

HTML + CSS 部分

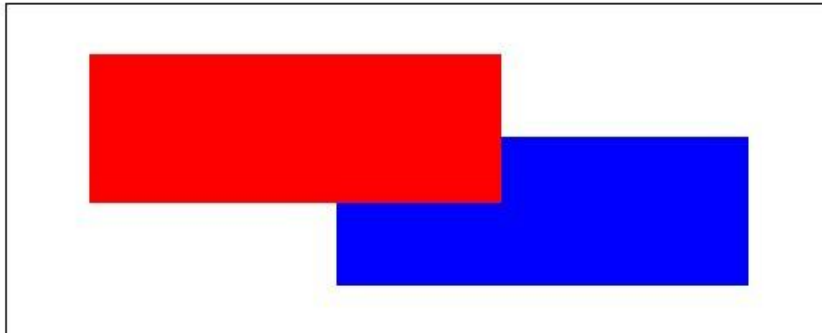
1. 请说明 HTML 文档中 DTD 的意义和作用（酷讯）

DTD，文档类型定义，是一种保证 html 文档格式正确的有效方法，在解析网页时，浏览器将使用 DTD 来检查页面的有效性（是否符合规范，元素和标签使用是否正确）并且采取相应的措施。同时它还会影响浏览器的渲染模式（工作模式）。

2. 准确的指出以下代码的显示结果（酷讯）

```
<style> a{ position:relative;
  height:200px; width:500px;
  border:solid 1px #000;
  background:#FFF;
  }
  #b,#c{ position:absolute; width:250px; height:90px;}
  #b{ top:30px; left:50px; background:#FF0000; z-index:1;}
  #c{ bottom:30px; right:50px; background:#0000FF;}
</style>
<div class="a">
<div id="b"></div>
<div id="c"></div>
</div>
```

结果如图所示：



3. 指出以下代码的显示结果（酷讯）

```
<table border=1 width=500 style="text-align:center;">
<tr>
  <td rowspan=2 width=50% height=50>a</td>
  <td width=50% height=25>d</td>
</tr>
<tr><td width=50% height=25>b</td></tr>
<tr height=25><td colspan=2>c</td></tr>
</table>
```

显示效果如图：

a	d
	b
c	

4. 如何产生带有数字列表符号的列表？（ ）

- A.
- B. <dl>
- C. D. <list>

答案： C

5. 如何显示没有下划线的超链接？（ ）

- A. a {text-decoration:none}
- B. a {text-decoration:no underline}
- C. a {underline:none}
- D. a {decoration:no underline}

答案： A

6. 以下一段 XHTML 代码片段，它不符合 W3C 规范，请问出现了几处错误？

```
<div id="newsmore">
  <ul>
    <li></li>
    <li>印度总理辛格访华专题</li>
    <li>
      <ul id="newsmore">
        <li>专访辛格： 早日解决边界问题是战略目标</li>
        <li>辛格称北京奥运将为印度举办赛事提供灵感</li>
        <li>印度总理辛格访华参观北京奥运场馆建设</li>
      </ul>
    </li>
  </ul>
</div>
```

有两处， div 和 ul 有一个相同的 id 属性 newsmore， img 标签没有关闭。

7. 请描述出两点以上 XHTML 和 HTML 最显著的区别

- (1).XHTML 必须强制指定文档类型 DocType, HTML 不需要
- (2).XHTML 所有标签必须闭合, HTML 比较随意
- (3).XHTML 严格区分大小写, 所有标签的元素和属性的名字都必须使用小写
- (4).XHTML 要求所有的标记都必须要有个相应的结束标记
- (5).XHTML 规定所有属性都必须有一个值, 没有值的就重复本身

8. 请问, div 和 span, p 标签三者间的区别 (YG)

从语义化的角度来讲, div 和 span 标签对于它们包含的元素是没有意义的, 而 p 标签则是一个语义化标签, 表示一个段落。所以 div 和 span 一般是用来布局。

从显示的角度来讲, span 是内联的, 用在一小块的内联 HTML 中, 前后不断行, div 元素是块级的, 等同于其前后有断行, p 元素是块级的, 前后断行, 而且还要再隔一行, 相当于断两行。

9. CSS 中 class 和 id 的区别(YG)

id: 每个页 ID 是唯一的不能重复。 class: 可以设置多个 class 为同一个值。

10. 请列举几个 meta 元素(YG)

```
<meta name="keywords" content=""> //向搜索引擎说明你的网页的关键词
<meta name="description" content=""> //告诉搜索引擎你的站点的主要内容
<meta name="author" content="你的姓名"> //告诉搜索引擎你的站点的制作的作者
<meta http-equiv="Content-Type" content="text/html";charset=utf-8"> //指定字符集
<meta http-equiv="refresh" content="n;url="> //定时让网页在指定的时间 n 内跳转
<meta http-equiv=' refresh' content=" n;url" >
```

<meta http-equiv="expires" content="Mon,12 May 2001 00:20:00 GMT"> // 可以用于设定网页的到期时间, 一旦过期则必须到服务器上重新调用。需要注意的是必须使用 GMT 时间格式

```
<meta http-equiv="pragma" content="no-cache"> //禁用缓存
```

<meta http-equiv="set-cookie" content="Mon,12 May 2001 00:20:00 GMT"> //cookie 设定, 如果网页过期, 存盘的 cookie 将被删除。需要注意的也是必须使用 GMT 时间格式

11. 怪异模式和 XHTML 模式下 IE 的宽的区别(YG)

所谓的标准模式是指, 浏览器按 W3C 标准解析执行代码; 怪异模式则是使用浏览器自己的方式解析执行代码, 因为不同浏览器解析执行的方式不一样, 所以我们称之为怪异模式。

浏览器解析时到底使用标准模式还是怪异模式, 与你网页中的 DTD 声明直接相关, DTD 声明定义了标准文档的类型 (标准模式解析) 文档类型, 会使浏览器使用相应的方式加载网页并显示, 忽略 DTD 声明, 将使网页进入怪异模式(quirks mode)。

在 XHTML 模式下, 声明一个元素的宽度, 仅仅指的是在 css 中设置的 width, 而在怪

异模式下，元素的宽度包括 width 和 padding。

12. 实现框架的标签，使用举例(YG)

```
<frameset cols="25%,50%,25%">
  <frame src="frame_a.htm" />
  <frame src="frame_b.htm" />
  <frame src="frame_c.htm" />
</frameset>
```

13. 请写出以下 html 标签的含义：input form script style table b img (小米)

`<input />` 标签用于搜集用户信息。根据不同的 type 属性值，输入字段拥有很多种形式。输入字段可以是文本字段、复选框、掩码后的文本控件、单选按钮、按钮等等。

`<form>` 标签用于为用户输入创建 HTML 表单。表单能够包含 input 元素，比如文本字段、复选框、单选框、提交按钮等等。

`<script>` 标签用于定义客户端脚本，比如 JavaScript。script 元素既可以包含脚本语句，也可以通过 src 属性指向外部脚本文件。

`<style>` 标签用于为 HTML 文档定义样式信息。

`<table>` 标签定义 HTML 表格。简单的 HTML 表格由 table 元素以及一个或多个 tr、th 或 td 元素组成。

`` 呈现粗体文本效果。

`` 元素向网页中嵌入一幅图像。

14. 用 css、html 编写一个两列布局的网页，右侧固定宽度 200px，左侧自适应 (小米) 方案 1:

HTML 结构:

```
<div id="wrap">
  <div id="content" style="height:340px;">自适应区，在前面</div>
  <div id="sidebar" style="height:240px;">固定宽度区</div>
</div>
<div id="footer">后面的一个DIV，以确保前面的定位不会导致后面的变形</div>
```

CSS 样式:

```
<style>
  #content,#sidebar,#footer{border: 1px solid #ccc;}
  #wrap{display: table;width: 100%;}
  #content{display: table-cell;}
  #sidebar{display: table-cell; width: 200px;}
</style>
```

这种方案最简单，但不支持 IE7 及 IE7 以下的浏览器。

方案 2:

HTML 结构

```
<div id="wrap">
  <div id="content" style="height:140px;">
    <div id="contentb">
      content 自适应区, 在前面
    </div>
  </div>
  <div id="sidebar" style="height:240px;">sidebar 固定宽度区</div>
</div>
<div id="footer">后面的一个DIV, 以确保前面的定位不会导致后面的变形</div>
```

CSS 样式

```
#wrap {*zoom: 1; position: relative;}
#sidebar {width: 200px; float: right; border: 1px solid #ccc;}
#content {margin-left: -210px; float: left; width: 100%;}
#contentb {margin-left: 210px; border: 1px solid #ccc;}
#footer {border: 1px solid #ccc; margin-top: 10px; clear: both;}
```

这种方案, 兼容性好, 但是会产生额外的标签。

JavaScript 部分

1. JS 表单弹出对话框函数是?获得输入焦点函数是?

弹出对话框函数: alert(), prompt(), confirm() 获得输入焦点函数: focus()

2. JS 的转向函数是?怎么引入一个外部 JS 文件?

转向使用 window.location.href = "" 引入外

部 js 使用 <script src=""/>

3. 解释下面语句的意思: document.form["formName"].submit。(百度)

获取 name 为 formName 的表单并将该表单数据提交到服务器。但这行代码是不能运行的, 有两处错误, 一是通过表单名称来获取表单得到时候, 应该写成 document.forms['formName'], 提交表单事件使用 submit 方法, 需要加上括号, 下面给出简单的示例代码:

```

7 <body>
8   <form action="a.php" method="post" name="login" id='f1'>
9     用户名: <input type="text" name="username" /> <br />
10    <button type="button" id="btn">提交</button>
11  </form>
12  <script type="text/javascript">
13    document.getElementById("btn").onclick = function(){
14      if(document.forms['login'].username.value != ''){
15        document.forms['login'].submit();
16      }
17    }
18  </script>
19 </body>

