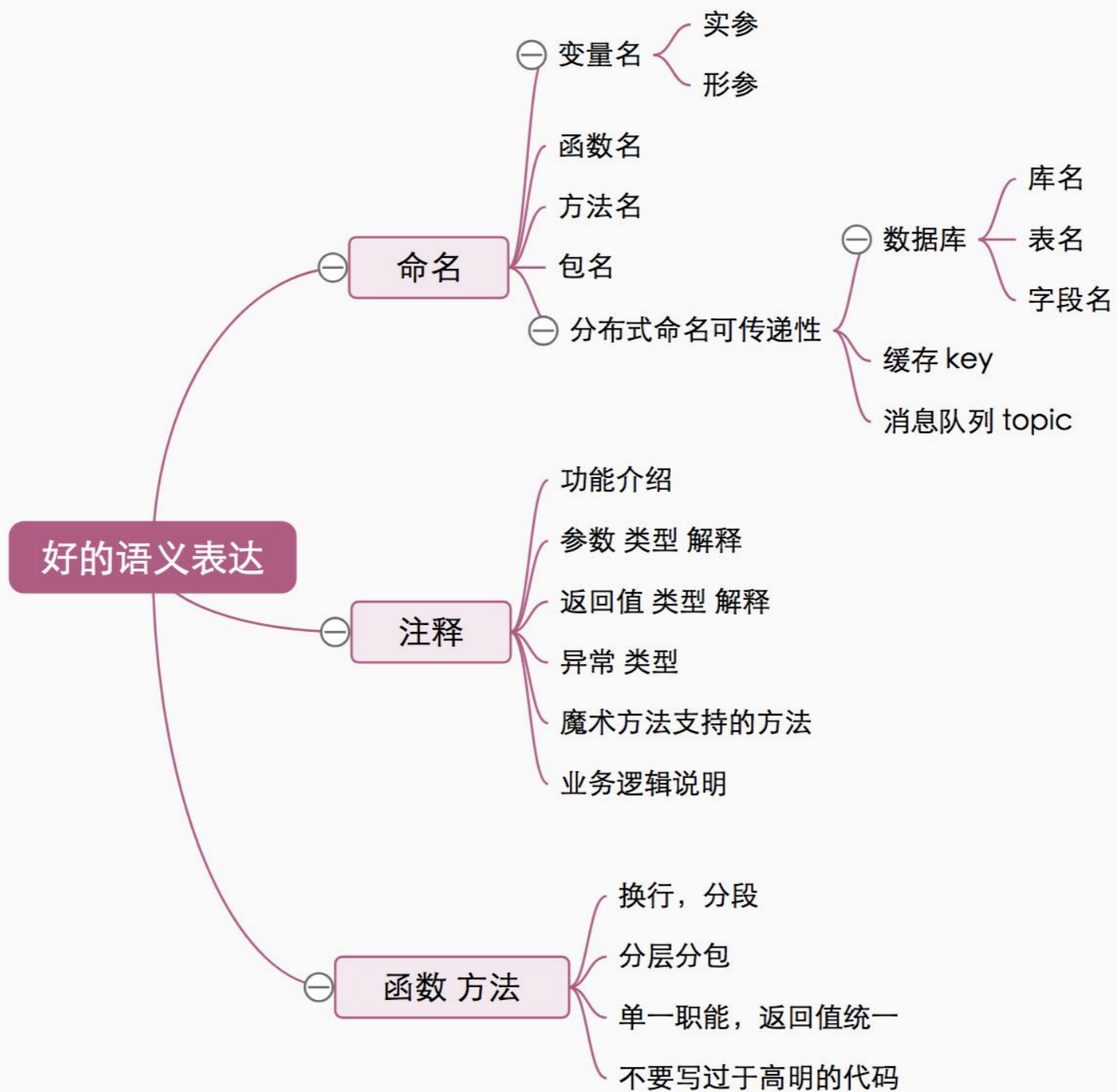


你有担心过别人看到你代码会骂你吗？





怎么做好语义表达





你有注意自己代码的语义表达吗？

```
function getGoods($query, $shopId)
{
    $g = Goods::add($query["uid"], $query["name"]);
    return Shop::add($g, $shopId);
}

class Goods
{
    public static function add($uid, $name)
    {
        $id = mt_rand(1, 100000);
        return $id;
    }
}

class Shop
{
    public static function add($goodsId, $shopId)
    {
        $id = mt_rand(1, 100000);
        return $id;
    }
}
```



