性能：

LoadRunner\Jemter\QTP

自动化：

PC端的：Python + Selenium\Python中的unittest模块

APP端的：

抓包工具：

Winshark、Postman

复习Java语法

复习Python语法

复习PHP语法

复习

网络相关知识\TCP的三次握手\Ajax方法\JSON方法

复习：

MySQL数据库

复习：

Linux相关知识

PHP部分知识：

参考菜鸟教程

PHP 是一种创建动态交互性站点的强有力的服务器端脚本语言。

PHP:Hypertext Preprocessor 超文本预处理器

PHP官网：<http://www.php.net>

PHP开发工具包含：

PHP服务器组件；WampServer\XAMPP

PHP IDE；Zend Studio\Eclipse for PHP\PhpStorm

MySQL管理工具；Navicat for MySql

文本编辑器:Notepad++\editplus

PHP基本语法

1. PHP脚本以<?php 开始，以?>结束；
2. PHP文件的默认文件扩展名是”.php”；
3. PHP用分号来分割语句；：
4. 变量以$符号开始，后面跟着变量的名称，如$test = “Hello World!”
5. PHP中的注释：
6. 单行注释：// 、#
7. 多行注释：/\*...\*/

对于初学者建议使用集成的服务器组件，它已经包含了 PHP、Apache、Mysql 等服务,免去了开发人员将时间花费在繁琐的配置环境过程。

①WampServer ②XAMPP

PHP 脚本在服务器上执行，然后将纯 HTML 结果发送回浏览器。

有两种在浏览器输出文本的基础指令：echo 和 print。有什么区别呢？

echo:可以输出一个或多个字符串；没有返回值 ；速度比print快；可以加括号也可以不加括号；

print:只允许输出一个字符串，返回值总为1；有返回值 ；可以加括号也可以不加括号；

PHP变量名命名要注意的点：

变量名不能包含空格；

变量名是区分大小写的；($y和$Y是两个不同的变量)

PHP有四种不同的变量作用域：

Local 局部变量

Global 全局变量 如果要在一个函数中访问一个全局变量，需要使用 global 关键字

Static 当函数执行完了，static修饰的局部变量不会被删除

Parameter

PHP 将所有全局变量存储在一个名为 $GLOBALS[index] 的数组中。 index 保存变量的名称。这个数组可以在函数内部访问，也可以直接用来更新全局变量

echo PHP\_EOL 这个是什么意思呢？

好像是定义一个字符串来的

var\_dump 这个是什么意思呢？

var\_dump() 函数返回变量的数据类型和值

**PHP的数据类型：**

String - 字符串：

$x = “Hello World!” //可以用单引号也可以用双引号

Integer - 整型：

十进制， 十六进制（ 以 0x 为前缀）或八进制（前缀为 0）

Float - 浮点型：

浮点数是带小数部分的数字，或是指数形式。

Boolean - 布尔型：

布尔型可以是 TRUE 或 FALSE。用于判断

Array - 数组：

用array来进行声明

Object - 对象：

在 PHP 中，对象必须声明。

首先，你必须使用class关键字声明类对象。类是可以包含属性和方法的结构。

然后我们在类中定义数据类型，然后在实例化的类中使用数据类型：

$this的用法：

关键字this就是指向当前对象实例的指针，不指向任何其他对象或类

NULL - 空值：

可以通过设置变量值为 NULL 来清空变量数据

常量：

常量值被定义后，在脚本的其他任何地方都不能被改变。

常量名不需要加 $ 修饰符

设置常量，使用 define() 函数

bool define ( string $name , mixed $value [, bool $case\_insensitive = false ] )

常量在定义后，默认是全局变量，可以在整个运行的脚本的任何地方使用。

操作字符串的函数和运算符

并置运算符.用于把两个字符串值连接起来

strlen()函数返回字符串的长度(字符数)