```

4. 有下面语句: `<input id="txt" type="text" value="baidu" />` 编写代码, 当鼠标划过文本框, 自动选中文本框中的内容。(百度)

```

7 <body>
8   <input id="txt" type="text" value="baidu" />
9   <script type="text/javascript">
10    var textBox = document.getElementById('txt');
11    textBox.onmouseover = function(){
12      this.select();
13    }
14  </script>
15 </body>

```

5. 设计一个网页, 使得打开它时弹出一个全屏的窗口, 该窗口中有一个文本框和一个按钮。用户在文本框中输入信息后点击按钮就可以把窗口关闭, 而输入的信息却在主网页中显示。

(新浪)

主窗口:

```

<a href="" id = 'a1' target = 'new'>新窗口</a>
<div id="msg"></div>
<script type="text/javascript">
  var a1 = document.getElementById('a1');
  a1.onclick = function(){
    window.open('new.html', 'new', 'location=no, toolbar=no');
    return false;
  }
</script>

```

弹出窗口:

```
<input type="text" name = 'message' id = 'm1' /> <br />
<input type="button" value = '关闭' id = 'btn' />
<script type = "text/javascript">
    var btn = document.getElementById('btn');
    var message = document.getElementById('m1');
    btn.onclick = function(){
        var div = window.opener.document.getElementById('msg');
        div.innerHTML = message.value;
        window.close();
    }
</script>
```

6. 可以使用哪些方法使用 JavaScript 向服务器发出请求且不开当前页面，简单对比各自的特点（如果存在）（酷讯）

使用 XMLHttpRequest 对象异步请求

7. 判断以下代码是否正确，如果有错，请指出错误，如果正确，请指出运行结果（酷讯）

```
var arr = new Array(
    new Array(1,2,3,4),
    new Array("abc", "def", "xyz"),
);
for(i = 0; i < arr.length; i++) { document.write(arr[0])
}
```

怀疑这个代码写错了，第二个 new array 后面的逗号是多余的，如果此处没问题，则代码是正确的，输出的结果是 1,2,3,4 1,2,3,4

注：arr 是一个二维数组，该数组有两个元素，第一个元素是数组[1,2,3,4]，第二个元素也是一个数组为["abc", "def", "xyz"]，for 循环语句执行了两次，但都是输出第一个元素，即数组[1,2,3,4]。

8. 用 JavaScript 写一个函数,其功能是可以删除数组中重复的元素。

```
<script>
function array_unique(arr){
    var result = arr;
    for(var i = 0; i < arr.length; i++){
        for(var j = 0; j < arr.length; j++){
            temp = arr[i];
            //如果当前元素与后面某一个元素相等，则移除该元素
            if((i + j + 1) < arr.length && temp === arr[i + j + 1]){
                result.splice(i + j + 1,1);
            }
        }
    }
    return result;
}
var a = [4,7,8,5,8,6,43,7,0,false,'',{)];
var b = array_unique(a);
alert(b); //4,7,8,5,6,43,0,false,,[object Object]
</script>
```

9. 以下哪条 JavaScript 语句会产生运行错误：（ ）

- A. var obj = ();
- B. var obj = [];
- C. var obj = { };
- D. var obj = //;

答案：A

10. 请选择结果为真的表达式：（ ）

- A. null instanceof Object
- B. null === undefined
- C. null == undefined
- D. NaN == NaN

答案：C

11. foo 对象有 att 属性，那么获取 att 属性的值，以下哪些做法是可以的：（ ）

- A. foo.att
- B. foo("att")
- C. foo["att"]
- D. foo{"att"}
- E. foo["a"+"t"+"t"]

答案: ACE

12. 如何添加 html 元素的事件,有几种方法, 举例

- (1).直接作为元素的属性, 如
- (2).使用 DOM 0 级事件, 简单, 兼容性好, 如 img.onclick = function(){}
- (3).使用 DOM 2 级事件, 功能更强大, 在非 IE 等标准浏览其中, 使用 addEventListener, 在 IE 浏览器中则使用 attachEvent 来实现。

13. JavaScript 能否定义二维数组, 如果不能你如何解决?

JavaScript 不支持二维数组定义, 可以用 arr[0] = new array()来解决。

14. 假设 a.html 和 b.html 在同一个文件夹下面, 用 JavaScript 实现当打开 a.html 五秒钟后, 自动跳转到 b.html。

```
<script type="text/javascript">
  function go2b(){
    window.location.href = "http://www.baidu.com";
    window.close();
  }
  setTimeout( "go2b()",5000 ); //5秒钟后自动执行go2b()
</script>
```

15. 请使用 JavaScript 写出三种产生一个 image 标签的方法 (提示: 从方法、对象、HTML 角度考虑)

- (1). var img = new Image();
- (2). var img = document.createElement("image")
- (3). img.innerHTML = ""

16. js 中网页前进和后退的代码

前进: history.forward();或者 history.go(1); 后

退: history.back (); 或者 history.go(-1);

17. 请写出一条至少 3 个节点的 DOM 树 (YG)

```
<script type="text/javascript">
  var div = document.createElement("div");
  var a = document.createElement("a");
  a.href="http://www.baidu.com";
  var span = document.createElement("span");
  span.innerHTML= "百度";
  a.appendChild(span);
  div.appendChild(a);
  document.body.appendChild(div);
</script>
```

结果 HTML:

```
<div><a href="http://www.baidu.com"><span>百度</span></a></div>
```

18. 实现点击按钮弹出窗口的代码 (YG)

```
document.getElementById('button').onclick = function(){ window.open
('page.html');
}
```

其中, button 是按钮的 id, page.html 是要弹出的窗口页面。

19. JavaScript 包括那些基本数据类型? (小米)

JavaScript 中包括 5 种基本数据类型, 分别是 Number, String, Boolean, Null 和 Undefined。

PHP 语言基础

1、 php7.0 新增的一些特性？

- a)、标量类型声明
- b)、?? --NULL 合并运算符
- c)、匿名类
- d)、更多 Error 错误可以进行异常处理
- e)、结合比较运算符 (<=>)
- f)、定义数组常量
- g)、标量类型声明 有两种模式: 强制 (默认) 和 严格模式。 现在可以使用下列类型参数 (无论用强制模式还是严格模式): 字符串(string), 整数 (int), 浮点数 (float), 以及布尔值 (bool)。它们扩充了 PHP5 中引入的其他类型: 类名, 接口, 数组和 回调类型。
- h)、返回值类型声明

2. strlen()与 mb_strlen()的作用分别是什么(新浪网技术部)

strlen 和 mb_strlen 都是用于获取字符串长度的, 其中 strlen 只针对单字节编码字符, 也就是说它计算的是字符串的总字节数, 如果是多字节编码, 如 gbk 和 utf8, 使用 strlen 得到的不是字符的个数, 而是该字符的总字节数, 可以使用 mb_strlen 获取其字符个数, 使用 mb_strlen 要注意两点, 一是要开启 mbstring 扩展, 二是要指定字符集。

示例:

```
1 <?php
2 $str1="php";
3 echo strlen($str1);
4
5 $str2 = "千锋教育PHP学院";
6 echo strlen($str2);//当前页面是utf-8字符集, 如果是gbk, 则结果为15个字符
7 echo mb_strlen($str2);//注意, 未指定字符集, 使用内部字符编码(单字节)
8 echo mb_strlen($str2,'utf-8');
```

相关题目 1: 实现中文字串截取无乱码的方法。

方法一, 使用 php 内置函数 mb_substr()

方法二, 自定义函数, 以 utf8 为例, 如下:

```

1 <?php
2 header('Content-Type:text/html;charset=utf8');
3 /*
4 utf8编码字符串截取无乱码
5 $str string 要处理的字符串
6 $start int 从哪个位置开始截取
7 $length int 要截取字符串的个数 默认从开始位置到末尾
8 string 截取字符串
9 */
10 function substr_utf8($str,$start,$length = null){
11     return join("",array_slice(
12         preg_split("//u",$str,-1,PREG_SPLIT_NO_EMPTY),$start,$length))
13 }
14 //实例
15 $str = '千锋教育PHP学院';
16 echo substr_utf8($str,2,4);//输出

```

相关题目 2: 如何求解字符串“中国 2 北 333 京”的字符数（一个中文一个字符），并找到第四个字符“北”

方法一，使用 php 内置函数，确保配置中已打开 mbstring 扩展

```

1 <?php
2 header('Content-Type:text/html; charset = utf8');
3 $a = '中国2北333京';
4 echo mb_strlen($a,'utf8');
5 echo mb_substr($a,3,1,'utf8');

```

方法二，自定义函数实现求其长度，截取使用上面定义的 substr_utf8

```

1 <?php
2 header('Content-Type:text/html;charset=utf8');
3 /**
4 * 计算utf编码字符串的长度
5 * @param $str string 要计算的字符串
6 * @return int 字符串的长度
7 */
8 function strlen_utf8($str){
9     return count(preg_split('//u', $str,-1,PREG_SPLIT_NO_EMPTY));
10 }
11 $a = '中国2北333京';
12 echo strlen_utf8($a);

```

3. 下列哪个函数是用正则表达式将字符串分割到数组中()

- A. split
- B. implode

C. explode D. join

答案: A

4、 写出下列程序的输出结果(新浪网技术部)

```
1 <?php
2 $x = 87;
3 $y = ($x % 7) * 16;
4 $z = $x > $y ? 1 : 0;
5 echo $z;
6 ?>
```