你有注意自己代码的语义表达吗？

第一印象以为是在某个商店里根据某些条件搜索获取商品列表

明明是写操作为什么用 get ?

query 是什么鬼？
query 不是查询吗？
模糊 kv 是噩梦



返回的结果是什么类型？能在返回值变量名里体现出来吗？

```
function getGoods($query, $shopId)
{
    $g = Goods::add($query["uid"], $query["name"]);
    return Shop::add($g, $shopId);
}

class Goods
{
    public static function add($uid, $name)
    {
```

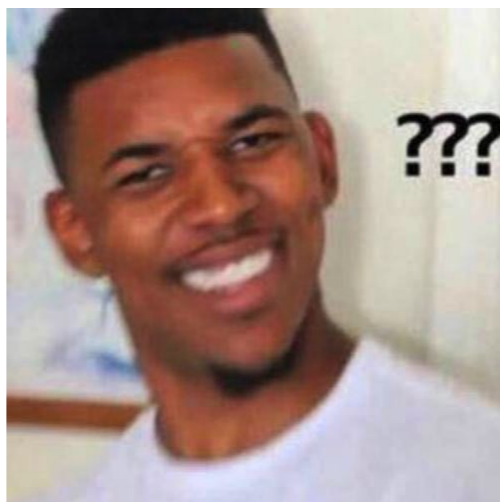



你有注意自己代码的语义表达吗？

```
function getUserInfo($steamId, $youId = [])  
{  
  
}
```



你有注意自己代码的语义表达吗？



获取列表加上 list 关键字是否更好，否则容易以为是单个对象

you 原来是已经有的意思，也就是需要被排除的 uid 集合

```
function getUserInfo($teamId, $youId = [])  
{  
  
}
```



你有注意自己代码的语义表达吗？

```
class Db
{
    /**
     * @param string $table 数据库表名
     * @param array $data 新增数据
     *
     * @return int 新增主键
     */
    public static function insert(string $table, array $data)
    {
        $id = mt_rand(1, 1000);
        return $id;
    }
}

class ViewLogStore
{
    private $table = "view_log";

    function setHistory($data)
    {
        Db::insert($this->table, $data);
    }
}
```

set 一般操作 k-v 返回 bool
history 是想表达历史记录吗？直接省略
这里明明是新增，可以用 insert, record

又是模糊k-v的数据写入



分布式系统命名的传递性

```
class WechatUserModel{  
  
}  
  
class WechatGroupModel{  
  
}  
  
class WechatMessageModel{  
  
}
```



分布式系统命名的传递性

```
mysql -uroot -p123456

mysql> show tables like 'wechat%';
+-----+
| Tables_in_test (wechat%) |
+-----+
| wechat_user               |
+-----+
1 row in set (0.02 sec)

mysql> show tables like 'wechat_%';
+-----+
| Tables_in_test (wechat_%) |
+-----+
| wechat_user               |
+-----+
1 row in set (0.00 sec)

mysql> show tables like 'weixin%';
+-----+
| Tables_in_test (weixin_%) |
+-----+
| weixin_message           |
+-----+
1 row in set (0.00 sec)


mysql> show tables like 'wx%';
+-----+
| Tables_in_test (wx_%)    |
+-----+
| wx_group                 |
+-----+
1 row in set (0.00 sec)

mysql> show tables like 'w%';
+-----+
| Tables_in_test (w%)      |
+-----+
| wechat_user              |
| weixin_message           |
| wx_group                 |
+-----+
3 rows in set (0.00 sec)

mysql>
```