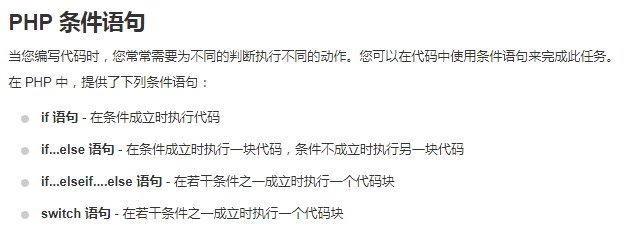
strops()函数用于在字符串内查找一个字符或一段指定的文本

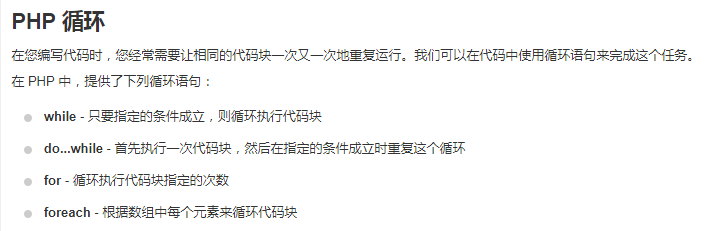
运算符

x ===y 绝对等于的意思 如果x等于y，且它们类型相同，则返回true 如：5===”5”返回false

三元运算符

组合比较符





数组：

数组是一个能在单个变量中存储多个值的特殊变量。

在PHP中，array()函数用于创建数组。

count()函数用于返回数组的长度

数值数组：

比如$car = array(“Volvo”,”BMW”,”Toyota”) 遍历用for循环

关联数组：

比如$age = array(“Peter”=>”35”,”Ben”=>”37”,”Joe”=>”43”)

sort() – 对数组进行升序排列

rsort() – 对数组进行降序排列

print\_r() 不知道是什么作用来的？

PHP超级全局变量，是PHP系统中自带的变量，在一个脚本的全部作用域中都可用。

PHP超级全局变量列表：

$GLOBALS

$\_SERVER

$\_REQUEST

$\_POST

$\_GET

$\_FILES

$\_ENV 这个超级全局变量是做什么的呢？

$\_COOKIE

$\_SESSION

PHP魔术变量：

PHP向它运行的任何脚本提供了大量的预定义常量

有8个魔术常量它们的值随着它们在代码中的位置改变而改变。

\_\_LINE\_\_ 文件中的当前行号

\_\_FILE\_\_ 文件的绝对路径

\_\_DIR\_\_ 文件所在的目录

\_\_FUNCTION\_\_ 返回该函数被定义时的名字(区分大小写)

...

PHP函数：



参数、返回

PHP中的符号->、=>和：：分别表示什么意思呢？

-> 一般是类方法的调用

=> 一般用在数组中，一个对应关系

：：类中 静态方法和静态属性的引用方法

例如

class Test{

public static $test = 1;

public static function test(){

…

}

}

可以不用实例化对象直接使用Test：：$test来取得$test属性的值

静态方法调用也同理Test：：test()；直接调用静态方法test

=== 判断两变量是否相等外，还会判断类型是否相等

MySQL部分

参考1：如果有时间，建议完成实验楼的实验，相信会有很大进步的；

https://www.shiyanlou.com/courses/9

参考2：菜鸟教程

http://www.runoob.com/mysql/mysql-tutorial.html

SQL = Structured Query Language 结构化查询语言

至今没有理解过来为什么？

CentOS：sudu yum install mysql mysql-server

添加root用户密码

mysqladmin -u root -p passsword ‘newpwd’

su 切换到root下面 (su = switch user)

mariadb 是什么数据库呢？

MySQL有三种主要数据类型：

1. 文本类 char、varchar、tinytext、text、blob、mediumtext、mediumblob、longtext longblob、enum(x,y,z,etc.) 、set
2. 数字类 tinyint、smallint、mediumint、int、bigint、float(size,d)、double(size,d)、decimal(size,d)
3. 日期类 date()、datetime()、timestamp()、time()、year()

MySQL目录结构：

* bin目录，存储可执行文件
* data目录，存储数据文件
* docs 文档
* include目录，存储包含的头文件
* lib目录，存储库文件
* share错误消息和字符集文件
* ...其他的配置信息

