每次拿到返回的数据  
都要和后端写的apl里的返回值做对比  
避免后端出现错误  
  
在将json字符串转换成json对象时 如果报错  
可能是json是非标准的字符串所以转错  
  
只能转换标准的字符串，也就是key value都是双引号形式  
  
在企业开发中，后端传过来非标准字符串，对方没错，别找对方改，需要我们通过方法自己转换

eval(不用)

*eval的作用就是把字符串解析成js代码并执行*

通过 eval()可将非标准字符串转换为json对象  
，eval是evalue的缩写，缩写也读evalue 这是window的方法可直接调用  
内传参str字符串  
注意只传字符串会报错 eval()内字符串首尾用()拼接例eval( "("+str+")" )这样写  
加括号作用使它为表达式 从而将内部解析成对象  
此方法能转非标准的也能转标准的 是万能用法 但是将代码当成js来执行如果用户传不好的字符串就很危险  
  
后端没把用户输入的内容返回给我们 是因为用户输入的内容可能很多 此时传输数据的体积就会变大  速度慢 不传自己也能拿到 可以提高我们的性能

可用each方法遍历数据 添加节点

Cookie

document下的方法cookie可缓存数据到浏览器 默认是一次性会话 即关闭页面就会删除缓存的数据 不关闭就哪怕刷新也还在

使用方法：

存储：

document.cookie(“key=value”);传键值对形式 键为名 对为值

获取：

document.cookie；

上面提到cookie是一次性对话关闭浏览器就删除 但是也可设置过期时间 没过期就不删除（哪怕关闭了浏览器），过期了就删除（哪怕浏览器没关闭）

document.cookie 将以字符串的方式返回所有的 cookie，类型格式： cookie1=value; cookie2=value; cookie3=value;

方法

document.cookie="username=John Doe; expires=Thu, 18 Dec 2043 12:00:00 GMT";

设置是expires后等于一个date时间 和键值对用分号隔开 如果是变量就要用字符串拼接且需要用toUTCString() 方法根据世界时 (UTC) 把 Date 对象转换为字符串。必须转。

date时间设置方法如下

let d = new Date();

d.setDate(d.getDate()+1)

此表示 设置为当前天数的第二天即明天 d.getDate()获取当月的某一天 返回天数 天数+1即第二天

d.setDate（）参数内传数字 一个月中天数的数字 表设置当前月的某一天 其他年月份不变 只天数变

此方法设置后 d的值会改 expires=d.toUTCString()就行

如果将上面的+1改为-1即昨天 哪怕不关闭页面也会删除缓存

删除 cookie 非常简单。您只需要设置 expires 参数为以前的时间(记得转utc)即可



任何cookie大小超过限制都被忽略，且永远不会被设置。

cookie不能一次性设置多个，只能一个一个的存储。

cookie作用范围是:

1. 同一浏览器同一路径下可访问，此处的同一路径指的除了文件名不同，前面的路径都一样，即在同一文件夹下。（浏览器控制台的path中可查看）
2. 浏览器默认情况下，路径的下一路径也可访问。但上一路径不可访问。
3. 若想在上一路径也可访问，在浏览器中通常会设置为根路径 即上级下级都能访问

Document.cookie = “username=zhangsan;path=/”;(用分号隔开 最后一个可不加分号)

如上需设置path=/

设置path=/ 只能让在上级下级都能访问 但是如果在二级域名网站存的 另一个二级域名网站是无法访问的 如果想让同一站点 二级域名不同情况也能访问cookie 就需要设置domian=根域名

所有设置都在document.cookie=””字符串里设置 分号隔开

设置了domian=根域名后 下面的所有二级域名都可以访问

例二级域名 edu.it666.com 和[www.it666.com](http://www.it666.com) 都是二级域名 则设置（译域名）domian=it666.com根域名

综上cookie除了设置过期时间 还需要设置path译路径 和domian 此操作才是完整操作

获取当前页面的默认路径（含文件名）

window.location.pathname

获取当前页面的域名

document.domian

PHP

1. php在浏览器打印东西乱码

原因:浏览器默认编码格式是GBK 后台php编码格式是utf-8 所以会乱码

解决:在php中写此代码:前表文本类型 后表字符集 让浏览器根据此类型和字符集运行

*设置服务器响应的文件类型*

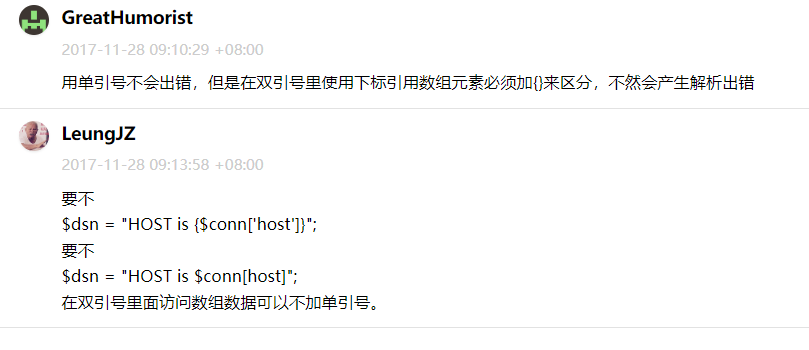
header("Content-Type:text/plain; charset=utf-8");

