



第 5 套 PHP 面试题

1、表单数据提交方式 POST 和 GET 的区别，URL 地址传递的数据最大长度是多少？

\$_GET 传参是请求 HTTP 协议通过 url 参数传递和接收,会在浏览器地址栏中看到

\$_GET 传参最多 2k 个字符

\$_POST 是实体数据传参,隐藏式的,可以通过表单提交大量信息

2、SESSION 和 COOKIE 的作用和区别，SESSION 信息的存储方式，如何进行遍历？

1) **SESSION**:存储在服务器端, **cookie**:存储在客户端

2) 两者都可通过时间来设置时间长短

3) **cookie** 不安全,考虑安全性还是用 **session**

4) **session** 保存到服务器端,如果访问量过大,对服务器性能很影响,应使用 **memcache** 缓存

session

5) 单个 **COOKIE** 在客户端限制是 3K,即存放的 **cookie** 不能超过 3K,**SESSION** 没有限制

6) 禁止了 **COOKIE** 后 **SESSION** 是不能使用的,**session** 因为存储在服务器端,具体路径在

php.ini 中 **session_dir** 目录中,是私有的每个用户产生 **session** 都不同

因为 **\$_SESSION** 是数组所以遍历就用数组的方式遍历!

3、通过 PHP 函数的方式对目录进行遍历，写出程序

```
<?php
```

```
dirs('F:\wwwroot');
```

云知梦，只为有梦想的人!



```
function dirs($path){  
  
    $path=str_replace('\\','/',$path);  
  
    $res=opendir($path);  
  
    readdir($res);  
  
    readdir($res);  
  
    while($filename=readdir($res)){  
  
        $newpath=$path.'/'.$filename;  
  
        if(is_dir($newpath)){  
  
            echo '<font color="red">目录:</font>'.$filename.'<br />';  
  
            dirs($newpath);  
  
        }else{  
  
            echo '文件: '.$filename.'<br />';  
  
        }  
  
    }  
  
}  
  
?>
```

4、Linux 的目录进行遍历，编写 shell 脚本

Tree 命令 //现在默认 Linux 都带有的命令

Ls -lR test

云知梦，只为有梦想的人!



5、什么是数据库索引，主键索引，唯一索引的区别，索引的缺点是什么？

数据库索引就是给数据库中的每张表建立索引,便于检索提高 select 效率

主键索引是给每张表的主键设置索引

唯一索引是给每张表的唯一值字段设置索引；

索引的缺点是建立索引需要增加存储空间，空间利用率低，并且当数据发生变化时改变索引资源消耗较大。

6、数据库设计时，常遇到的性能瓶颈有哪些，常有的解决方案？

1) 查询速度慢，避免全盘扫描,因为全盘扫描会导致大量磁盘 I/O 操作 用 sphinx 来解决

2) 读写过于频繁 读写分离来做

3) 设置主从,并做好冗余备份

4) 数据库压力过大 采用分布式数据库负载均衡来解决

5) memcache 缓存层

6) 链接数据库使用 PDO,或者 mysqli 预处理缓存字段

7) 索引优化

7、编写一个在线留言本，实现用户的在线留言功能，留言信息存储到数据库？

```
<form method="post" action="">
```

```
<input type="text" name="username" />
```

云知梦，只为有梦想的人！



```
<textarea name="main">  
  
</textarea>  
  
<input type="submit" name="sub"/>  
  
</form>
```

8、通过 JS 创建一个当前窗口的子窗口，通过程序实现子窗口对父窗口的操作实例？

父窗体 paren.html 中内容:

```
<html>  
  
  <title>父窗体</title>  
  
  <head>  
  
    <script>  
  
      function demo(){  
  
        window.open('children.html','name','height=200,width=200');  
  
      }  
  
    </script>  
  
  </head>  
  
  <body onload='demo()'>  
  
  </body>  
  
</html>
```

子窗体 children.html 中内容:

```
<html>
```

云知梦，只为有梦想的人！



```
<title> </title>
```

```
<head>
```

```
<script>
```

```
function change(){
```

```
    window.opener.document.body.bgColor='red'; //控制父窗体的
```

背景颜色

```
}
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<button onclick="change()">变色</button>
```

```
</body>
```

```
</html>
```

9、通过 JS 实现页面的前进和后退操作？

```
<a href="javascript:window.history.go(1)">前进</a>
```

```
<a href="javascript:window.history.go(-1)">后退</a>
```

10、浏览器 IE 和非 IE 浏览器的划分，区别是什么？

区别是浏览器内核的不同

CSS 和 js 兼容性的不同

云知梦，只为有梦想的人！



#navigator

11、设定网站的用户数量在千万级，但是活跃用户的数量只有 1%，如何通过优化数据库提高活跃用户的访问速度？

1)分表和分区

2)中间表(insert into table select * from stick where id>100

create view v_stick as select * from stick where id>100

show tables;

3)索引优化

12、服务器在运行的过程中，随着用户访问数量的增长，如何通过优化，保证性能？如果数据库已经达到最优化，请设计出继续升级的解决方案

squid->Ivs|集群->shpinx|memcache->php 静态化(缓存)->分区|主从|一主多从(读写分离)

首先可以通过

分库分表缓解一些负担，

应用缓存服务器，如 MemCache 服务器，

增加主从服务器和负载均衡服务器提高网站读取速度，

添加静态资源服务器，存放一些静态资源，如 CSS,Js,图片等，

提高检索速度可以架设内容检索服务器如 Shpinx，Xapian，

消息队列服务器，架设数据库集群，

云知梦，只为有梦想的人！



也可以考虑 NoSQL , 如谷歌的 BigTable,DB 连接池 , 使数据库读写分离

13、写出程序运行的结果

```
<?php
```

```
    $a=0;
```

```
    $b=0;
```

```
    If($a=3 || $b=3){
```

```
        $a++;
```

```
        $b++;
```

```
    }
```

```
Echo $a." " . $b;
```

```
?>
```

```
<?php
```

```
    $a=0;
```

```
    $b=0;
```

```
    If($a=4 | $b=3){
```

```
        $a++;
```

```
        $b++;
```

```
    }
```

```
Echo $a." " . $b;
```

云知梦，只为有梦想的人！



?>

100 011

结果：

1.1

8.4

14、实现无限级分类的数据库设计

第一步:

```
create table lt_cat(  
    id int(11) not null auto_increment primary key,  
    pid int(11) not null,  
    path varchar(255) not null,  
    name varchar(50) not null  
)engine=myisam default charset=utf8;
```

第二步: //path 中保存父级的 id

```
insert into lt_cat(pid,name,path) values(0,'China','0'); //id 为 1
```

```
insert into lt_cat(pid,name,path) values(0,'American','0'); //id 为 2
```

```
insert into lt_cat(pid,name,path) values(2,'new York','0,2'); //id 为 3
```

```
insert into lt_cat(pid,name,path) values(1,'beijing','0,1'); //id 为 4
```

```
insert into lt_cat(pid,name,path) values(4,'海淀','0,1,4'); //id 为 5
```

云知梦，只为有梦想的人！



```
insert into lt_cat(pid,name,path) values(5,'上地','0,1,4,5');      //id 为 6
```

第三步:

```
Select name,concat(path,' / ',id) as bpath from lt_cat  order by bpath;
```

Id name pid path

1 a 0 0

2 b 1 0-1

3 c 1 0-1

4 d 3 0-1-3

```
Select * from table order by concat(path," -" ,id);
```