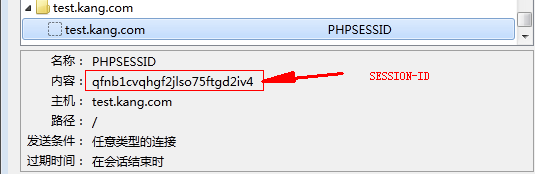
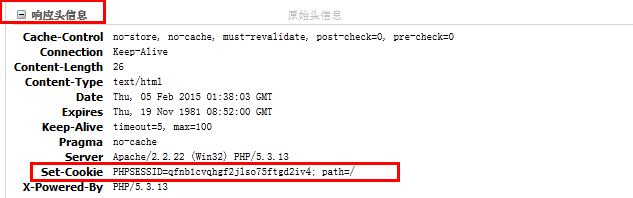
# SESSION

## Session原理

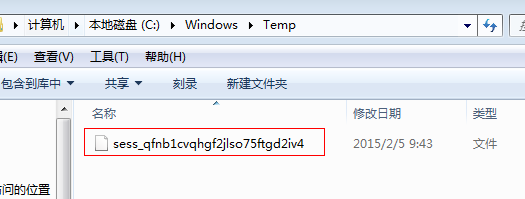
存储于浏览器端Cookie中的session-id，就是一个普通的cookie变量（在session机制中尤其特殊的含义）



也是在响应时将session-id cookie变量PHPSESSID设置好：



每个会话，所生成存储于服务器端的session数据区  
默认的，以文件的形式，存储于服务器端操作系统的临时目录中：



## Session数据属性

### 有效期

会话周期结束

### 有效路径

整站有效

### 有效域

当前域

### 是否仅安全连接传输

否

### 是否仅为HTTP使用

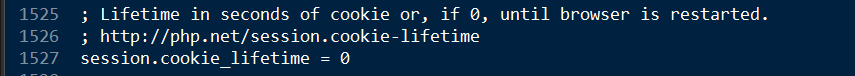
### 属性的管理

以上session数据的特征，都是由，浏览器cookie中所存储的session-idcookie变量的特征所导致的。

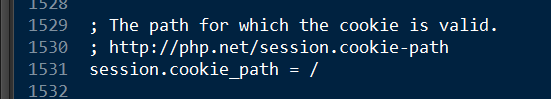
可见，如果需要更改session数据的属性，则需要更改存储session-ID的cookie变量PHPSESSID的属性：

Php.ini中存在该属性的配置：

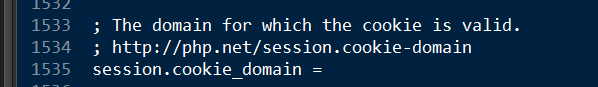
有效期：



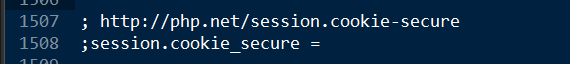
路径：



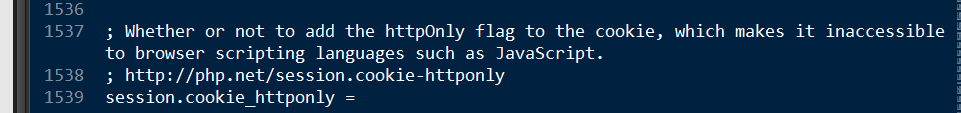
有效域：



仅安全连接传输：



HTTPONLY：



如果需要对默认属性加以修改，办法如下：

A计划：

更改php.ini配置文件即可（不建议）

B计划：

通过在脚本中，使用函数ini\_set()来进行配置的修改，仅在设置后的脚本周期内有效，要保证在开启session前设置完毕(可以选择)

Ini\_set(‘选项’, ‘值’)

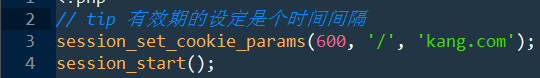
C计划：（推荐）

使用特定功能函数:

session\_set\_cookie\_params(有效期, 有效路径，有效域，是否仅安全传输，是否HTTPONLY)

完成设置。

在开启session前设置完毕



注意：

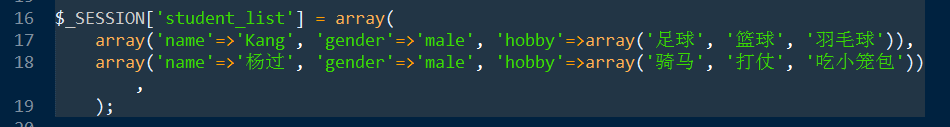
选项的设置是针对cookie中的sessionID，因此是针对所有session数据的。