注意要打印标签时加上面这句 会当做文本输出 把plain改成html就好

1. 在php中 打印时 双引号中 数组下标不加单引号 否则报错

不加会查找变量 变量没有就当字符串用

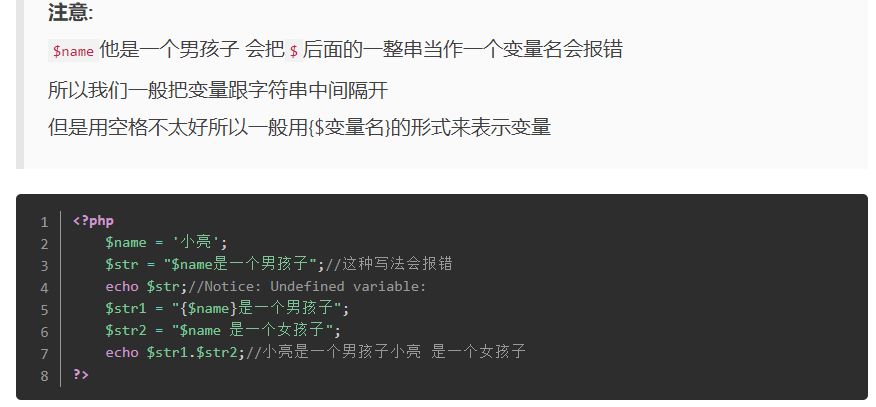
若一定要加单引号就用{}包裹



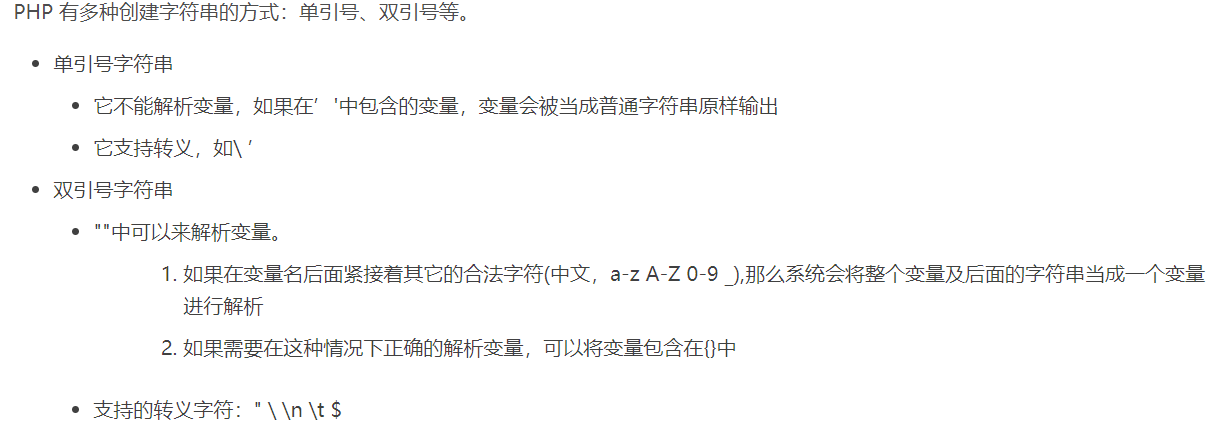
1. 字符串拼接

<https://www.jianshu.com/p/0866aed7ef54>

<https://blog.csdn.net/weixin_45520598/article/details/100061839>



综上用双引号拼接时 都用{}把变量包裹起来 避免出错

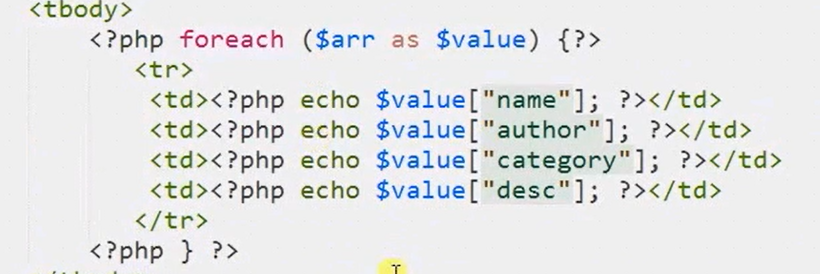


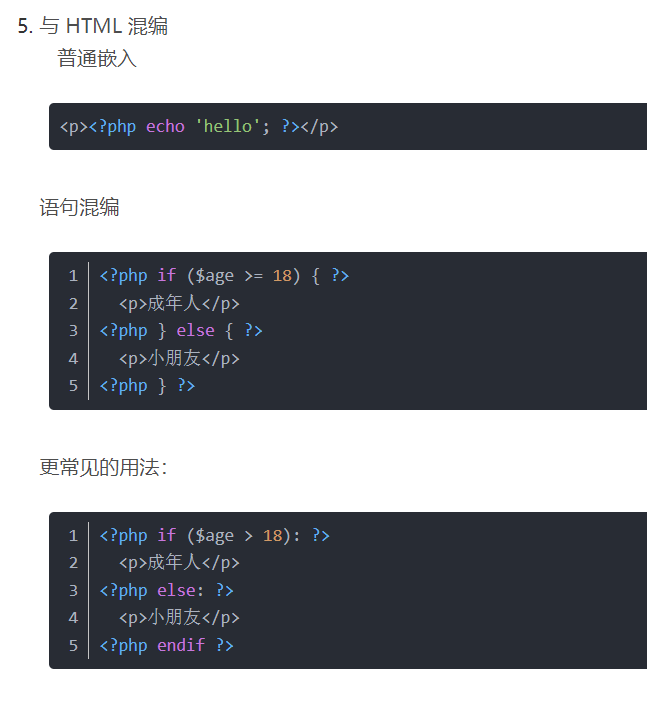
混编

正常输出内容需要拼标签 如果不想拼就用混编这样看着更方便 类似模板

注意:有关php的都用<?php ?>包裹起来 需要输出的用echo <?php ?>首尾和内容都要用空格隔开 echo空格内容再分号结束 空格和分号必须写否则语法错误