答案: 1

5、 写出下列几个预定义全局变量的作用(新浪网技术部)

`$_SERVER['DOCUMENT_ROOT']` //当前运行脚本所在的文档根目录

`$_SERVER['HTTP_HOST']` //当前请求的 Host: 头部的内容

`$_SERVER['REMOTE_ADDR']` //正在浏览当前页面用户的 IP 地址

`$_SERVER['HTTP_REFERER']` //链接到当前页面的前一页面的 URL 地址

`$_SERVER['SERVER_NAME']` //当前运行脚本所在服务器主机的名称

`$_FILES` //包含有所有上传的文件信息

`S_FILES['userfile']['name']` //客户端机器文件的原名称

`$_FILES['userfile']['type']` //文件 MIME 类型, 如果浏览器提供此信息的话, 如“image/gif”。

`$_FILES['userfile']['size']` //已上传文件的大小, 单位为字节

`$_FILES['userfile']['tmp_name']` //文件被上传后在服务端储存的临时文件名

`$_FILES['userfile']['error']` //和该文件上传相关的错误代码

5. include 和 require 都能把另外一个文件包含到当前文件中, 他们有什么区别? Include 和 include_once 又有什么区别? (新浪网技术部)

二者区别只有一个, 那就是对包含文件的需求程度, include 就是包含, 如果被包含的文件不存在的话, 那么则会提示一个错误, 但是程序会继续执行下去。

而 require 意思是需要, 如果被包含文件不存在或者无法打开的时候, 则会提示错误, 并且会终止程序的执行。

这两种结构除了如何处理失败之外完全一样。once 的意思是一次, 那么 include_once 和 require_once 表示只包含一次, 避免重复包含。

相关题目 1: What is the difference between include & include_once? include & require?

(Yahoo)

相关题目 2: 语句 include 和 require 都能把另外一个文件包含到当前文件中, 它们的区别是___; 为了避免多次包含同一文件, 可以用语句___来代替它们。

在如何处理失败时, include() 产生一个警告而 require() 则导致一个致命错误;require_once()/include_once()

相关题目 3: What functions can you use to add library code to the currently running script?

(Yahoo)

include、require

6. 用最少的代码写一个求 3 值最大值的函数. (51.com 笔试题)

```
1 <?php
2 //定义函数
3 function maximum($a,$b,$c){
4     return $a > $b ? ($a > $c ? $a : $c) : ($b > $c ? $b : $c);
5 }
6 //调用实例
7 echo maximum(24,15,8);
8 ?>
```

7、get 与 post 提交方法的区别? 除此之外你还知道其他的 HTTP 请求吗?

1. get 是把参数数据队列加到提交表单的 ACTION 属性所指的 URL 中, 值和表单内各个字段一一对应, 在 URL 中可以看到。post 是通过 HTTP post 机制, 将表单内各个字段与其内容放置在 HTML HEADER 内一起传送到 ACTION 属性所指的 URL 地址。用户看不到这个过程。
2. 对于 get 方式, 服务器端用 Request.QueryString 获取变量的值, 对于 post 方式, 服务器端用 Request.Form 获取提交的数据。
3. get 传送的数据量较小。post 传送的数据量较大, 一般被默认为不受限制。但理论上, IIS4 中最大量为 80KB, IIS5 中为 100KB。get 安全性非常低, post 安全性较高。

其他的 http 请求方法: head、put、delete、options、copy、patch、link.

8、有三个 php 文件位于同一目录下, 内容如下所示, 使用浏览器访问 c.php, 请问是否存在问题。如果存在问题, 请指出修正方法并写出浏览器查看效果, 如果不存在问题, 请写出浏览器查看效果 (酷讯 PHP 工程师笔试题)

A.php:-----

```
<?php function fa() { echo "in Function A\n"; }?>
```

B.php:-----

```
<?php include 'a.php'; ?>
<?php function fb() { fa(); echo "in Function B\n"; } ?>
```

C.php:-----

```
<?php include 'a.php'; ?>
<?php include 'b.php'; ?>
<?php fa(); fb(); ?>
```

答案: 存在问题, a.php 被包含了两次, 导致 fa()函数重复定义, 使用 include_once 避免重复包含。

9. echo(),print(),print_r()的区别? (新浪)

echo 是一个语言结构, 输出一个或多个字符串;

print() 实际上不是一个函数 (它是一个语言结构), 因此你可以不必使用圆括号来括起它的参数列表, 它输出一个字符串;

print_r () 是一个函数, 打印变量的信息, 基本类型, 数组, 对象。

```
echo $a,$b;
print('<br />');
echo $a.$b;
```

二者之间的一个区别

注意, 是 echo 本身支持的一种语法, 而. 则是字符串连接操作符, 使用, 的效率要高一些。

相关题目: What is the difference between "print()" and "echo()"? (腾讯)

answer: print is a function,echo is a language construct

10. 用 PHP 打印出前一天的时间格式是 2006-5-10 22:21:21。

```
1 <?php
2 echo date("Y-m-d H:i:s",time() - 3600*24);
3 //使用strtotime, 可以将任何字符串的时间表示转换成时间戳, 仅针对英文
4 echo '<br />';
5 echo date('Y-m-d H:i:s',strtotime("-1 day"));
6 //在实际开发的时候, strtotime非常有用, 因为对于表单来说, 提交的数据都是字符串
7 //比如, 2013-4-27, 需要将其转成时间戳然后存到数据库
```

相关题目: 求两个日期的差数, 例如 2009-3-1 ~ 2009-4-4 的日期差数
(strtotime("2009-4-4")-strtotime("2009-3-1"))/3600*24

11. 不使用第三个变量交换两个变量的值方案一:

```

77 echo $a, $b;
$a = 'PHP';
$b = 'MySQL';
echo $a,$b;
list($a,$b) = array($b,$a);
echo '<br />';
echo $a,$b;

```

方案二:

对字符进行编码，比如 `base64_encode`，要保证编码是可逆的

```

27 $a = 'PHP编程';
28 $b = 'MySQL数据库';
29 echo $a,$b;
30 // list($a,$b) = array($b,$a);
31 $a = base64_encode($a);
32 $b = base64_encode($b);
33 $a = $a . '&' . $b; //得到一个有 & 符号连接的字符串
34 //使用字符串分割函数得到数组
35 $b = explode('&', $a);
36 $a = base64_decode($b[1]); //得到第一个字符并解码
37 $b = base64_decode($b[0]); //得到第二个字符并解码
38
39 echo '<br />';
40 echo $a,$b;

```

12. 请说明 php 中传值与传引用的区别。什么时候传值什么时候传引用？

变量默认总是传值赋值。那也就是说，当将一个表达式的值赋予一个变量时，整个原始表达式的值被赋值到目标变量。这意味着，例如，当一个变量的值赋予另外一个变量时，改变其中一个变量的值，将不会影响到另外一个变量。

PHP 也提供了另外一种方式给变量赋值：引用赋值。这意味着新的变量简单的引用（换言之，“成为其别名”或者“指向”）了原始变量。改动新的变量将影响到原始变量，反之亦然。使用引用赋值，简单地将一个 `&` 符号加到将要赋值的变量前（源变量）。

对象默认是传引用。

对于较大的数据，传引用比较好，这样可以节省内存的开销。

相关题目 1: What would the following code print to the browser? Why?


```
1 <?php
2 $num = 10;
3 function multiply($num){
4     $num = $num * 10;
5 }
6 multiply($num);
7 echo $num;
```

输出 10

相关题目 2: **What is the difference between a reference and a regular variable? How do you pass by reference & why would you want to? (Yahoo)**

reference 传送的是变量的地址而非它的值，所以在函数中改变一个变量的值时，整个应用都见到这个变量的新值。

一个 regular variable 传送给函数的是它的值，当函数改变这个变量的值时，只有这个函数才见到新值，应用的其他部分仍然见到旧值。

13. 将 1234567890 转换成 1,234,567,890 每 3 位用逗号隔开的形式。
(百度)

```
1 <?php
2 $str = '1234567890';
3 function str($str){
4     //反转字符串得到0987654321
5     $str = strrev($str);
6     //使用，分割字符串，得到098,765,432,1,
7     $str = chunk_split($str,3, ',');
8     //再次反转字符串，得到,1,234,567,890
9     $str = strrev($str);
10    //去掉左边的，得到1,234,567,890
11    $str = ltrim($str, ',');
12    return $str;
13 }
```

相关题目 1: **如何实现字符串翻转?**

strrev(), 不过这种方法都不能解决中文字符串翻转的问题，会出错的。

```

1  <?php
2  //反转字符串 使用正则和数组实现
3  function strrev_utf8($str){
4      return join("",array_reverse(
5          preg_split("//u",$str)
6          ));
7  }
8  $str = "千锋教育PHP学院";
9  echo strrev_utf8($str); //院学PHP育教锋千 ?>
10

```

相关题目 2: 假设现在有一个字符串 **www.baidu.com** 如何使用 **PHP** 对它进行操作使字符串以 **moc.udiab**.输出? (亿邮)

```

$str3 = 'www.baidu.com';
echo strrev(str_replace('www','',$str3));

```

14. 用 **PHP** 写出显示客户端 **IP** 与服务器 **IP** 的代码。客户端 **IP**:

`$_SERVER["REMOTE_ADDR"]` 服务器端 **IP**: `$_SERVER["SERVER_ADDR"]`

15. 简述如何得到当前执行脚本路径, 包括所得到参数。

获取当前执行脚本路径使用 `$_SERVER["SCRIPT_FILENAME"]` 或 `__FILE__` 获取参数, 使用 `$_SERVER["QUERY_STRING"]`

16. **What is the difference between foo() & @foo()?** (Yahoo)

`foo()` 会执行这个函式, 任何解译错误、语法错误、执行错误都会在页面上显示出来。 `@foo()` 在执行这个函式时, 会隐藏所有上述的错误讯息。

很多应用程序都使用 `@mysqli_connect()` 和 `@mysqli_query` 来隐藏 `mysql` 的错误信息, 这是不对的, 因为错误不该被隐藏, 你必须妥善处理它们, 可能的话解决它们。

17. 下面哪个选项没有将 **john** 添加到 **users** 数组中? (百度)

- A. `$users[] = "john";`
- B. `array_add($users, "john");`
- C. `array_push($users, "john");`; D. `$users ||= "john";`

答案: BD

18. 检测一个变量是否有设置的函数是否?是否为空的函数是?