Tip：

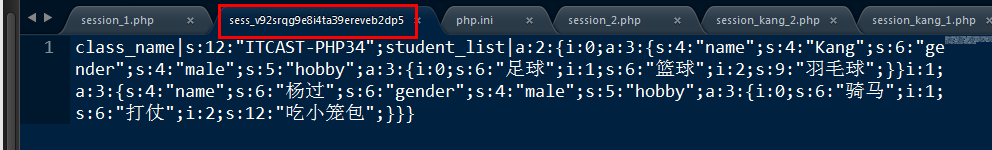
Session属性通常都会保持其默认值，不建议修改！

## Session使用语法问题

Session数据可以是任意的数据类型



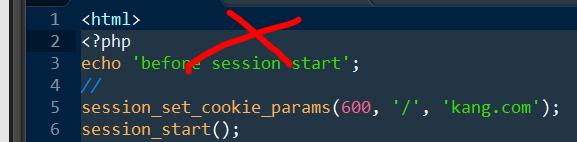
就是因为，session数据区，内的数据是序列化后才存储的！



$\_SESSION数组元素的下标，仅仅可以是字符串类型



Session\_start()，类似于header函数，前不应该有输出！



## Session数据区

在脚本周期外，持久存储当前会话session数据的区域。

在脚本周期内，使用$\_SESSION这个变量管理的会话session数据。

观察：$\_SESSION与session数据区的交互情况：

## Session销毁

使用函数

session\_destroy()

完成。

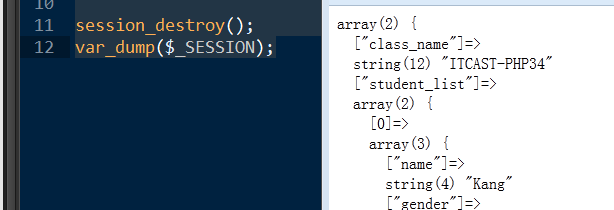
销毁：删除当前session对应的数据区，关闭session机制！



关闭session机制后，导致余下的session操作都不处理！

注意：

$\_SESSION 变量，在销毁session后，是不会自动消失，但结束不完成写操作。因此下次脚本周期，就不能获取到存储的session数据。



如何完整删除与当前session相关的全部数据？

Session\_destroy(); 数据区

Unset($\_SESSION); 销毁变量

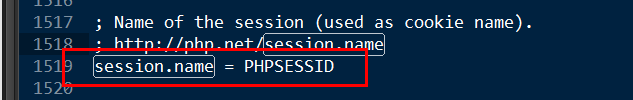
//setCookie(‘PHPSESSID’, ‘’, time()-1), 销毁cookie中的session-iD

setCookie(session\_name(), ‘’, time()-1), 销毁cookie中的session-iD

tip：

PHPSESSID，称之为session.name 可以被配置的php.ini

使用函数session\_name()来获取当前的值！



如何清空session数据？

不要 unset($\_SESSION)

需要：$\_SESSION = array();

## 重写session的存储机制

目的：

1. 便于管理大量的session数据
2. 便于web服务器集群共享session数据。

方案：

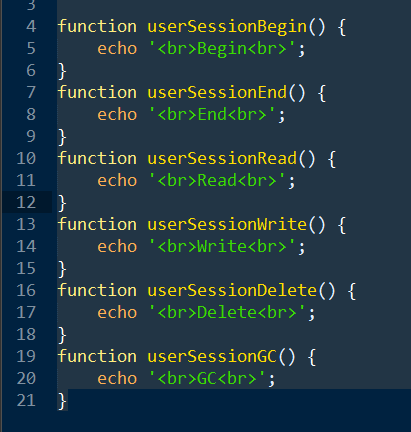
入库，入内存

实现过程：

定义（实现）自定义的相关的存储处理函数

将其设置为session机制需要的存储函数(告知给session机制，使用我们的函数完成存储处理)

Session机制，共需要6个存储处理函数



设置session的存储机制函数：

使用PHP函数：

Session\_set\_save\_handler(开始处理器，结束处理器，读处理器，写处理器，删除处理，垃圾回收处理器)



常规使用session机制即可：