前端就是写页面 后端就是拿到前端的页面 将需要动态生成的用后端语言在拿到的页面上更改进行动态生成 下面这部分就是后端操作

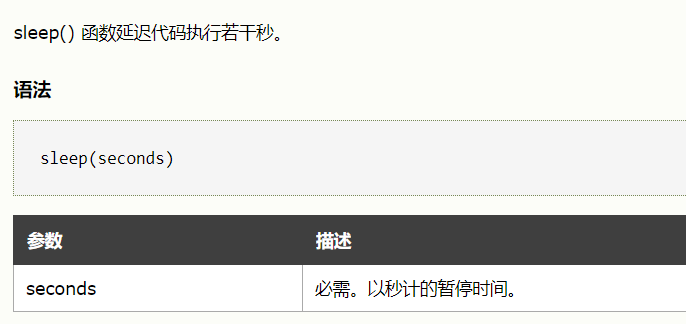




1. json\_encode($arr)

此方法将数组或对象转成json形式字符串 若数组下标为012则返回原数组 若是自定义下标则和json一样(前提下标和值都是双引号)

5.sleep()函数暂停多少秒执行

例: //暂停 10 秒 sleep(10);

encodeURI()

在IE浏览器中get请求参数(post不会)为中文会乱码 所以需要此操作转码

encodeURI() 函数可把字符串作为 URI 进行编码。

**语法**

encodeURI(*URIstring*)

|  |  |
| --- | --- |
| **参数** | **描述** |
| URIstring | 必需。一个字符串，含有 URI 或其他要编码的文本。 |

同步和异步的区别

同步:等你做完了我再做下一步 没做完我就一直等待 阻塞

异步:各做各的 你做你的 我不会等 等你做好了 我收到通知就来处理 不阻塞

在ajax中,就是浏览器把这件事交给ajax来做,且安排好了收到响应后要执行的回调函数,不等待继续做自己的,ajax和服务器进行来往,收到响应后直接执行,两者不会阻塞.