MySQL语句的规范

* 关键字与函数名称全部大写
* 数据库名称、表名称、字段名称全部小写
* SQL语句必须以分号结尾

主键：不能有重复的，不允许为空

外键：外键可以重复，可以是空值，用来和其他表建立联系的。

面试题：你的数据库用什么存储引擎？区别是？

答案：常见的有MyISAM和InnoDB。

MyISAM：不支持外键约束。不支持事务。对数据大批量导入时，它会边插入数据边建索引，所以为了提高执行效率，应该先禁用索引，在完全导入后再开启索引。

InnoDB：支持外键约束，支持事务。对索引都是单独处理的，无需引用索引。

可以参考网址：

http://www.cnblogs.com/smyhvae/p/4042303.html

|  |  |
| --- | --- |
| 功能 | 命令格式 |
| 添加或删除数据库 | |
| 查看当前有哪些数据库 | show databases; |
| 添加数据库 | create database database\_name; |
| 删除数据库 | drop database database\_name; |
| 使用某个数据库 | use database\_name |
| 添加或删除数据表 | |
| 创建表 | drop table if exists table\_name;  create table table\_name(  ...); |
| 在某个数据库中查看表 | show tables; |
| 查看某个表的结构 | describe table\_name; |
| 删除某个数据表 | drop table table\_name |
| 给数据表添加或删除某列 | |
| 增加列 alter table [table\_name] add [column\_name] [data\_type] [not null] [default] | |
| 删除列 alter table [table\_name] drop [column\_name] | |
| 修改列信息  alter table [table\_name] change [old\_column\_name] [new\_column\_name] [data\_type] | |
| 修改表名  alter table [table\_name] rename [new\_table\_name] | |
|  | |
| 查看表数据  select \* from table\_name  select col\_name1,col\_name2,... from table\_name | |
| 插入数据  insert into [table\_name] values(值1,值2,...)  insertt into [table\_name] (列1,列2...) values(值1,值2) | |
| where语法  select \* from table\_name where col\_name 运算符 值  between like  and or | |
| null的判断用is / is not，而不能用等号  select \* from table\_name where col\_name is null | |

|  |  |
| --- | --- |
| 功能 | 命令格式 |
| distinct的作用:  select distinct column\_name from table\_name | |
| order by  select \* from table\_name [where 子句] order by col\_name [asc/desc] | |
| 使用limit截取查询结果  select \* from table\_name [where子句] [order by子句] limit [offset,] rowCount | |
| insert into 和select的组合 (主要是数据迁移)  insert into table\_name select \* from other\_table\_name [where子句] | |
| Update 语法  update 表名 set 列名1 = XXX ，列名2=XXX... [where子句] | |
| Delete 语句  Delete from table\_name [where 子句] | |
| where中的in操作符  select \* from 表名 where 列名 in (value1,value2...)  select \* from 表名 where 列名1 in (select 列名 from 表名2) | |
| where中的between操作符  select \* from 表名 where 列名 between 值1 and 值2  select \* from 表名 where 列名 not between 值1 and 值2 | |
| where中的like操作符  select \* from 表名 where 列名 [not] like pattern  pattern:匹配模式 ‘abc’ ‘%abc%’ | |

内连接：只有符合条件的才显示

select \* from table\_name1 a inner join table\_name2 b on (a.studentId = b.studentId);

左连接：左边的表是全部数据，右边的只有符合条件的才有数据

select \* from table\_name1 a left join table\_name2 b on (a.studentId = b.studentId);

右连接：右边的表是全部数据，左边的只有符合条件的才有数据

select \* from table\_name1 a right join table\_name2 b on (a.studentId = b.studentId);

MySql不支持全连接FUll JOIN，不过可以通过联合UNION来模拟。

Linux知识点

目前市面上较知名的发行版有：

Ubuntu、RedHat、CentOS、Debian、Fedora、SuSE、OpenSUSE、Arch Linux、SolusOS

Linux系统启动过程，可以分为5个阶段

* 内核的引导
* 运行init
* 系统初始化
* 建立终端
* 用户登录系统

Linux关机(重启)命令：shutdown

语法：shutdow [选项][参数]

【选项】

|  |  |
| --- | --- |
| 选项 | 相应功能 |
| -a | 指定权限 |
| -r | 重启计算机(和reboot)命令一样 |
| -k | 模拟关机(只向用户发出警告信息，但不关机) |
| -h | 关闭计算机并关闭电源(常用) |
| -n | 不调用init进程关闭计算机(不推荐) |
| -c | 取消正在执行的关机命令 |
| -f | 重启计算机，但不进行磁盘检测 |
| -F | 重启计算机，进行磁盘检测 |
| -t(秒) | 指定发出警告信息与删除信息时要延迟的秒数 |
|  |  |

【附加参数】

|  |  |
| --- | --- |
| 参数 | 相应介绍 |
| 时间 | 指关闭计算机的时间。  可以为详细时间，如22:00，晚上10点关闭计算机  也可以作为详细分钟，如“+5”，则5分钟后关机 |
| 警告信息 | 可以是任意文本，信息，需要引号括起来才能使用 |

不管是重启系统还是关闭系统，首先要运行sync命令，把内存中的数据写到磁盘中。

关机：shutdown -h now \ halt \poweroff \init 0

重启系统：shutdown -r now \ reboot \init 6

Linux系统目录结构：

通过命令“ls / ”可以查看到用户的目录结构

/bin:bin是Binary的缩写，这个目录存放着最经常使用的命令

/boot:这是存放的是启动Linux时使用的一些核心文件，包括一些连接文件以及镜像文件

/dev:dev是Device的缩写，该目录下存放的是Linux的外部设备，在Linux中访问设备的方式和访问文件的方式是相同的。