`isset` 检测一个变量是否设置 `empty` 检测是否为空注意二者的区别，如果 变量 是非空或非零的值，则 `empty()` 返回 `FALSE`。换句话说，`""`、`0`、`"0"`、`NULL`、`FALSE`、`array()` 以及没有任何属性的对象都将被认为是空的。

19. 在 PHP 中，当前脚本的名称（不包括路径和查询字符串）记录在预定义变量_____中；而链接到当前页面的前一页面 URL 记录在预定义变量_____中。

当前脚本名称：`$_SERVER["PHP_SELF"]` 或者 `$_SERVER["SCRIPT_NAME"]`；链接到当前页面的前一页面的 URL 地址：`$_SERVER["HTTP_REFERER"]`。

20. `sort()`、`assort()`、和 `ksort()` 有什么分别？它们分别在什么情况下使用？

`sort()`，根据数组中元素的值，以英文字母顺序排序，索引键会由 0 到 n-1 重新编号。主要是当数组索引键的值无关紧要时用来把数组排序。

`assort()`，PHP 没有 `assort()` 函式，所以可能是 `asort()` 的笔误。

`asort()`，对数组进行排序，数组的索引保持和单元的关联。主要用于对那些单元顺序很重要的结合数组进行排序。

`ksort()`，根据数组中索引键的值，以英文字母顺序排序，特别适合用于希望把索引键排序的关联数组。

21. 在 PHP 中 `error_reporting` 这个函数有什么作用？打开或者关闭错误报告，如：

```
error_reporting(0);
error_reporting(E_ALL & ~ E_NOTICE);
error_reporting(E_ALL);
```

相关题目：`error_reporting(2047)` 什么作用？（新浪）

$2047 = 1 + 2 + 4 + 8 + 16 + 32 + 64 + 128 + 256 + 512 + 1024$ 其中，1 对应 `E_ERROR`，2 对应 `E_WARNING`，4 对应 `E_PARSE`，8 对应 `E_NOTICE`，16 对应 `E_CORE_ERROR`，32 对应 `E_CORE_WARNING`，64 对应 `E_COMPILE_ERROR`，128 对应 `E_COMPILE_WARNING`，256 对应 `E_USER_ERROR`，512 对应 `E_USER_WARNING`，1024 对应 `E_USER_NOTICE`。`error_reporting(2047)`意味着上述错误都会显示出来。

22. 写出以下程序的输出结果（CBSI）

```
1 <?php
2 $str = 'cd';
3 $$str = 'hotdog';
4 $$str .= 'ok';
5 echo $cd;
```

hotdogok

相关题目：什么是可变变量？

获取一个普通变量的值作为这个可变变量的变量名。

23. 常量如何定义？如何检测一个常量是否被定义？

定义常量：define()，检测常量是否定义：defined()，如

```
define("TEST","happy new year!");
if( defined("TEST")){ echo TEST;}
```

24. 执行程序段<?php echo 8%(-2) ?>将输出

%为取模运算,输出 0

\$a % \$b 其结果的正负取决于 \$a 的符号。

```
echo ((-8)%3)."<br />"; //输出 -2 echo
(8%(-3))."<br />"; //输出 2
```

25. 数组函数 arsort 的作用是____; 语句 error_reporting(2047)的作用是____。

arsort: 对数组进行逆向排序并保持索引关系 error_reporting(2047)的作用 report All errors and warnings, 见 21 题中相关题目。

26. 一个函数的参数不能是对变量的引用，除非在 php.ini 中把____设为 on.

allow_call_time_pass_reference : 是否启用在函数调用时强制参数被按照引用传递，默认关闭。

27. 在 PHP 中，heredoc 是一种特殊的字符串，它的结束标志必须____ 顶格写，并且不能包含任何其它字符除";"，如：

```
$str = <<<SHOW my name is
    Jiang Qihui!
SHOW;
```

28. echo count("abc"); 输出什么？(新浪?)

输出 1

`count` : 计算数组中的单元数目或对象中的属性个数, 通常是一个 `array`, 任何其它类型都只有一个单元。

对于对象, 如果安装了 `SPL`, 可以通过实现 `Countable` 接口来调用 `count()`。该接口只有一个方法 `count()`, 此方法返回 `count()` 函数的返回值。

如果 `var` 不是数组类型或者实现了 `Countable` 接口的对象, 将返回 `1`, 有一个例外, 如果 `var` 是 `NULL` 则结果是 `0`。

相关题目: **What will be the output of the following PHP code:** (腾讯)

```
<?php
    echo count(strlen("http://php.net"));
?> answer:
1
```

30. 写个函数用来对二维数组排序 (新浪)

```
1 <?php
2 /**
3  * 根据某列对二维数组进行排序
4  * @param $arr array 要排序的数组
5  * @param $row string 排序依据列
6  * @param $type string asc表示正序,默认值, desc表示逆序
7  * @return array 返回排序好的数组
8  */
9 function array_sort($arr,$row,$type = 'asc'){
10     $arr_temp = array();
11     //将排序依据列作为数组的键名
12     foreach ($arr as $v) {
13         $arr_temp[$v[$row]] = $v;
14     }
15     //然后按照键名对数组排序, 并保持索引关系
16     if ($type == 'asc') {
17         ksort($arr_temp);
18     } elseif ($type == 'desc') {
19         krsort($arr_temp);
20     } else {
21         # code...
22     }
23     //返回排序结果
24     return $arr_temp;
25 }
26 $person = array(
27     array('id' => 2, 'name' => 'zhangsan', 'age' => 23),
28     array('id' => 5, 'name' => 'lisi', 'age' => 28),
29     array('id' => 3, 'name' => 'apple', 'age' => 17)
30 );
31 //调用实例
32 $person = array_sort($person,'name');
33 print_r($person);
```

31. 写 5 个不同的自己的函数, 来获取一个全路径的文件的扩展名, 允许封装 php 库中已有的函数。(新浪)

```

<?php
$path = str_replace('\\', '/', __FILE__);

//方法一
function ext_name1($path) {
    return substr(strrchr($path, '.'), 1);
}

//方法二
function ext_name2($path) {
    return substr($path, strrpos($path, '.')+1);
}

//方法三
function ext_name3($path) {
    $path_parts = pathinfo($path);
    return $path_parts['extension'];
}

//方法四
function ext_name4($path) {
    $sarr = explode('.', $path);
    return $sarr[count($sarr)-1];
}

//方法五
function ext_name5($path) {
    $pattern = '/^[^\.]+\.\([\w]+\)$/';
    return preg_replace($pattern, '${1}', basename($path));
}
  
```

32. PHP 的意思，它能干些什么？

PHP (Hypertext Preprocessor, 超文本预处理器的字母缩写) 是一种被广泛应用的开放源代码的多用途脚本语言，它可嵌入到 HTML 中，尤其适合 web 开发。

PHP 能做任何事。PHP 主要是用于服务端的脚本程序，因此可以用 PHP 来完成任何其它的 CGI 程序能够完成的工作，例如收集表单数据，生成动态网页，或者发送 / 接收 Cookies。但 PHP 的功能远不局限于此。

PHP 脚本主要用于以下三个领域：

- 1) 服务端脚本。这是 PHP 最传统，也是最主要的目标领域。
- 2) 命令行脚本。可以编写一段 PHP 脚本，并且不需要任何服务器或者浏览器来运行它。通过这种方式，仅仅只需要 PHP 解析器来执行。
- 3) 编写桌面应用程序。

33. Name a few ways to output (print) a block of HTML code in PHP? (Yahoo)

你可以使用 PHP 中任何一种输出语句，包括 echo、print、printf，大部分人都使用如下例的 echo：

```
echo "My string $variable";  
你也可以使用这种方法：  
echo <<<END
```

This text is written to the screen as output and this \$variable is parsed too. If you wanted you can have HTML tags in here as well. The END; remarks must be on a line of its own, and can't contain any extra white space. END;

34. 写出以下程序的输出结果 (CBSI)

```
1 <?php  
2 $b = 201;  
3 $c = 40;  
4 $a = $b > $c ? 4 : 5;  
5 echo $a;  
6 ?>
```

输出结果为 4

35. \$arr = array('james', 'tom', 'symfony'); 请打印出第一个元素的值，并将数组的数值用','号分隔并合并成字符串输出。

打印第一个元素：echo \$arr[0]; 以','合并成字符串：echo implode(',',\$arr);

36. \$a = 'abcdef'; 请取出\$a 的值并打印出第一个字母 echo \${0} 或 echo \$a[0]

相关题目：\$string="abcdefg",那么\$string{4}的值是？（卓望）值是 e

37. What does === do? What's an example of something that will give true for '==', but not '==='? (Yahoo)

=== 表示全等，是指的两个变量的值和类型都相等。

如 if (strpos('abc','a') == false) 和 if (strpos('abc','a') === false)

