php

**一.简介**

1.Personal Home Page Hypertext Preprocessor（PHP超文本预处理器）

2.PHP是一种流行的服务器端脚本语言（可以直接运行不需要编译，但是在服务器上运行，客户端运行不了），适用于Web开发，用于创建网站

3.是一门后端语言

4.文件扩展名  **.php**

<?php

echo "Hello World";

?>

**需要通过服务器来执行**

**文件全部存在路径 /Library/WebServer/Documents/PHPwork**

可以包含html代码，且html中可以嵌入php代码

用echo输出到客户端

echo可以输出HTML代码如echo '<hr>';

**二.变量语法**

（一）变量

$变量名=变量值;

命名规范：1变量名以$开头

2只能由数字、字母、下划线组成，但不能以数字开头

3建议使用英文单词命名

$age=18;

变量

1.不支持同时定义多个变量（$a=5,$b=4;这是不被允许的）

2.可变变量（一个变量的变量名可以动态的设置和使用）

$aa='bb';

$bb='cc';

echo $$aa;

输出的结果就是cc

3.值传递和引用传递

$a=111;

$b=$a; 将值赋给变量，即按值传递

$c=&$a; 将引用赋给变量，即按引用传递（使用&符号）

$a=222;

echo $b; 输出111

echo $c; 输出222

（二）数据类型

1.查看数据类型

**echo gettype($n**)查看数据类型

**var\_dump($n)**输出变量的类型和值

2.数据类型

string字符串、int整型、float浮点型、bool布尔、NULL空类型、array数组、object对象

$a='bb';

$b=18;

$c=18.5;

$d=false;

$e;

$d=[1,2,3,4,5]

3数据类型转化

(1).**(数据类型)变量** 将变量转换为指定的数据类型

(2).转化成bool值---0、空字符串、null会被转化成false、其他值都会转化为true

(3).echo true; 输出结果是1

(4).echo false; 输出的结果不会显示

**三.字符串的处理和常量**

1.字符串的拼接

在php里+不是拼接，**使用点号进行拼接**（点号左右必须有空格）

echo 'bb' . 'cc' 输出bbcc

2.单双引号

区别，双引号里可以写变量，单引号里写什么都是字符串

echo “数字：”. $num;

echo “数字：$num”; 双引号中的变量会被解析

echo ‘数字：$num’; 单引号中的变量不会被解析，当做普通字符串。效率更高

echo “数字：{$num}。”;变量与中文进行拼接时，要用{ }分隔，避免引起歧义

3.常量（不能改）

**define(变量名,变量值);**

define(‘PI’,3,14);

PI=3.14不能改

常量命名规范

（1）所有字母都是大写，多个字母之间以下划线连接

define(“USER\_STATUS\_ENABLE”,1);

（2）不需要以$开头

**四.运算符，选择结构，循环结构**

和js里几乎是一样

**while循环**

$i=1;

$sum=0;

while($i<=100){

if($i%7==0){

$sum++;

}

$i++;

}

**foreach循环数组**

$names=["tom","jack","luse"];

foreach($names as $one){

echo $one;

}

**五.php和html的嵌套**

1.使用php脚本<?php ?>用于执行php代码

<?php

$name = 1;

echo “<h2>姓名：$name</h2>”

?>

<h2>姓名：<?php echo $name; ?></h2>

2.使用php表达式<?=表达式 ？>用于输出表达式的值

<h2>姓名：<?=$name ?></h2>

3.if中嵌套html代码 不能直接写

<?php

$age = 25;

if($age>18){

?>

<h3>你已成年</h3>

<?php

}else{

?>

<h3>你未成年</h3>

<?php

}

?>

**六.数组**

1.定义数组

$arr1=array(值1，值2，值3);

$arr2=[值1，值2，值3];

2.获取元素

echo $arr1[0];

print\_r($arr);输出复杂类型的变量，能看到数组里的元素内容

count($names);输出数组的元素个数

3.可以往数组的末尾加元素

$names[count($names)]=n;

简写：$names[ ]=n;

4.遍历数组

第一种

for($i=0;$i<count($names);$i++){

print\_r($names[$i]);

}

第二种($key是下标，$value是值)

foreach($names as $key=>$value){

print\_r(“$value : $key” . “<br>”);

}

5.键值对数组

$arr = [

键1 => 值1,

键2 => 值2

...

];

6.判断数组是否为空

$nums=[ ];

if($nums){

echo “不为空”;

}else{

echo “为空”;

}

7.数组函数

（1）count($nums);返回数组中元素的个数

（2）list($a,$b,$c)=$names; 解构函数然后赋值

等价于$a=$names[0];

$b=$names[1];

$c=$names[2];

（3）in\_array() 判断数组中是否函授指定元素

in\_array(“jack”,$names);

（4）array\_search()查找指定元素第一次出现的位置，有则返回索引，无则返回false

array\_search(“jack”,$names);

（5）implode()将数组中的元素拆开用指定符号链接，返回字符串

implode(“#”,$names);

（6）explode()将字符串按特定符号拆分成数组

explode(“#”，“hello#world”);

（7）unset($names[0]);删除元素（索引不删，这样无法是for循环，可以重置索引）

（8）array\_values($names)对数组索引进行重新排序

...