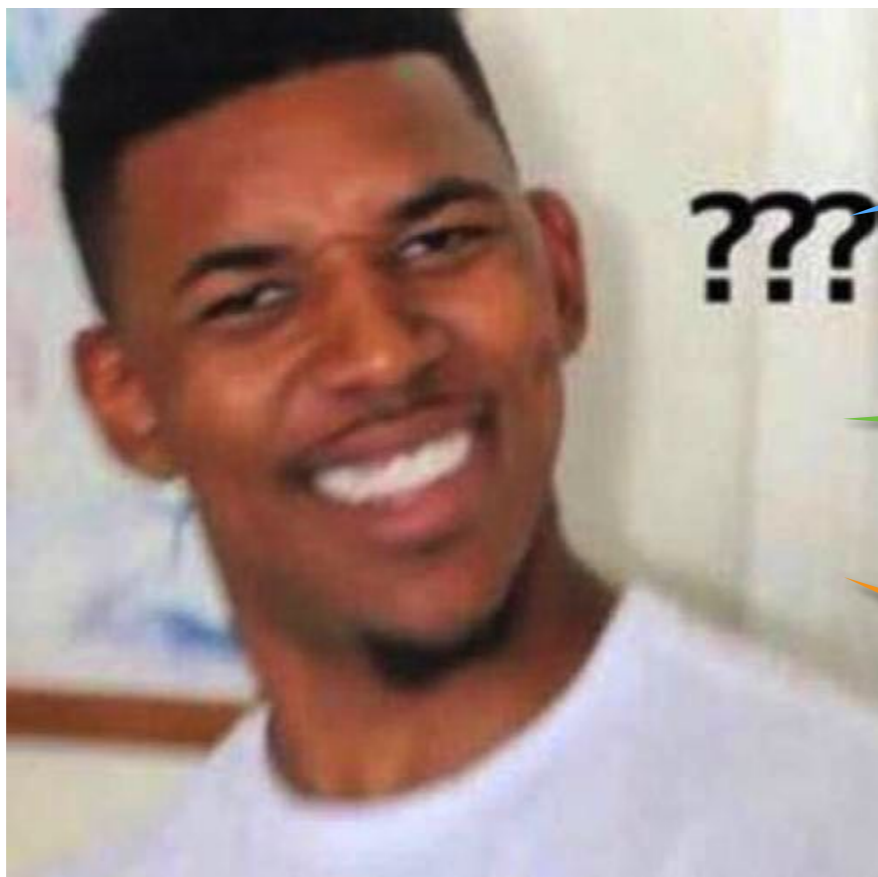
分布式系统命名的传递性

aid 或者 article_id

```
mysql -uroot -p123456   
  
mysql> show create table wechat_article\G;  
***** 1. row *****  
      Table: wechat_article  
Create Table: CREATE TABLE `wechat_article` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `uid` int(11) NOT NULL COMMENT '用户 id',  
  `a_id` int(11) NOT NULL COMMENT '文章 id',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4  
1 row in set (0.00 sec)  
  
ERROR:  
No query specified  
  
mysql> 
```



来自灵魂深处的拷问



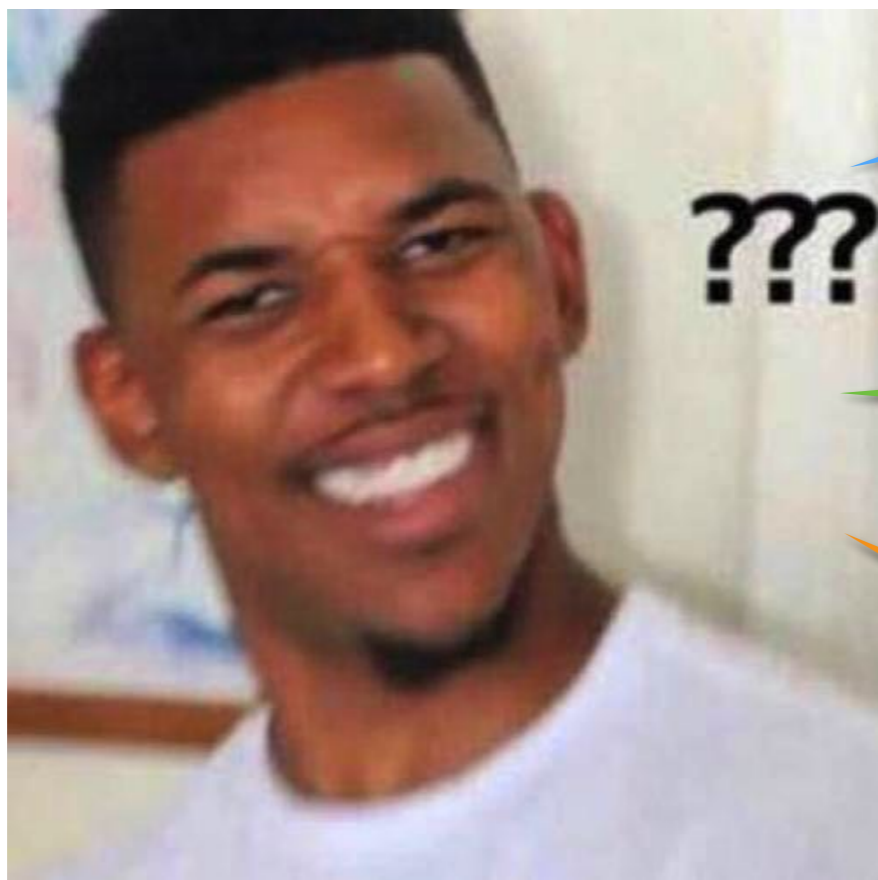
线上故障能根据代码里面的类找到对应的表吗?

