

PHP

WEB服务器

WEB服务器：负责监听并接收客户端请求，查找需要的文件或数据，发送给客户端。

WEB服务器分为两类：

(1) 静态WEB服务器

提供的内容任何人在任何时间访问都是一成不变的

静态WEB技术：HTML/CSS/JS/图片/音视频/Flash....

(2) 动态WEB服务器：提供的内容不同人在不同时间访问都是可能不同的，一般都需要访问数据库

常见的动态WEB技术：JSP / PHP / .NET / Node.js ...

PHP背景知识

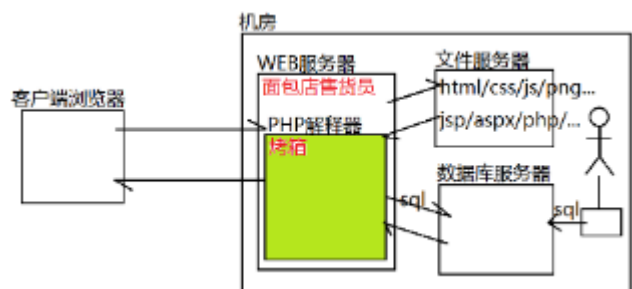
1994年：Personal Home Page诞生，为了记录网页的访问情况；

1999年：更名为PHP is HypertextPreprocessor，成立商业公司来维护该语言。

特点：开源、简单、跨平台、占用资源少、尤其适合于中小型的需要快速开发的WEB项目。

搭建PHP开发环境

输出HelloWorld



(1) 服务器端：安装一款静态WEB服务器，如Apache

c:/xampp/apache/bin/httpd.exe

(2) 服务器端：安装PHP解释器

c:/xampp/php/php.exe

(3) 服务器端：整合配置WEB服务器和PHP解释器

注意：上面三步可以使用服务器端套件来完成，如XAMPP

(4)服务器端：启动WEB服务器，保证端口被打开——售货员开门迎客

(5)服务器端：编写网页保存在客户端可以访问到的目录下

默认路径为c:/xampp/htdocs

(6)客户端：打开浏览器，在地址栏中输入WEB服务器的地址

PHP 中的变量和常量

变量：值可能发生改变的数据，如用户的年龄、商品的价格、用户的密码、帖子的回复数量....

定义一个变量： `$变量名 = 值;`

输出一个变量： `echo $变量名;`

```
$x = 2;
```

```
$y = 3;
```

```
$z = $x + $y;
```

注意：变量名中可以包含数字、字母、下划线，不能以数字开头。

合法的变量名：`$i`、`$user_age`、`$userAge`、`$age2`

非法的变量名：`age`、`$2age`、`$user-age`

PHP 中的数据类型

(1)标量(只有大小没有方向)类型

- `int/integer`：整数，-2147483648~2147483647
溢出就自动变为小数类型。以0开头的整数表八进制；以0x开头表十六进制数字。
- `float/double`：小数，`float`和`double`含义相同，如1.0、1.5、3.5E-5等
- `string`：字符串，注意单引号字符串和双引号字符串问题！
- `bool/boolean`：布尔，
PHP与MySQL不同，有真正意义上的`true/TRUE`和`false/FALSE`！
`false`使用`echo`输出为""，`true`使用`echo`输出为"`1`"

(2)复合类型 `array`、`object`

(3)其它类型 `null/NULL`、资源

PHP 中的运算符

(1) 算术运算符

\+ - * / %(求余/取模) ++(自加) --(自减)

(2) 比较运算符

\> >= = < <= == != =(全等于) !(不全等)

(3) 逻辑运算符

&&(与/并且) ||(或者) !(非/取反)

(4) 赋值运算符

= += -= *= /= %=

(5) 拼接运算符

.(字符串拼接) .=

(6) 三目运算符

?:

PHP 中的逻辑结构

程序 = 数据 + 算法

算法的三种基本结构：

(1) 顺序执行

(2) 选择/分支执行

(3) 循环执行

PHP 中的数组

for循环	foreach循环
for(;;){ }	foreach(\$arr as \$v){ }
可以遍历索引数组	可以遍历索引数组
不可以遍历关联数组	可以遍历关联数组/项目中一般不要此功能
for(\$i=1; \$i<count(\$arr); \$i+=2){ echo \$arr[\$i]; } 可以跳序、倒序访问	只能从第一个到最后一个依次访问
输出1/2/3/4/5	不能输出1/2/3/4/5

PHP 预定义数组

PHP解释器预定义数组变量：

(1)\$_REQUEST：请求，其中保存着“客户端提交给服务器的请求数据”

(2)\$_GET

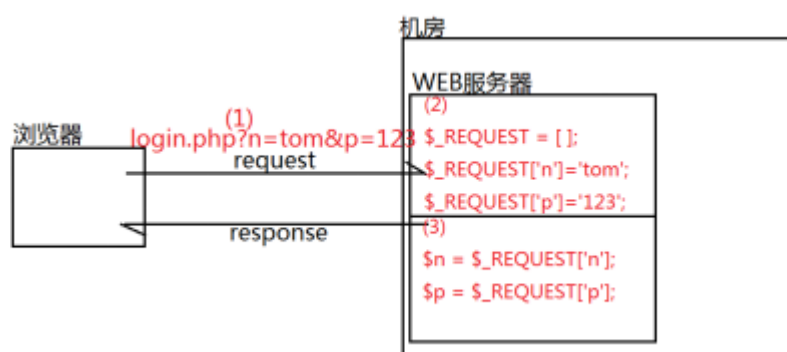
(3)\$_POST

(4)\$_FILES

(5)\$_COOKIE

(6)\$_SESSION

默认初始情况下，上述六个变量都是 array(0){ }



访问一个动态网页的过程：

(1)客户端在浏览器地址栏中输入要访问的地址以及请求数据

<http://127.0.0.1/login.php?uname=tom&upwd=123>

?k=v&k=v&k=v称为查询字符串，本质就是在向服务器提交请求数据。

(2)WEB服务器接收到请求，会裁剪获取其中的请求数据，保存在一个预定义数组中：

```
$_REQUEST = [ ];
```

```
$_REQUEST['uname'] = 'tom';
```

```
$_REQUEST['upwd'] = '123';
```

此步骤是WEB服务器自动完成的，不需要程序员的参与！

(3)编写程序读取客户端提交给服务器的请求数据

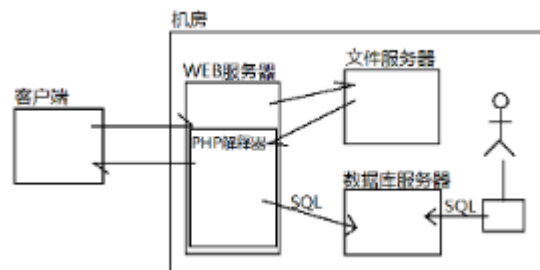
```
 $n$  = $_REQUEST['uname'];
```

```
 $p$  = $_REQUEST['upwd'];
```

编写程序对用户名和密码做处理.....

PHP 预定义函数

—— MySQL连接函数



PHP解释器连接数据库服务器的步骤：

(0)启动数据库服务器、WEB服务器、编写PHP文件

(1)PHP解释器连接到数据库服务器 connect：连接

```
mysqli_connect();
```

(2)向数据库服务器发送SQL查询命令 query：查询

```
mysqli_query();
```

(3)查看SQL命令的执行结果（是成功还是失败？）

```
if(){ }else{ }
```

(4)关闭数据库连接 close：关闭

```
#mysqli_close(); 可以省略
```