**七.内置函数**

（1）phpinfo();返回php的环境信息以及版本信息

（2）header();设置响应头Headers部分

header(“Content\_Type:text/plain;charset=utf-8”); 设置响应类型或字符集，设置成纯文本

header(“location:https://www.baidu.com”); 设置地址栏，这里相当于页面跳转至百度

（3）exit()、die()、return终止脚本的运行，插入到脚本里会终止脚本的运行

exit(“程序终止”);

die(“程序终止”);

return; 写在函数里

（4）isset()、empty() 判断变量有没有设置返回布尔值

var\_dump(isset($a)); 判断变量是否已经设置，并且是非空，这样就返回true；否则返回false

var\_dump(empty($a)); 判断变量是否为空，或者值是否等于false

（5）is\_类型() 判断值的类型 返回布尔值

is\_string(“3”);

string字符串、int整型、float浮点型、bool布尔、NULL空类型、array数组、object对象

（6）sleep() 让程序进入休眠

sleep(3);暂停三秒

（7）rand() 产生随机数

rand(0,9) 产生一个[0，9]之间的整数

（8）round() 四舍五入，默认不保留小数

round(3.1415,3) 保留三位小数

（9）ceil()向上取整 floor()向下取整 ..

（10）echo()、print()、print\_r()、var\_dump() 输出

echo(11);

echo 11,22,33

print(11)

print\_r($nums) 输出复杂类型的变量

var\_dump($a) 输出类型和值

**八.字符串函数**

（1）strlen(“你好hello”)字符串长度，（按字节算长度，中文一个字是三个字节）//11

（2）mb\_strlen(“你hao”)多字节字符串，（按字符处理）

需要启用多字节字符号串处理模块

在终端输入命令

1.查看已启用模块 php -m

2.启用模块（macOS已经开启）

1创建php的配置文件 php.ini

2找到php.ini.default（;开头的是注释，打开）

3复制php.ini.default，重命名为php.ini

4.extension\_dir=“ext”指定拓展模块的目录

5.extension=php\_mbstring.dll 加载多字节字符串模块

6.重启Apache

（3）mb\_substr()截取

mb\_substr(“hello你111”,2,4)在字符串中从索引2开始截取截取四个字节

（4）mb\_strpos()子字符串出现的索引

mb\_strpos(“1234”,4);

（5）str\_replace() 替换

str\_replace(“e”,“x”,“hello”);把e换x

（6）trim()去除首尾的空格

**九.日期函数**

（1）time() 返回自Unix纪元1970年1月1日0时0分0秒至当前时间的秒数

（2）1.临时修改临时时区date\_default\_timezone\_set(“Asia/Shanghai”);设置时区

2.设置全局时区，对所有页面有效

编辑php.ini文件

找到date.timezone 936行

取消注释后修改date.timezone = Asia/Shanghai

重启Apache

（3）date() 对日期时间进行格式化（需要先设置时区）

date(“Y-m-d H:i:s”); 对当前时间进行格式化

date(“Y-m-d H:i:s”,time()-7\*24\*60\*60); 七天前的时间进行格式化

（4）mktime() 返回指定日期的秒数 （参数依次传入时分秒月日年）

$time=mktime(20,00,00,8,8,2008)返回秒值

date(“Y-m-d H:i:s”,$time); 2008年8月8日20：00：00

**十.正则表达式函数**

（1）preg\_match() 判断字符串是否匹配对应的正则规则（类似与js里的text） 返回1表示匹配，不匹配返回0

preg\_match(“/hello$/”,“helloworld”)

（2）preg\_replace() 替换

echo preg\_replace(“/\d{4}$/”,“\*\*\*\*”，“13237463678”) 电话末尾换成\*\*\*\*

**十一.自定义函数function**

1.function 函数名(参数1,参数2,...){

函数体

}

函数名();

2.参数可以有默认值，

function 函数名(参数1,参数2=参数2的默认值.){

函数体

}

此时参数2就有默认值，那调用函数时就可以传，也可以不传参数（必填的参数靠前，有默认值的参数写在靠后）

3参数长度不确定

function fn($a,$b,...$args){

函数体

}

fn($a的参数值,$b参数值,然后可以传入任意个值)

$a和$b是普通参数

$args是一个数组

4.引用参数，在函数形参前添加&，表示引用参数，在函数内对引用参数的修改，在函数外可以获取到,相当于储存的地址相同了，引用参数既可以传入值，也可以返回值

function fn(&$a,&$b){

}

5.函数内不能直接访问函数外的全局变量

如何使用全局变量

$num=5;

function fn(){

global $num; //通过global引用全局变量

echo $num;