38. Which of the following snippets prints a representation of 42 with two decimal places? (腾讯)

- A. printf("%.2d\n", 42);
- B. printf("%1.2f\n", 42);
- C. printf("%1.2u\n", 42);

answer: B

39. Given `$text = 'Content-Type:text/xml'`; Which of the following prints 'text/xml'? (腾讯)

- A. `print substr($text, strpos($text, ':'));`
- B. `print substr($text, strpos($text, ':') + 1);`
- C. `print substr($text, strpos($text, ':') + 1);`
- D. `print substr($text, strpos($text, ':') + 2);` E. `print substr($text, 0, strpos($text, ':'));`

answer: C

40. What is the value of `$a`? (腾讯)

```
1 <?php
2     $a = in_array('01', array('1')) == var_dump('01' == 1);
3 ?>
```

- A. True B. False

answer: A

41. What is the value of `$result` in the following PHP code? (腾讯)

```
1 <?php
2     function timesTwo($int) {
3         $int = $int * 2;
4     }
5
6     $int = 2;
7     $result = timesTwo($int);
```

answer: NULL

42. What is the best all-purpose way of comparing two strings? (腾讯)

- A. Using the `strpos` function
- B. Using the `==` operator
- C. Using `strcmp()` D. Using `strcasecmp()` answer: C

43. 运行以下程序，`$a` 的值是多少？

```
1 <?php
2 $empty = '';
3 $null = NULL;
4 $bool = FALSE;
5 $notSet;
6 $array = array();
7 $a = "hello";
8 $b = &$a;
9 unset($b);
10 $b = "world";
11 echo $a;
```

hello

44. 运行以下程序，\$b 的值是多少？

```
1 <?php
2 $empty = '';
3 $null = NULL;
4 $bool = FALSE;
5 $notSet;
6 $array = array();
7 $a = 1;
8 $x = &$a;
9 $b = $a++;
10 echo $b;
```

1

45. 运行以下程序，\$x 的值是多少？

```
1 <?php
2 $empty = '';
3 $null = NULL;
4 $bool = FALSE;
5 $notSet;
6 $array = array();
7 $x = empty($array);
8 echo $x ? "true" : "false";
```

ture

46. 将字符 09 转换成十进制数字。（百度）

使用 intval 函数，echo intval("09")，或者使用 int 强制类型转换。

47. 请写一个函数，实现以下功能：字符串 "open_door" 转换成 "OpenDoor"、"make_by_id" 转换成 "MakeById"。

```

1  <?php
2  /**
3   * 字符串转换，如open_door->OpenDoor,make_by_id->MakeById
4   * @param $str string 要转换的字符串
5   * @return string 转换后的字符串
6   */
7  function change_str($str){
8      $arr = explode('_', $str);
9      $arr = array_map('ucfirst', $arr);
10     return implode('', $arr);
11 }
12 //测试
13 $str1 = "open_door";
14 $str2 = "make_by_id";
15 echo change_str($str1); //OpenDoor
16 echo change_str($str2); //MakeById

```

48. 要求写一段程序，实现以下数组\$arr1 转换成数组\$arr2:

```

$arr1 = array (
    '0' => array ('fid' => 1, 'tid' => 1, 'name' => 'Name1' ),
    '1' => array ('fid' => 1, 'tid' => 2, 'name' => 'Name2' ),
    '2' => array ('fid' => 1, 'tid' => 5, 'name' => 'Name3' ),
    '3' => array ('fid' => 1, 'tid' => 7, 'name' => 'Name4' ),
    '4' => array ('fid' => 3, 'tid' => 9, 'name' => 'Name5' )
);

```

```

$arr2 = array (
    '0' => array (
        '0' => array ( 'tid' => 1, 'name' => 'Name1'),
        '1' => array ( 'tid' => 2, 'name' => 'Name2'),
        '2' => array ( 'tid' => 5, 'name' => 'Name3'),
        '3' => array ( 'tid' => 7, 'name' => 'Name4')
    ),
    '1' => array (
        '0' => array ( 'tid' => 9, 'name' => 'Name5' )
    )
);

```

参考代码:

```

10 $arr_temp = array();
11 $arr2 = array();
12
13 foreach ($arr1 as $v) {
14     //方法一
15     $arr_temp[$v['fid']][] = array_slice($v, 1);
16     //方法二
17     // $arr_temp[$v['fid']][] = array(
18     //     'tid'=>$v['tid'],
19     //     'name'=> $v['name']
20     // );
21 }
22 foreach ($arr_temp as $v) {
23     # code...
24     $arr2[] = $v;
25 }

```

49. 如何将一个数组元素的排列顺序反转过来(例如以下代码反转以后的顺序是: `array('d', 'c', 'b', 'a')`)? (选择 2 个答案)

```

<?php
    $array = array ('a', 'b', 'c', 'd');
?>

```

- A. `array_flip()`
 B. `array_reverse()`
 C. `sort()` D. `rsort()`

答案: BD

50. `$val = max('string', array(2, 5, 7), 42);` `$val` 值为 `array(2,5,7)` max
 用法示例如下:

```

1 <?php
2 echo max(1, 3, 5, 6, 7); // 7
3 echo max(array(2, 4, 5)); // 5
4 echo max(0, 'hello'); // 0
5 echo max('hello', 0); // hello
6 echo max(-1, 'hello'); // hello
7
8 // 对多个数组, max 从左向右比较。
9 // 因此在本例中: 2 == 2, 但 4 < 5
10 $val = max(array(2, 4, 8), array(2, 5, 7)); // array(2, 5, 7)
11
12 // 如果同时给出数组和非数组作为参数, 则总是将数组视为
13 // 最大值返回
14 $val = max('string', array(2, 5, 7), 42); // array(2, 5, 7)

```

51. 定义常量 `MYPI=3.14` ; `define("MYPI",3.14);`

52. 简述单引号和双引号的用法:

- 1、双引号里面解释变量，单引号不解释变量。
- 2、双引号执行转义字符。
- 3、单引号不执行转义字符，但是它执行转义 `\\` 和 `\'`
- 4、能用单引号，就不要用双引号，因为单引号的效率更高。
- 5、如果变量和字符串在一起的时候，请用.（点）拼接字符串。
- 6、双引号里面插入单引号，单引号里面插入变量，变量会解释。
- 7、如果双引号里面插入变量的时候，请在后面跟空格或特殊字符，否则会将变量后面的字符视为一整个变量。如果不想要空格或特殊字符，请用大括号将变量包起来。

53. **Switch** 完整语法和注意事项使用 **Switch** 语句可以避免冗长的 `if..elseif..else` 代码块, `case` 只能处理整数, 或者能像整数一样运算的类型, 比如 `char`, 使用时 `case` 后用", 注意 `break` 不能少, `default` 是为了处理一些之前没有包含到的情况, 这样更为安全。

54. 用 **PHP** 编写代码在页面输出当前的北京时间, 格式为“**2007-01-18 09:22:03**”

```
date_default_timezone_set('PRC'); echo date("Y-m-d H:i:s",time());
```

55. 简述 **GBK**、**GB2312**、**BIG5** 、**GB18030**

GB2312 支持的汉字较少, **GBK** 是相比 **GB2312** 汉字更为丰富, 包括全部中日韩汉字, **GB18030** 相比 **GBK** 增加了一些少数民族汉字汉字库更为多样, 但是常人很难用到, 一般简体中文使用 **GBK** 而繁体中文使用 **BIG5**。

56. 计算某段字符串中某个字符出现的次数 (例如: **gdfgfdgd59gmkgblg** 中 **g** 的次数)

```
$text = 'gdfgfdgd59gmkgblg';  
echo substr_count ($text,'g');
```

57. 以下语句可能存在错误，如果存在错误请指出什么错误（每个语句单独考虑）

```
print_r $val = 333; // error, print_r 是一个函数，需加上括号，如果是 print 则可以  
print_r($val = 333); // 正确  
$a += ($b = 4) + 5; // notice, $a 未定义，但可以正确运行  
$foo[bar] = 'enemy'; // notice, 最好 bar 加上引号  
function x($a = "1", $b){} // 无，但给参数默认值应尽量从右向左
```

58. 写出如下程序的输出结果（小米）

```
1 <?php  
2 $str1 = null;  
3 $str2 = false;  
4 echo $str1 == $str2 ? '相等' : '不相等';
```

相等

```
1 <?php  
2 $str3 = '';  
3 $str4 = 0;  
4 echo $str3 == $str4 ? '相等' : '不相等';
```

相等

```
1 <?php  
2 $str5 = 0;  
3 $str6 = '0';  
4 echo $str5 === $str6 ? '相等' : '不相等';
```

不相等

60. 写出如下程序的输出结果

```
1 <?php  
2 $test = 'aaaaaa';  
3 $abc = &$amp;test;  
4 unset($test);  
5 echo $abc;
```

aaaaaa

61. 写出如下程序的输出结果

```
1 <?php
2 $count = 5;
3 function get_count(){
4     static $count = 0;
5     return $count++;
6 }
7 echo $count;
8 ++$count;
9 echo get_count();
10 echo get_count();
```

501

在 PHP 中，作用域是不重叠的，函数之外的是全局变量，函数内部定义的则是局部变量，二者是两个不同的变量，除非在函数内使用 `global` 显式声明使用全局变量或直接用 `$_GLOBALS` 来引用。

62. 写出如下程序的输出结果

```
1 <?php
2 $GLOBALS['var1'] = 5;
3 $var2 = 1;
4 function get_value(){
5     global $var2;
6     $var1 = 0;
7     return $var2++;
8 }
9 get_value();
10
11 echo $var1;
12 echo $var2;
```

52

63. 写出如下程序的输出结果

```
1 <?php
2 function get_arr($arr){
3     unset($arr[0]);
4 }
5 $arr1 = array(1, 2);
6 $arr2 = array(1, 2);
7
8 get_arr(&$arr1);
9 get_arr($arr2);
10
11 echo count($arr1);
12 echo count($arr2);
```