陌生人能根据代码里的模块找到所有的表吗?

你在开发的时候有参考之前的一些默认的命名规则吗?



同学你注释玩得 6 吗？



注释里还有什么学问？
Are you kidding me?

一个数组对象成员
你知道怎么写吗？

类的魔术方法调用的注释
你知道怎么写吗？



同学你注释玩得 6 吗？

```
/**
 * @var int $i
 */
public $adsList = [];

/**
 * 资料文档
 * @var string
 */
public $doc;
}

$live = new Live();
foreach ($live->adsList as $ads){
    $ads->
```

- f link
- f img
- f desc
- f title

Press ^Space again to see more variants >>

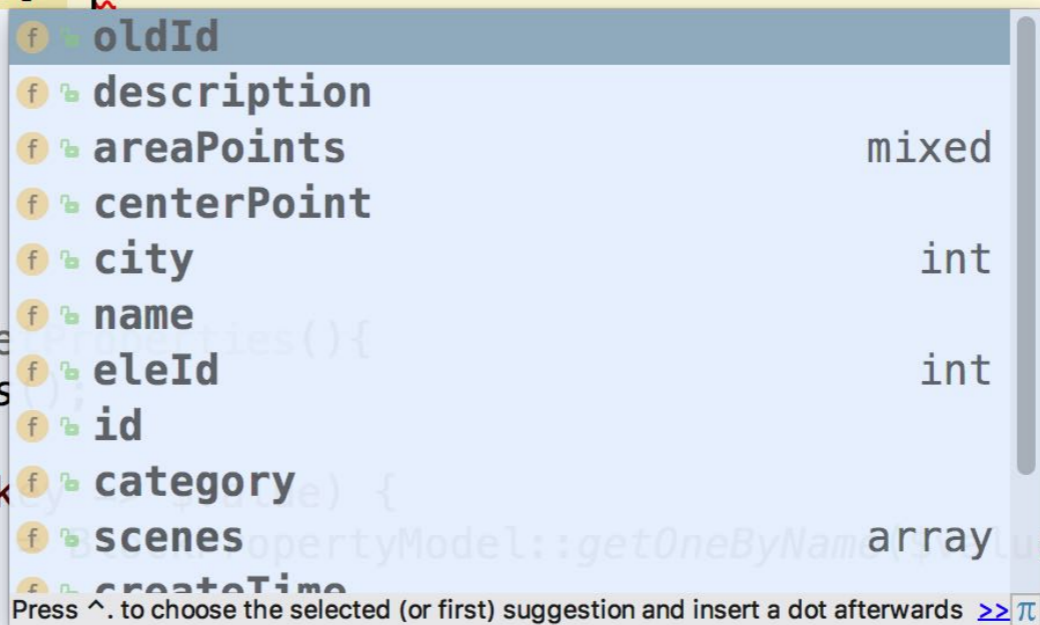
π



同学你注释玩得 6 吗？

```
}  
  
public static function import(array $rows){  
  
    $blocks = [];/** @var $blocks Block[] */  
    foreach ($rows as $row){  
        $oldId = $row["I"];  
        if (!isset($blocks[$oldId])) {  
            $blocks[$oldId] = self::initBlock($row);  
        }else{  
            $blocks[$oldId]->  
        }  
    }  
}  
  
private static function getProperties(){  
    $fields = self::fields();  
    $properties = [];  
    foreach ($fields as $key => $value) {  
        $properties[$key] = $value; // @property int $key  
    }  
  
    return $properties;  
}
```

