复习：

掌握新语言的步骤

(1)了解背景

(2)搭建环境

(3)变量常量

(4)数据类型

标量类型：int / float / string / bool

复合类型：array(索引/关联) / object

其它类型：null / resource

(5)运算符

算术：+ - \* / % ++ --

比较：> >= < <= == != === !==

逻辑：&& || !

位： << >>

赋值：= += -= \*= /= %=

拼接：. .=

三目：?:

其它：=> []

(6)逻辑结构

顺序：

选择：if...else... switch..case..break...

循环：while... do..while... for... foeach...

(7)通用小程序

累加、打星星、九九乘法表、水仙花、质数...

百度：程序员必做的50题

(8)函数和对象

(9)第三方库/框架

(10)实用项目

声明数组： $变量名 = [ ]; $变量名 = [ 值, 值, ... ];

数组长度： echo count( $变量名 );

访问元素： $变量名[下标]=值; echo $变量名[下标];

添加元素： $变量名[ ] = 值;

索引数组的遍历：

for($i=0; $i<count($变量名); $i++){

echo $变量名[$i];

}

关联数组的声明： $变量名 = ['下标'=>值, '下标'=>值...];

关联数组不能使用for循环遍历！

|  |
| --- |
| 结论：PHP中的二维数组（即数组的数组），就对应于数据库/网页中的一个“表格” |

今日目标：

(1)foreach循环

(2)PHP预定义数组 —— 重点&难点

(3)函数 —— 掌握

(4)PHP预定义函数——MySQL数据库访问相关——最重点

1.循环结构之四 —— foreach循环

语法： foreach只用于遍历数组（尤其是关联数组）

foreach( $数组名 as $元素变量名 ){

//循环主体

}

foreach( $数组名 as $下标变量名=>$元素变量名 ){

//循环主体

}

含义：对于数组中的每个元素，都看做是一个指定的变量，对每个这样的变量执行一次指定的循环体

示例： $ageList = [20, 23, 21, 25];

foreach( $ageList as $v){

echo "$v<br>";

}

==================

$user = ['uid'=>101, 'uname'=>'tom'];

foreach( $user as $v){

echo "$v<br>";

}

练习：创建一个数组，表示商城中的一个订单，包含编号、接收人姓名、接收人电话、是否货到付款，使用foreach遍历其中的每个元素。

练习：创建一个数组，保存一个员工的信息，包括姓名、性别、工资、生日、政治面貌，使用foreach输出每个下标和对对应的值。

练习：创建一个数组保存5个学生的高考成绩，使用foreach循环输出每个成绩的下标和对应的成绩。

|  |  |  |
| --- | --- | --- |
|  | for | foreach |
| 索引数组 | 可以！正序、倒序、条序！——灵活但不简单 | 可以！只能正序依次遍历——简单但不灵活 |
| 关联数组 | 不可以！ | 可以！ |

2.PHP中的预定义数组 —— 难点！

预定义：由PHP解释器自己创建的，程序员不需要声明可以直接使用的。PHP预定义了如下的数组变量：

$\_GET

$\_POST

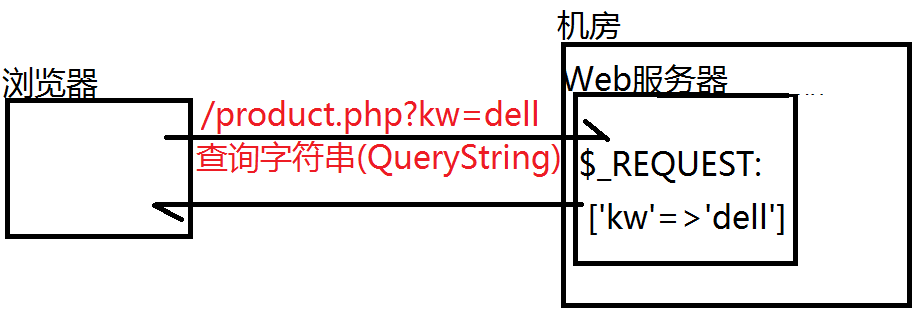
$\_COOKIE

$\_FILES

$\_REQUEST：保存着客户端提交给服务器的“请求”数据

$\_SESSION

上述变量默认都是 array(0){ }



Web项目中，客户端(浏览器)如何给服务器传数据：

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

PHP服务器会**自动**把?后面的数据存储到$\_REQUEST数组中：

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

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

程序员如何读取客户端提交的数据：

**echo $\_REQUEST['uname'];**  //tom

**echo $\_REQUEST['upwd'];**  //123

练习：客户端浏览器访问服务器上的add.php页面，并传输num1和num2两个数字作为请求数据；编写add.php网页，读取客户端提交的请求数据，向客户端输出这两个数字的代数和。