12

64. `$arr = array('james', 'tom', 'symfony');` 请将 'jack' 添加到 \$arr 数组的开头，并把 jack 打印出来。

```
array_unshift($arr,'jack');
echo $arr[0];
```

65. `$arr = array(3,7,2,1,'d','abc');` 请将 \$arr 按照从大到小的顺序排序，并保持其键值不变？

```
arsort($arr); print_r($arr);
```

66. `$mail = "gaofei@163.com"`；请将此邮箱的域(163.com)取出来并打印，看最多能写出几种方法？

```
echo    strstr($mail,'163');
echo substr($mail,7);
echo  substr($mail, strpos($mail, '@')+1);
$arr = explode("@",$mail); echo $arr[1];
```

67. PHP 中的注释的写法，要列出三种？

```
// 注释内容 C 风格的单行注释
# 注释内容 unix 风格的单行注释
```


/* 注释内容 */ C 风格的多行注释

68. 如何在一个 php 函数中使用具有全局属性的变量，说出两种方式？

global \$a 或者 \$GLOBALS['a']

69. 若\$aa='aa';则 print('aa is \'aa\'.')
会输出什么？

aa is 'aa'

70. 如何把\$a = '123'转化成 123，列出两种方式？

```
1 <?php
2 //如何把$a = '123'转化成123，列出两种方式？
3 $a = '123';
4 $a = (int)$a; //第一种方式
5 $a = intval($a); //第二种方式
6 settype($a, 'int'); //第三种方式
```

相关题目：字符串怎么转成整数，有几种方法？怎么实现？

- 1) 强制类型转换: (整型)字符串变量名;
- 2) 直接转换: settype(字符串变量,整型);
- 3) intval(字符串变量);

73. 说说 break 和 continue 可以用在哪些语句里，并且列出二者的区别？

break 和 continue 可以用在 switch 语句中，或是循环结构中

它们的区别主要是体现在循环语句中，break 表示 跳出整个循环，不再执行，循环将终止，而 continue 则表示跳出本次循环，继续下一次的循环，不会终止循环。

74. 函数中的默认参数位置如何放置？

当使用默认参数时，任何默认参数必须放在任何非默认参数的右侧；否则，函数将不会按照预期的情况工作。

75. 如何往一个数组后面添加一个值？列出两种方式？

```
$arr[] = $value;  
array_push($arr, $value);
```

76. 说出数组涉及到的常用函数。

```
array -- 声明一个数组  
count -- 计算数组中的单元数目或对象中的属性个数  
foreach -- 遍历数组  
list -- 遍历数组  
  
explode -- 将字符串转成数组  
implode -- 将数组转成一个新字符串  
array_merge -- 合并一个或多个数组  
is_array -- 检查是否是数组  
  
print_r -- 输出数组  
  
sort -- 数组排序  
  
array_keys -- 返回数组中所有的键名  
array_values -- 返回数组中所有的值  
key -- 从关联数组中取得键名
```

77. 字符串的常用函数？

```
trim() -- 去除字符串首尾处的空白字符（或者其他字符）  
strlen() -- 字符串长度  
substr() -- 截取字符串  
str_replace() -- 替换字符串函数  
substr_replace() -- 对指定字符串中的部分字符串进行替换  
strstr() -- 检索字符串函数  
explode() -- 分割字符串函数  
implode() -- 将数组合并成字符串  
str_repeat() -- 重复一个字符串  
addslashes(); -- 转义字符串  
htmlspecialchars() -- html 实体转义
```

78. 以下代码的执行后是，\$result 值为：（奇矩互动）

```
1 <?php
2 $srcArray=array('a','b','c','d');
3 $randValue=array_rand($srcArray);
4 $result=is_string($randValue);
5 ?>
```

- A. a
- B. false
- C. true
- D. b
- E. c 答案: B

`mixed array_rand(array $input [, int $num_req])`

从数组中随机取出一个或多个单元，它接受 `input` 作为输入数组和一个可选的参数 `num_req`，指明了你想取出多少个单元 — 如果没有指定，默认为 1。

如果你只取出一个，`array_rand()` 返回一个随机单元的键名，否则就返回一个包含随机键名的数组。

所以使用 `array_rand` 函数，返回结果要么是键名，要么是数组，在这个例子中，数组是索引数组，所以返回的是整型。

79. 以下代码的执行后是，`$result` 值为：（奇矩互动）

```
1 <?php
2 $a = '01';
3 $result = 1;
4 if (in_array($a,array('1'))){
5     $result = 2;
6 } elseif ($a == '1'){
7     $result = 3;
8 } elseif ($a == '01'){
9     $result = 4;
10 } else {
11     $result = 5;
12 }
```

- A. 1
 - B. 2
 - C. 3
 - D. 4
 - E. 5
- 答案: B

80. php 函数名是否区分大小写？（奇矩互动）

- A. 不区分
- B. 区分

答案：A

函数名是大小写无关的，不过在调用函数的时候，通常使用其在定义时相同的形式。

81. 以下代码的执行后是，\$result 值为：（奇矩互动）

```
1 <?php
2 $x = '';
3 $result = is_null($x);
4 ?>
```

- A. null
- B. true
- C. false
- D. 1

答案：C

is_null -- 检测变量是否为 NULL，如果变量是 null 则返回 TRUE，否则返回 FALSE。

在下列情况下一个变量被认为是 NULL：

- 1) 被赋值为 NULL
- 2) 尚未被赋值
- 3) 被 unset()

82. 翻转字符串中的单词，字符串仅包含大小写字母和空格，单词间使用空格分隔。如：输入"This is PHP"，输出"PHP is This"（非必要请不要使用 PHP 自带函数）（小米）

```
1 <?php
2 function reverse($str){
3     $arr = explode(' ', $str);
4     $arr = array_reverse($arr);
5     return implode(' ', $arr);
6 }
7
8 $str = 'This is PHP';
9 echo reverse($str);
```

83. 请列举出你所知道的全局环境变量（亿邮）

```

_ENV;
_SERVER;
_REQUEST;
_FILES;
_SESSION;
_COOKIE;
_GET;
_POST;
GLOBALS;

```

84. 假设有这样一个数组\$a=array(array(id=>0),array(id=>1),array(id=>2).....)id=>0-10000, 请问你如何使用 PHP 判断 id=>500 是否在这个数组内?（亿邮）使用 in_array 函数, 代码如下:

```

1  <?php
2  $a = array(
3      array('id' => 0),
4      array('id' => 1),
5      array('id' => 2),
6      array('id' => 500)
7  );
8  $b = array('id' => 500);
9  if (in_array($b, $a)){
10     echo 'Yes';
11 } else {
12     echo 'No';

```

85. 在 PHP 中两个数组怎么连接合并在一起（亿邮）

使用 array_merge()函数

86. 运行以上代码后\$a \$b \$c 分别是?（卓望）

```

1  <?php
2  $a = $b = $c = 0;
3  $a = $b && $c;

```

\$a 为 false, \$b = \$c = 0

87、写出下列错误代码分别代表什么意思?

Fatal error: Call to undefined method ge_user() in /website/index.php on line 39

调用了未定义的方法 ge_user(), 检查程序中有没有定义此方法

Fatal : error: Class 'client' not found in /website/index.php on line 173

类 client 没有找到, 检查文件中有没有 client 类, 或者有没有包含 client 类文件

88、写一个二维数组排序算法函数，能够具有通用性，可以调用

php 内置函数

//二维数组排序, \$arr 是数据, \$keys 是排序的键值,

\$order 是排序规则, 1 是升序, 0 是降序

```
function array_sort($arr, $keys, $order=0) {
```

```
if (!is_array($arr)) {
```

```
return false;
```

```
}
```

```
$keysvalue = array();
```

```
foreach($arr as $key => $val) {
```

```
$keysvalue[$key] = $val[$keys];
```

```
}
```

```
if($order == 0){
```

```
asort($keysvalue);
```

```
}else {
```

```
arsort($keysvalue);
```

```
}
```

```
reset($keysvalue);
```

```
foreach($keysvalue as $key => $vals) {
```

```
$keysort[$key] = $key;
```

```
}
```

```
$new_array = array();
```

90、请列出 3 种 PHP 数组循环操作的语法，并注明每种循环的优缺点。

- 1、Foreach() 可以遍历关联数组和索引数组，操作简单。
- 2、For() 只能遍历索引数组。
- 3、Each list while 以关联和索引两种方式遍历数组，操作复杂。

91、如何提高 php 应用的性能?