由于 PHP 的弱类型
更加依赖注释的书写





同学你注释玩得 6 吗?

```
8
9  /**
10 * @link http://manual.phpdoc.org/HTMLframesConverter/default/
11 *
12 * @method static int search(string $query,$limit = 10,$offset = 0)
13 */
14 class SearchServiceProxy
15 {
16     public static function __callStatic($method, $arguments)
17     {
18         if (!method_exists(object: "SearchService", $method)) {
19             throw new \LogicException(__CLASS__ . "::" . $method . " not found");
20         }
21
22         try {
23             $data = call_user_func_array(["SearchService", $method], $arguments);
24         } catch (\Exception $e) {
25             error_log($e->getMessage());
26             return false;
27         }
28
29         return $data;
30     }
31 }
32
33 class SearchService{
34
35     public static function search(string $query,$limit = 10,$offset = 0){
36         return [
37             ["id" => 1, "aaa"],
38             ["id" => 2, "bbb"],
39         ];
40     }
41 }
42
43 $res = SearchServiceProxy::s
44 var_export($res);
45
46
```

@method
标注不存在的方法

search(query : string, [limit : int = 10], [o... int
class
Press ^Space again to see more variants >>



同学你注释玩得 6 吗？

```
32
33 class SearchService{
34
35     /**
36      * @param string $query
37      * @param int    $limit
38      * @param int    $offset
39      *
40      * @return array
41      * @deprecated
42     */
43     public static function search(string $query,$limit = 10,$offset = 0){
44         return [
45             ["id" => 1, "aaa"],
46             ["id" => 2, "bbb"],
47         ];
48     }
49 }
50
51 $res = SearchServiceProxy::search( query: "xxx");
52
53 var_export($res);
54
55
56 /**
57  * Class SearchDemo
58  * @see SearchServiceProxy::search()
59  */
60 class SearchDemo{
61     public static function test(){
62         SearchService::|
63     }
64 }
```

@deprecated
标注方法已废弃

```
m search(query : string, [limit : int = 10], ... array
class
^↓ and ^↑ will move caret down and up in the editor >>
```



注释其他注意事项

注释解释张冠李戴，方法名更新，方法的功能业务注释没更新

复制别人的代码把 @author 信息也复制过来了，错误了还要把锅甩给别人

<http://manual.phpdoc.org/HTMLframesConverter/default/>



函数方法注意事项

```
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
```

```
/**
 * 生成查询所需要的条件,排序方式
 * @param mixed $searchfields 快速查询的字段
 * @param boolean $relationSearch 是否关联查询
 * @return array
 */
protected function buildparams($searchfields = null, $relationSearch = null)
{
    $searchfields = is_null($searchfields) ? $this->searchFields : $searchfields;
    $relationSearch = is_null($relationSearch) ? $this->relationSearch : $relationSearch;
    $search = $this->request->get( name: "search", default: '');
    $filter = $this->request->get( name: "filter", default: '');
    $op = $this->request->get( name: "op", default: '', filter: 'trim');
    $sort = $this->request->get( name: "sort", default: "id");
    $order = $this->request->get( name: "order", default: "DESC");
}
```

驼峰命名

分段分层

单一职责

返回模糊



不好的代码

不妨碍它成为一个优秀的软件

PHP MySQL 烂代码多的去了



减少黑洞代码，减少安全隐患

```
class Db
{
    /**
     * @param string $table 数据库表名
     * @param array $data 新增数据
     *
     * @return int 新增主键
     */
    public static function insert(string $table, array $data)
    {
        $id = mt_rand(1, 1000);
        return $id;
    }
}

class ViewLogStore
{
    private $table = "view_log";

    function setHistory($data)
    {
        Db::insert($this->table, $data);
    }
}
```

set 一般操作 k-v 返回 bool
history 是想表达历史记录吗？直接省略
这里明明是新增，可以用 insert, record

又是模糊k-v的数据写入



改进方案 (1) - 数据格式化校验, 代码简单, 无黑洞

```
class ArrayUtils
{
    public static function fetch($arr, $keys, $setNull = false)
    {
        $ret = [];
        foreach ($keys as $key) {
            if ($setNull) {
                $ret[$key] = $arr[$key];
            } else {
                isset($arr[$key]) && $ret[$key] = $arr[$key];
            }
        }
        return $ret;
    }
}
```

```
class ViewLogStore
```

```
{
```

```
    private $logger;
```