练习：客户端访问star.php，给服务器提交rows和cols表示要打印的星星行数和列数。服务器端编写star.php，读取客户端提交的rows和cols，输出指定行数和列数※

练习：服务器端页面 max.php，接收客户端提交的n1/n2/n3/n4四个请求数据（都是整数），服务器端页面返回这四个数字中的最大值。

练习：服务器端页面leijia.php，接收客户端提交的start和end两个整数请求数据，计算出从start~end的累加和。

3.函数基础认识 —— 掌握

Function：功能体/函数，用于封装一段需要反复执行/不易编写的代码 —— 饺子机。

**声明一个函数：**

function 函数名( ){

//函数主体

}

**执行/调用一个函数：**

函数名( );

练习：创建一个函数 add()，可以计算三个数字的和，并输出；调用该函数两次。

练习：创建一个函数 print99()，其中输出一个九九乘法表；调用该函数三次。

**声明一个带参数的函数：**

function 函数名(参数名1, 参数名2, ... ){

//函数主体

}

**执行/调用一个函数：**

函数名( 值1, 值2, ... );

声明函数时指定的参数其实就是变量名——形式参数；调用函数时给出指定的具体的值——实际参数。

练习：创建一个函数printStar()，可以接收两个参数rows、cols，分别用于指定输出的※的行数和列数，输出※。调用此函数两次，分别输出5\*10的※矩阵、3\*30的※矩阵。

练习：创建一个函数min2()，可以接收四个整数做参数，函数体中找出这四个数中的最小值，并输出。调用此函数三次。

**声明一个带返回值的函数：**

function 函数名( [参数名1, 参数名2, ...] ){

//函数主体

**return**  函数运算的结果值; //应该是整个函数中最后一句

}

**执行/调用一个有返回值函数：**

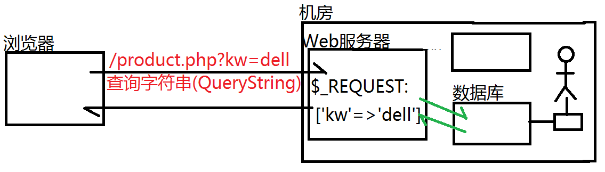
**$变量名 =** 函数名( [值1, 值2, ...] );

使用一个变量接收函数的返回值。

练习：创建一个函数sum( )，接收一个整数数组做形参，返回数组中所有元素的和。调用该函数2次，传入不同的数组做实参，接收其返回值，并输出。

练习：创建一个函数avg( )，接收一个整数数组做形参，返回数组中所有元素的平均值。调用该函数2次，传入不同的数组做实参，接收其返回值，并输出。

4.PHP预定义函数 —— 访问MySQL数据库必需的函数 ——重点&难点



提示：历史上，PHP官方提供了两套访问MySQL服务器的函数：

mysql\_xxxx( ) 性能不够优秀

mysqli\_xxxx( ) Improved：改进提升版

使用MySQLI函数库访问数据库服务器的步骤(与管理员从命令行中访问数据库的步骤一样)：

(1)连接到数据库服务器

$conn = mysqli\_connect(...);

(2)提交SQL语句给服务器执行

$sql = "INSERT/DELETE/UPDATE/SELECT...";

$result = mysqli\_query($conn, $sql);

(3)查看执行结果

if($result===false){ //执行失败

}else{ //执行成功

}

(4)关闭到数据库服务器的连接(可以省略)

mysqli\_close( $conn );

课后练习：

(1)编写网页emp\_add.php，连接到数据库，提交一条INSERT语句给emp表，输出执行结果

(2) 编写网页emp\_del.php，连接到数据库，提交一条DELETE语句给emp表，输出执行结果

(3) 编写网页emp\_update.php，连接到数据库，提交一条UPDATE语句给emp表，输出执行结果

练习目标：尽可能多见各种错误！设法解决！记住这个错误！下次课难度很大！！