- 1、如果能将类的方法定义成 `static`，就尽量定义成 `static`，它的速度会提升将近 4 倍。
- 2、`$row['id']` 的速度是 `$row[id]` 的 7 倍。
- 3、`echo` 比 `print` 快，并且使用 `echo` 的多重参数
(译注：指用逗号而不是句点)代替字符串连接，比如 `echo$str1,$str2`。
- 4、在执行 `for` 循环之前确定最大循环数，不要每循环一次都计算最大值，最好运用 `foreach` 代替。
- 5、注销那些不用的变量尤其是大数组，以便释放内存。
- 6、尽量避免使用 `__get`，`__set`，`__autoload`。
- 7、`require_once()` 代价昂贵。
- 8、`include` 文件时尽量使用绝对路径，因为它避免了 PHP 去 `include_path` 里查找文件的速度，解析操作系统路径所需的时间会更少。
- 9、如果你想知道脚本开始执行(译注：即服务器端收到客户端请求)的时刻，使用 `$_SERVER['REQUEST_TIME']` 要好于 `time()`。
- 10、函数代替正则表达式完成相同功能。
- 11、`str_replace` 函数比 `preg_replace` 函数快，但 `strtr` 函数的效率是 `str_replace` 函数的四倍。
- 12、如果一个字符串替换函数，可接受数组或字符作为参数，并且参数长度不太长，那么可以考虑额外写一段替换代码，使得每次传递参数是一个字符，而不是只写一行代码接受数组作为查询和替换的参数。
- 13、使用选择分支语句(译注：即 `switch case`)好于使用多个 `if`，`else if` 语句。
- 14、用 `@` 屏蔽错误消息的做法非常低效，极其低效。
- 15、打开 `apache` 的 `mod_deflate` 模块，可以提高网页的浏览速度。
- 16、数据库连接当使用完毕时应关掉，不要用长连接。
- 17、错误消息代价昂贵。
- 18、在方法中递增局部变量，速度是最快的。几乎与在函数中调用局部变量的速度相当。
- 19、递增一个全局变量要比递增一个局部变量慢 2 倍。
- 20、递增一个对象属性(如：`$this->prop++`)要比递增一个局部变量慢 3 倍。
- 21、递增一个未预定义的局部变量要比递增一个预定义的局部变量慢 9 至 10 倍。
- 22、仅定义一个局部变量而没在函数中调用它，同样会减慢速度(其程度相当于递增一个局部变量)。PHP 大概会检查是否存在全局变量。
- 23、方法调用看来与类中定义的方法的数量无关，因为我(在测试方法之前和之后都)添加了 10 个方法，但性能上没有变化。
- 24、派生类中的方法运行起来要快于在基类中定义的同样的方法。

- 25、调用带有一个参数的空函数，其花费的时间相当于执行 7 至 8 次的局部变量递增操作。类似的方法调用所花费的时间接近于 15 次的局部变量递增操作。
- 26、Apache 解析一个 PHP 脚本的时间要比解析一个静态 HTML 页面慢 2 至 10 倍。尽量多用静态 HTML 页面，少用脚本。
- 27、除非脚本可以缓存，否则每次调用时都会重新编译一次。引入一套 PHP 缓存机制通常可以提升 25% 至 100% 的性能，以免除编译开销。
- 28、尽量做缓存，可使用 memcached。memcached 是一款高性能的内存对象缓存系统，可用来加速动态 Web 应用程序，减轻数据库负载。对运算码 (OP code) 的缓存很有用，使得脚本不必为每个请求做重新编译。
- 29、当操作字符串并需要检验其长度是否满足某种要求时，你想当然地会使用 strlen() 函数。此函数执行起来相当快，因为它不做任何计算，只返回在 zval 结构(C 的内置数据结构，用于存储 PHP 变量)中存储的已知字符串长度。但是，由于 strlen()是函数，多多少少会有些慢，因为函数调用会经过诸多步骤，如字母小写化(译注：指函数名小写化，PHP 不区分函数名大小写)、哈希查找，会跟随被调用的函数一起执行。在某些情况下，你可以使用 isset() 技巧加速执行你的代码。调用 isset()恰巧比 strlen()快，因为与后者不同的是，isset()作为一种语言结构，意味着它的执行不需要函数查找和字母小写化。也就是说，实际上在检验字符串长度的顶层代码中你没有花太多开销。
- 34、当执行变量 \$i 的递增或递减时，\$i++ 会比 ++\$i 慢一些。这种差异是 PHP 特有的，并不适用于其他语言，所以请不要修改你的 C 或 Java 代码并指望它们能立即变快，没用的。++\$i 更快是因为它只需要 3 条指令(opcodes)，\$i++则需要 4 条指令。后置递增实际上会产生一个临时变量，这个临时变量随后被递增。而前置递增直接在原值上递增。这是最优化处理的一种，正如 Zend 的 PHP 优化器所作的那样。牢记这个优化处理不失为一个好主意，因为并不是所有的指令优化器都会做同样的优化处理，并且存在大量没有装配指令优化器的互联网服务提供商(ISPs)和服务器。
- 35、并不是事必面向对象(OOP)，面向对象往往开销很大，每个方法和对象调用都会消耗很多内存。
- 36、并非要用类实现所有的数据结构，数组也很有用。
- 37、不要把方法细分得过多，仔细想想你真正打算重用的是哪些代码？
- 38、当你需要时，你总能把代码分解成方法。
- 39、尽量采用大量的 PHP 内置函数。
- 40、如果在代码中存在大量耗时的函数，你可以考虑用 C 扩展的方式实现它们。
- 41、评估检验(profile)你的代码。检验器会告诉你，代码的哪些部分消耗了多少时间。Xdebug 调试器包含了检验程序，评估检验总体上可以显示出代码的瓶颈。
- 42、mod_zip 可作为 Apache 模块，用来即时压缩你的数据，并可让数据传输量降低 80%。

- 43、在可以用 `file_get_contents` 替代 `file`、`fopen`、`feof`、`fgets` 等系列方法的情况下，尽量用 `file_get_contents`，因为他的效率高得多!但是要注意 `file_get_contents` 在打开一个 URL 文件时候的 PHP 版本问题;
- 44、尽可能的少进行文件操作，虽然 PHP 的文件操作效率也不低的;
- 45、优化 Select SQL 语句，在可能的情况下尽量少的进行 Insert、Update 操作;
- 46、尽可能的使用 PHP 内部函数(但是我却为了找个 PHP 里面不存在的函数，浪费了本可以写出一个自定义函数的时间，经验问题啊!);
- 47、循环内部不要声明变量，尤其是大变量：对象(这好像不只是 PHP 里面要注意的问题吧?);
- 48、多维数组尽量不要循环嵌套赋值;
- 49、在可以用 PHP 内部字符串操作函数的情况下，不要用正则表达式;
- 50、foreach 效率更高，尽量用 foreach 代替 while 和 for 循环;
- 51、用单引号替代双引号引用字符串;
- 52、“用 `i+=1` 代替 `i=i+1`。符合 c/c++的习惯，效率还高”;
- 53、对 global 变量，应该用完就 `unset()`掉。

92、写一个函数，能够遍历一个文件夹下的所有文件和子文件夹。

```
<?php

function my_scandir($dir)
{
    $files = array();
    if ( $handle = opendir($dir) ) {
        while ( ($file = readdir($handle)) !== false ) {
            if ( $file != "." && $file != ".." ) {
                if ( is_dir($dir . "/" . $file) ) {
                    $files[$file] = my_scandir($dir . "/" . $file);
                }else {
                    $files[] = $file;
                }
            }
        }
        closedir($handle);
        return $files;
    }
}
```

93、写一个函数，尽可能高效的，从一个标准 url 里取出文件的扩展名 例如: `http://www.sina.com.cn/abc/de/fg.php?id=1` 需要取出 `php` 或 `.php`

答案 1:

```
function getExt($url){
    $arr = parse_url($url);
    $file = basename($arr['path']);
    $ext = explode(".", $file);
    return $ext[1];
}
```

答案 2:

```
function getExt($url) {
    $url = basename($url);
    $pos1 = strpos($url, ".");
    $pos2 = strpos($url, "?");
    if(strpos($url, "?")){
        return substr($url, $pos1 + 1, $pos2 - $pos1 - 1);
    } else {
        return substr($url, $pos1);
    }
}
```

94、请写一个函数验证电子邮件的格式是否正确

```
function checkemail($email){
    echo preg_match( '/^\[\w]+\@[\w]+\.[\w]+\$/ ',$email) ? 'email 格式正确'
    : 'email 格式不正确';
}
```

95、写出匹配 L URL 的正则表达式。

```
/^[a-zA-z]+:\V\(\w+(-\w+)*\)\.(\w+(-\w+)*)*(\?S*)?/
```

96、有一个网页地址，比如开发资源网主页：

<http://www.phpres.com/index.html>，如何得到它的内容？

获取网页内容：

```
$url="http://www.phpres.com/index.html";
$str=file_get_contents($url);
```

97、@all 单文件 4G，内存只有 128M，请问如何找到这个文件中任意位置出现的 abc 三个字符。

实现办法：方案一 用 SplfileObject，一种是用 KMP 算法。

KMP 算法思想：

```
//1 打开文件

//2 循环读取文件，每次 100M，然后调用 KMP 算法查找和'ABC'相等的字符串的位置，
将位置存到数组 pos 中

//3 读取完毕后 关闭文件

//4 数组 pos 保存了 abc 字符串在文件中的位置

//计算部分匹配表和字符串匹配算法

//Tstring 源串
//Pstring 要查找的串
//next next 数组
// pos 保存位置
//start 起始位置

function kmp($Tstring, $Pstring, &$amp;next, &$amp;pos, $start)
{
    $n = strlen($Tstring); // 字符串
    $m = strlen($Pstring);

    for ($i = 0, $q = 0; $i < $n; ++$i)
    {
        while($q > 0 && $Pstring[$q] != $Tstring[$i])
            $q = $next[$q-1];

        if ($Pstring[$q] == $Tstring[$i])
```

```
        {
            $q++;
        }
        if ($q == $m)
        {
            //记录找到的位置
            $pos[] = $start + $i - $m + 1;
        }
    }
}

//计算 next 数组
function makeNext($Pstring, &$next)
{
    $m = strlen($Pstring); // 此时为数组
    $next[0] = 0;
    for ($q = 1, $k = 0; $q < $m; ++$q)
    {
        while($k > 0 && $Pstring[$q] != $Pstring[$k])
            $k = $next[$k-1];
        if ($Pstring[$q] == $Pstring[$k])
        {
            $k++;
        }
        $next[$q] = $k;
    }
}

$next = array(0); //保存 next 数组
```

```
$pos = [];//位置数组

//计算 next 数组

makeNext($Pstring,$next);

//打开文件

$fp = fopen("filename.txt",'r');

$filePos = 0;//文件块

//每次读 100M

$len = 1024 * 1024 * 100;//100M

while(1) {

    //一次读 100M

    $content = fread($fp,$len);//要查找的内容

    //查找相等字符串的位置，位置放到了$pos 数组中

    kmp($content, 'abc', $next,$pos,$filePos * $len);

    $filePos++;//块数++

    //向后移动文件指针

    fseek($fp,$filePos * $len);

    //读完，关闭

    if (!$fp) {

        break;

    }

}

fclose($fp);//关闭文件
```