改进方案 (2) - 面向对象改造, 无黑洞, 代码清澈见底

```
72 class Entity
73 {
74     public function toArray(){
75         return ArrayUtils::Obj2Array($this);
76     }
77 }
78
79 class ViewLogEntity extends Entity
80 {
81     /** @var int ...*/
84     private $uid;
85
86     /** @var string ...*/
89     private $url;
90
91     /** @var string ...*/
94     private $referer;
95
96     /** @var string ...*/
99     private $createdTime;
100
101     /** @param int $uid ...*/
104     public function setUid(int $uid){...}
108
109     /** @param string $url ...*/
112     public function setUrl(string $url){...}
116
117     /** @param string $referer ...*/
120     public function setReferer(string $referer){...}
124
125     /** @param string $createdTime ...*/
128     public function setCreatedTime(string $createdTime){...}
132 }
133
134
135 class ViewLogStore
136 {
137     private $table = "view_log";
138
139     function record(ViewLogEntity $viewLogEntity)
140     {
141         Db::insert($this->table, $viewLogEntity);
142     }
143 }
```

完整代码: <https://github.com/zhoumeng kang/bianxingji/blob/master/01/3.php>



这还是函数吗？（不仅仅是语义，属于错误）

```
/**
 * @method mixed fetchList(string $sql, array $argv);
 */
class Model
{
    public function __construct($table)
    {
    }
}

function getUserList($startId, $lastId, $limit = 100)
{
    if ($lastId > 0) {
        $startId = $lastId;
    }

    $sql = "select * from `user` where id > ? order by id asc limit ?,?";

    $model = new Model('user');
    return $model->fetchList($sql, [intval($startId), intval($limit)]);
}
```

重复的参数



尽量减少参数数量，不做参数引用

```
function bad($input1, $input2, &$input3)
{
    //...logic

    $input3 = "xxx";

    return true;
}
```

mysql php 源码很多这种套路做法也违反了单一原则



参数类型明确，返回值类型明确，不要出现 mixed

```
str_replace ( mixed $search , mixed $replace , mixed $subject [ , int &$count ] ) : mixed
```

参数

如果 **search** 和 **replace** 都是数组，它们的值将会被依次处理。

search

查找的目标值，也就是 *needle*。一个数组可以指定多个目标。

replace

search 的替换值。一个数组可以被用来指定多重替换。

subject

执行替换的数组或者字符串。也就是 *haystack*。

如果 **subject** 是一个数组，替换操作将遍历整个 **subject**，返回值也将是一个数组。

count

如果被指定，它的值将被设置为替换发生的次数。

返回值

该函数返回替换后的数组或者字符串。

返回值统一，写的是函数（方法）还是 api ？

```
function addUser($nickname)
{
    if (wordsCheck($nickname)) {
        return array("error", "有敏感词");
    }

    if (duplicateCheck($nickname)) {
        return array("error", "昵称被占用");
    }

    return 9527;//id
}

/**
 * 是否有敏感词
 *
 * @param $string
 *
 * @return bool
 */
function wordsCheck($string)
{
    return time() % 2 == 0 ? true : false;
}

function duplicateCheck($string)
{
    return time() % 2 == 0 ? true : false;
}
```

```
function joinTeam($uid, $teamId)
{
    return true;
}

function addUserBiz($nickname, $teamId)
{
    $res = addUser($nickname);

    if (is_array($res)) {
        return $res;
    }

    joinTeam($res, $teamId);

    return $res;
}

function addUserAction()
{
    $res = addUserBiz("xxxx", 1);

    if (is_array($res)) {
        $res["success"] = false;
        echo json_encode($res);
    }

    echo json_encode(["success" => true, "id" => $res]);
}
```