之所以刷新页面是浏览器要等待 就要让页面为空白 拿到数据 再完整加载一次

不等待 就不用让页面空白 继续做自己的 等你回来再刷新局部的

单线程+事件队列

*单线程+事件队列*

*事件队列中的任务执行的条件：*

*1、主线程已经空闲*

*2、任务满足触发条件*

*1、定时函数（延时时间已经达到）*

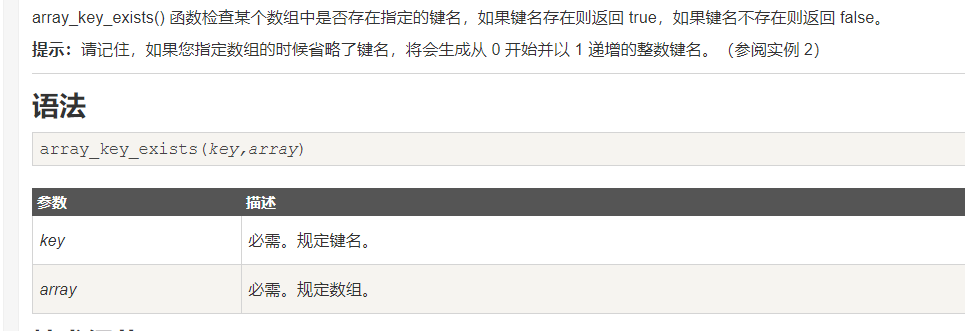
*2、事件函数（特定事件被触发）*

*3、ajax的回调函数（服务器端有数据相应）*

只有以上三函数属于事件队列里,每次主线程空闲后,都会有循环机制看任务队列有没有满足任务的,如果有再拿出来执行.循环机制会重复此操作

php方法 array\_key\_exists(ley,array)

检查数组里是否有指定的键 有就返回1 没有就空



跨域



jsonp

Jsonp(JSON with Padding) 是 json 的一种"使用模式"，可以让网页从别的域名（网站）那获取资料，即跨域读取数据。为什么我们从不同的域（网站）访问数据需要一个特殊的技术( JSONP )呢？这是因为同源策略。

jsonp所有浏览器都支持.无兼容性问题

script的src属性可以实现跨域请求 获取资源,src不仅可以获取js文件 还可以获取php文件传输的数据.

script有静态写和动态创建两种.

静态:直接写死的.若后台传变量回来,在此script标签下面的script可以直接打印变量,相当于获取的变量在全局声明了,需先声明后调用,即对位置有要求,发请求的script在前面.此为弊端,对src传参也不方便.

*script标签里面的async属性表示异步加载资源，默认情况下是同步加载*

只写个async就行

例:  <script async type="text/javascript" src="http://tom.com/data.php?flag=1">

一般不写异步 因为异步请求会导致调用时可能数据还没回来

jsonp原理如下

现在一般用动态创建

即动态创建标签 设置src属性添加到页面 但是:

*动态创建script标签发出去的请求是异步请求 这就是规则 不可修改*

所以就会导致console.log()会报错

技巧解决.服务器传回来的是函数调用,而我们在页面写的函数,两个是同一函数,这样就不存在异步影响,

只有服务器数据返回回来才会调用这个函数.

注意:1.是服务器响应的内容调用的.

2.服务器返回的字符串形式的js代码,会自动解析为js代码.例若为json字符串形式对象,直接返回对象

前面提到两个函数名字相同才可以使用回调函数,这样还是不方便,函数命名一般由前端决定,所有前端通常在url路径后传参键和值,值是函数名前端自己定,键就要和后端统一,因为后端需要get键名来获取值 即得到对应回调函数名字.

这种形式就是jsonp 本质,动态创建script标签,通过src属性发送异步跨域请求,服务端响应的数据就是函数调用,

所以发送请求前必须声明一个函数,且函数名字和参数中传递的名字一致,这里声明的函数由服务器响应的内容调用,(实际是一段js代码)来调用

jquery中的jsonp方法

通过$.ajax()使用jsonp,虽然看着和ajax一样 但是内部原理完全不同,使用时把dataType类型写为jsonp,前面有提到传参键和值指定回调函数.此时通过jsonp属性指定键,jsonpCallback属性指定值,这是两个属性,分开写,此键和值主要是为了发送请求时传参给后台 让后台通过键获取值,根据值写回调函数名字.

如果还需要传业务层面数据例账号密码 就通过data来写.指定回调 和传账号密码是不同的 所以需要分开 别混淆.

jsonp和jsonpCallback是为了把键与值 传给jquery方法 jquery内部根据我们要求 拼接到url后传给后台.所以中间隔了一层

如果不传参,jquery中默认键是callback 值是随机生成的.

注意:有时候成功了,但是在success中得不到数据,需要在error中取;

不需要传type:post或get 默认就是get请求.传post也没用 因为jsonp只能通过get请求 url后拼接参数发送请求 其他不行.

若传多个参数时,后端怎么知道通过哪个参数键获取值即回调函数的名字?

所以后端接口的参数键是写死的,前端参数键需要和后端的键写得一样,然后后端通过该键获取名 来设置回调函数名 这里要注意

参数键配合后端写 回调名后端配合前端写

jsonp对同源的数据也能获取 即跨域和同源都能获取,注有时候跨域获取数据会失败 是因为对方接口设置了不允许跨域获取 没开放此权限.