}

这样就可以使用全局变量

6.函数的静态变量，只在函数第一次被调用时初始化，之后不再初始化，实现变量共享

function fn(){

static $count = 1; //只有第一调用会执行，这一条之后就不会再执行

echo $count;

$count++;

}

fn(); //1

fn(); //2

7.动态调用函数，根据外部传入的函数名来决定要调用的函数

function hello($name){

echo “hello”;

}

function world($name){

echo “world”;

}

$method = “hello”; //定义函数名字是hello

$name=tom;

method($name);

**十二.访问MySQL**

（1）mysql\_connect(数据库地址和端口，账号，密码，数据库名字); 连接数据库，如果连接成功返回mysqli Object对象，如果连接失败返回false

重新配置root密码

1. mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY '824166449';

2. flush privileges; 更新命令

3. $conn = mysqli\_connect("localhost:3306", "root", "824166449", "test");

<pre>是用于格式化代码

<pre>

<?php

访问数据库

$conn = mysqli\_connect("localhost:3306", "root", "824166449", "test");

print\_r($conn);

?>

</pre>

输出：

mysqli Object

(

[affected\_rows] => 0

[client\_info] => mysqlnd 5.0.12-dev - 20150407 - $Id: 7cc7cc96e675f6d72e5cf0f267f48e167c2abb23 $

[client\_version] => 50012 客户端的版本

[connect\_errno] => 0 连接的错误码0就是没有错误

[connect\_error] =>

[errno] => 0

[error] =>

[error\_list] => Array

(

)

[field\_count] => 0

[host\_info] => localhost:3306 via TCP/IP 主机的地址

[info] =>

[insert\_id] => 0

[server\_info] => 8.0.21

[server\_version] => 80021 服务器的版本

[stat] => Uptime: 1222536 Threads: 2 Questions: 598 Slow queries: 0 Opens: 380 Flush tables: 3 Open tables: 285 Queries per second avg: 0.000

[sqlstate] => 00000 状态

[protocol\_version] => 10

[thread\_id] => 60

[warning\_count] => 0

)

（2）mysqli\_connect\_error()获取连接的时的错误信息

（3）mysqli\_connect\_errno()获取连接时的错误码，如果连接成功返回0，如果失败返回非0的数字

（4）mysqli\_query()执行数据库操作，如增删改查操作，事务操作

1、增删改操作（切记把连接数据成功作为第一个参数，返回一个布尔值，成功返回true，失败返回false）

$sql="insert into dept (deptno,dname,loc) values (50,'市场部','南京')";

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

2、查询操作 （成功返回一个查询得到的mysqli\_result Object对象，失败返回false）

$sql= "select".

" deptno,dname,loc".

"from".

" dept";

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

（5）mysqli\_close()关闭连接

mysqli\_close($conn);

（6）mysqli\_error()获取执行时的错误信息

mysqli\_error($conn);

（7）mysqli\_insert\_id()获取插入时生成的主键，主键会自动增长

$sql= "insert into dept".

" (deptno,dname,loc)".

"values".

" (50, 市场部','南京')";

$result=mysqli\_query($conn,$sql); 获取返回的主键

if($result){

echo mysql\_insert\_id($conn);

}else{

echo mysqli\_error($conn);

}

（8）mysqli\_fetch\_assos()从结果中读取一行数据，每次获取一行，返回一个键值数组，列名为key，列值为value，调用几次返回几组键值数组，当获取不到时，返回null，利用循环获取

$sql= "select".

" deptno,dname,loc".

"from".

" dept";

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

if($result){

print\_r(mysqli\_fetch\_assoc($result));

循环：

while($row= mysqli\_fetch\_assoc($result)){

print\_r($row);

echo “<br>”;

}

}else{

echo mysqli\_error($conn);

}

（9）mysqli\_num\_rows()获取结果集中行的数量

$sql= "select".

" deptno,dname,loc".

"from".

" dept";

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

if($result){

while($row= mysqli\_fetch\_assoc($result)){

print\_r($row);

echo “<br>”;

}

echo mysqli\_num\_rows($result); 获取总行数

}else{

echo mysqli\_error($conn);

}

**十三.事务操作**

transaction

转账是一个事务，由两个sal操作所组成，如张三给李四转账

$conn= mysqli\_connect("localhost:3306", "root", "824166449", "test");

一次性插入三条数据,要么都插入，要么都插入失败

mysqli\_query($conn, "begin"); (开启事务)

$flag=true; (用于标识程序执行是否正常，true表示正常)

$dnames=["a1","b1","c1"];

foreach($dnames as $dname){

$sql="insert into dept"

" (deptno,dname)"

"values"

" (".rand(100,99) . ",'$dname')";

mysqli\_query($conn,$sql);

if(mysqli\_errno($conn)!=0){

$flag=false;

}

}

if($flag){

mysqli\_query($conn, "commit"); (提交事务)

echo "success";

}else{

echo mysqli\_query($conn, "rollback"); (回滚事务)

echo "rollback";

}