返回值统一，写的是函数（方法）还是 api ？

```
class BizException extends RuntimeException{
}

function addUser($nickname)
{
    if (wordsCheck($nickname)){
        throw new BizException("昵称非法");
    }

    if (duplicateCheck($nickname)) {
        throw new BizException("昵称被占用");
    }

    return 9527;//id
}

function wordsCheck($string)
{
    return time() % 2 == 0 ? true : false;
}

function duplicateCheck($string)
{
    return time() % 2 == 0 ? true : false;
}
```

```
/**
 * @param $nickname
 * @param $teamId
 *
 * @return int
 * @throws BizException
 */
function addUserBiz($nickname, $teamId)
{
    $res = addUser($nickname);

    joinTeam($res, $teamId);

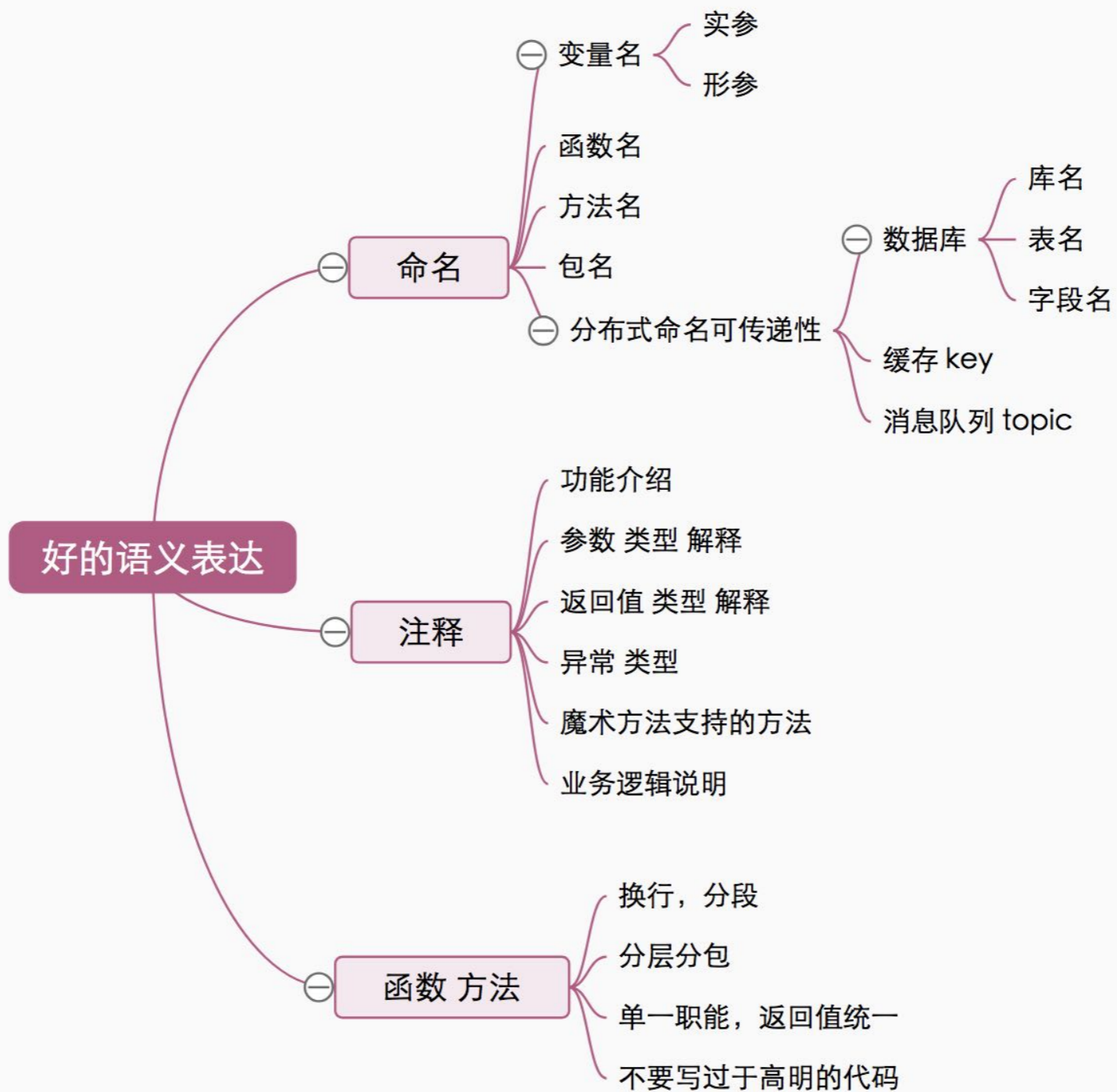
    return $res;
}

function addUserAction()
{
    try{
        $res = addUserBiz("xxxx", 1);
    }catch (BizException $e){
        $res["success"] = false;
        echo json_encode($res);
    }

    echo json_encode(["success" => true, "id" => $res]);
}
```



怎么做好语义表达



**今天主要聊代码的语义表达
都是细节**

接着聊一些代表性的细节



你的代码运行够安全吗？ (广义的安全)

XSS

SQL
Inject

CSRF

Web
Shell

SSRF

越权



你的代码运行够安全吗？（广义的安全）



你打我啊

今天
都不讲!!!



你的代码运行够安全吗？

```
function tags1()
{
    return array(1 => "java", 2 => "php");
}

function tags2()
{
    return array(1 => "java", 3 => "python");
}

function tags3()
{
    return time() % 2 == 0 ? array() : null;
}

function getTopTags()
{
    $tags1 = tags1();
    $tags2 = tags2();
    $tags3 = tags3();

    return array_merge($tags1, $tags2, $tags3);
}
```

需要去重吗？

考虑过 null 的感受吗？



你的代码运行够安全吗？

```
$tag = "互联网产品、";  
$text = rtrim($tag, ",");
```

```
print_r($text);
```

品，实际结果是 互联网产。为什么会这

我们可能以为会得到的结果是 互联网产
样呢？



忽视并发情况

```
class Bootstrap{

    public function __construct()
    {
        $this->init();
    }

    public function init(){
        $uid = decodeCookie();
        checkAndRegister($uid);
    }
}

function decodeCookie(){
    return 9527;
}

function checkAndRegister($uid){

}
```



业务更新伴随 sql 无视性能

```
/**
 * @method mixed fetchList(string $sql, array $argv);
 */
class Model
{
    public function __construct()
    {

    }
}

function getHotTags($num)
{
    $sql = "select * from tag order by follow_num desc limit ?";
    $model = new Model();

    return $model->fetchList($sql, [intval($num)]);
}

function getHotTags2($num, $type)
{
    $sql = "select * from tag where tag_type=? order by follow_num desc limit ?";
    $model = new Model();

    return $model->fetchList($sql, [intval($type), intval($num)]);
}
```

```
mysql> explain select * from tag where tag_type=? order by follow_num desc limit 10;
```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	tag	ALL	NULL	NULL	NULL	NULL	32161	Using where; Using filesort

```
1 row in set (0.00 sec)
```



项目上线有检查代码语法错误吗？

PHP 继承类语法错误导致的故障

错把数组当对象

文件错误没有覆盖

有做静态扫描吗？静态扫描能覆盖全？

什么是世界上最好的语言？

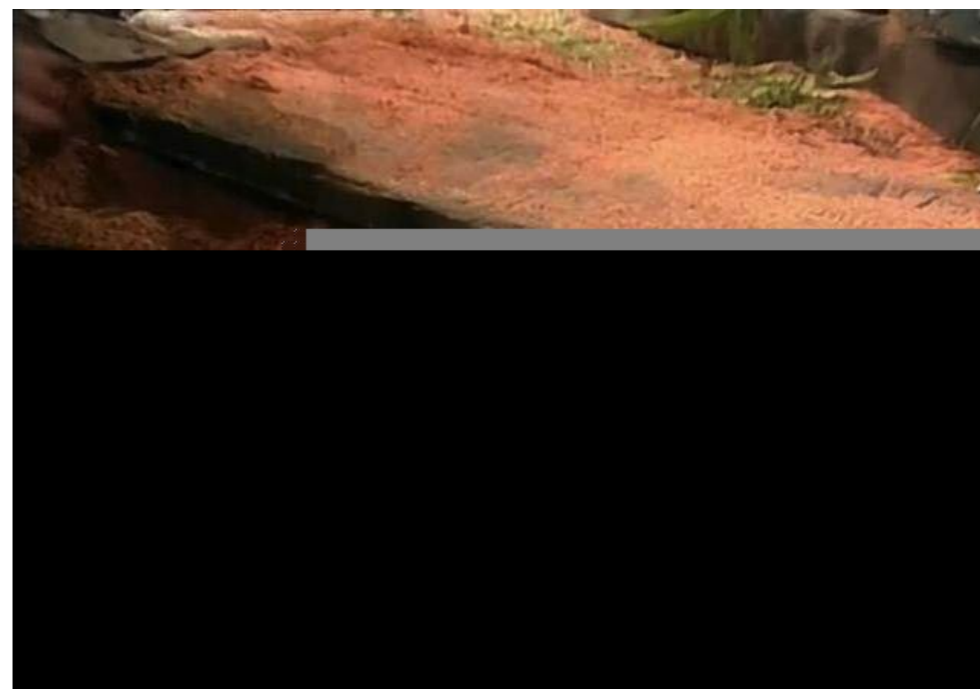
兄弟们下班了告辞！



默默的继续加班



给产品经理削铅笔



**企业级产品里 PHP 就像一个玩具
不过你知道乐高吗？**