98、以下代码的输出的结果是什么？

```
$arr = array(5=>1, 12=>2);  
$arr[] = 56;  
$arr["x"] =>42;  
var_dump($arr);
```

输出结果为:

```
array(4) { [5]=>int(1) [12]=>int(2) [13]=>int(56) ["x"]=>int(42) }
```

99、

以下代码的输出结果为:

```
$father = "mother";  
$mother = "son";  
echo $$father;
```

输出的结果为: son;

100、

```
//以下代码的输出结果为:  
$x = array("aaa", "", "ccc", "ddd", "");  
$y = array_unique($x);  
echo count($x) . ", " . count($y);
```

输出的结果为 5 4

101、

```
//下列代码的输出是
$x=dir(".");
while($y=$x->read())
{
    echo $y;
}
$x->close();
```

答案是：显示当前文件夹下的所有文件名。

PHP 面向对象

1. 写出 php 的 `public`、`protected`、`private` 三种访问控制模式的区别
(新浪网技术部)

`public`: 公有，任何地方都可以访问 `protected`: 继承，只能在本类或子类中访问，在其它地方不允许访问 `private`: 私有，只能在本类中访问，在其他地方不允许访问

相关题目：请写出 **PHP7** 权限控制修饰符

`private` `protected` `public`

2. 设计模式考察：请用单态设计模式方法设计类满足如下需求

请用 **PHP** 代码编写类实现在每次对数据库连接的访问中都只能获得唯一的一个数据库连接，具体连接数据库的详细代码忽略，请写出主要逻辑代码（新浪网技术部）

```

1 <?php
2 class Mysql{
3     private static $instance = null;
4     private $conn;
5     //构造方法,设置为privege, 不允许通过new获得对象
6     private function __construct(){
7         $conn = mysql_connect('localhost','root','123456');
8     }
9     //获取实例方法
10    public static function getInstance(){
11        if (! self::$instance instanceof self){
12            self::$instance = new self;
13        }
14        return self::$instance;
15    }
16    //禁止克隆
17    private function __clone(){ }
18 }
19
20 //获得对象
21 $db = Mysql::getInstance();

```

3. 写出下列程序的输出结果（新浪网技术部）

```

1 <?php
2 class a{
3     protected $c;
4     public function a(){
5         $this->c = 10;
6     }
7 }
8 class b extends a{
9     public function print_data(){
10        return $this->c;
11    }
12 }
13 $b = new b();
14 echo $b->print_data();

```

输出结果 10

4. PHP5 中魔术方法函数有哪几个，请举例说明各自的用法（腾讯 PHP 工程师笔试题）

- __sleep serialize 之前被调用
- __wakeup unserialize 时被调用
- __toString 打印一个对象时被调用

__set_state 调用 var_export 时被调用, 用__set_state 的返回值作为 var_export 的返回值

__construct 构造函数, 实例化对象时被调用

__destruct 析构函数, 当对象销毁时被调用

__call 对象调用某个方法, 若存在该方法, 则直接调用, 若不存在, 则调用__call 函数

__callStatic 调用一个不存在的静态的方法的时候, 自动触发。

__get 读取一个对象的私有属性时, 若属性存在, 则直接返回, 若不存在, 则调用__get 函数

__set 设置一个对象的私有属性名时, 若属性存在, 则直接赋值, 若不存在, 则调用__set 函数

__isset 检测一个对象的私有属性是否存在时被调用

__unset 删除或销毁 一个对象的私有属性时被调用

__clone 克隆对象时被调用

__autoload 实例化一个对象时, 如果对应的类不存在, 则该方法被调用

__invoke 把实例化的对象当成函数使用的时候, 自动触发 invoke 方法。

__debugInfo() var_dump 一个对象的时候, 如果写了 debugInfo 自动执行 debugInfo. 返回值必须返回一个数组。

相关题目: 请写出 **php** 的构造函数和析构函数

构造函数: __construct 析构函数: __destruct

5. 如何使用下面的类,并解释下面什么意思?

```
1 <?php
2 class test{
3     function Get_test($num){
4         $num=md5(md5($num)."En");
5         return $num;
6     }
7 }
```

双重 md5 加密

\$testObject = new test();

\$sencryption = \$testObject->Get_test("itcast");

6. How would you declare a class named “myclass” with no methods or properties? (Yahoo)

```
class myclass{}
```

相关题目：如何声明一个名为“myclass”的没有方法和属性的类？

7. How do you access and set properties of a class from within the class?

(Yahoo)

使用语句：\$this->propertyName, 例如：

```
1 <?php
2 class myclass{
3     private $propertyName;
4     public function __construct(){
5         $this->propertyName = "value";
6     }
7 }
```

8. The code below _____ because _____. (腾讯)

```
1 <?php
2 class Foo{
3     ?>
4 <?php
5     function bar(){
6         print "bar";
7     }
8 }
9 ?>
```

- A. will work, class definitions can be split up into multiple PHP blocks.
- B. will not work, class definitions must be in a single PHP block.
- C. will not work, class definitions must be in a single file but can be in multiple PHP blocks.
- D. will work, class definitions can be split up into multiple files and multiple PHP blocks.

答案： B

10. 类的属性可以序列化后保存到 session 中,从而以后可以恢复整个类,这要用到的函数是_____。

serialize() 和 unserialize()

11. 接口和抽象类的区别是什么？

抽象类是一种不能被实例化的类，只能作为其他类的父类来使用。抽象类是通过关键字 `abstract` 来声明的。

抽象类与普通类相似，都包含成员变量和成员方法，两者的区别在于，抽象类中至少要包含一个抽象方法，抽象方法没有方法体，该方法天生就是要被子类重写的。

抽象方法的格式为：`abstract function abstractMethod();`

接口是通过 `interface` 关键字来声明的，接口中的成员常量和方法都是 `public` 的，方法可以不写关键字 `public`，

接口中的方法也是没有方法体。接口中的方法也天生就是要被子类实现的。

抽象类和接口实现的功能十分相似，最大的不同是接口能实现多继承。在应用中选择抽象类还是接口要看具体实现。

子类继承抽象类使用 `extends`，子类实现接口使用 `implements`。

13. 类中如何定义常量、如何类中调用常量、如何在类外调用常量。

类中的常量也就是成员常量，常量就是不会改变的量，是一个恒值。定义常量使用关键字 `const`，例如：`const PI = 3.1415326;`

无论是类内还是类外，常量的访问和变量是不一样的，常量不需要实例化对象，

访问常量的格式都是类名加作用域操作符号（双冒号）来调用，即：类名:: 类常量名。

14. `autoload()`函数是如何运作的？

使用这个魔术函数的基本条件是类文件的文件名要和类的名字保持一致。

当程序执行到实例化某个类的时候，如果在实例化前没有引入这个类文件，那么就自动执行 `__autoload()` 函数。这个函数会根据实例化的类的名称来查找这个类文件的路径，当判断这个类文件路径下确实存在这个类文件后就执行 `include` 或者 `require` 来载入该类，然后程序继续执行，如果这个路径下不存在该文件时就提示错误。

15. 哪种 OOP 设置模式能让类在整个脚本里只实例化一次？（奇矩互动）

- A. MVC
- B. 代理模式
- C. 状态模式
- D. 抽象工厂模式
- E. 单件模式

- F. 答案：E

16. 借助继承，我们可以创建其他类的派生类。在 PHP 中，子类最多可以继承几个父类？（奇矩互动）

- A. 1 个
- B. 2 个
- C. 取决于系统资源
- D. 3 个
- E. 想要几个有几个

答案：A

17. 执行以下代码，输出结果是（奇矩互动）

```
1 <?php
2 abstract class a{
3     function __construct(){
4         echo 'a';
5     }
6 }
7 $a = new a();
8 ?>
```

- A. a
- B. 一个错误警告
- C. 一个致命性的报错

答案：C 因为类 a 是抽象类，不能被实例化

18. 执行以下代码，输出结果是

```

1  <?php
2  class a{
3      function __construct(){
4          echo 'echo class a something';
5      }
6  }
7  class b extends a{
8      function __construct(){
9          echo 'echo class b something';
10     }
11 }
12 $a = new b();

```

- A. echo class a something echo class b something
- B. echo class b something echo class a something
- C. echo class a something
- D. echo class b something 答案: D

类 b 继承自类 a，两个类都定义了构造函数，由于二者名字相同，所以子类中的构造函数覆盖了父类的构造函数，要想子类对象实例化时也执行父类的构造函数，需要在子类构造函数中使用 `parent::__construct()` 来显示调用父类构造函数。

19. 请定义一个名为 **MyClass** 的类，这个类只有一个静态方法 **justDoIt**。（卓望）

```

1  <?php
2  class MyClass{
3      public static function justDoIt(){
4
5      }
6  }

```

20. 写出你知道的几种设计模式，并用 **php** 代码实现其中一种。（卓望）

单例模式，工厂模式，观察者模式，门面模式，反射，适配器模式，策略设计模式，依赖注入等等。

反射设计模式的代码：

```
class UserModel
{

}

class Person
{

    public function __construct()
    {

    }

    public function getAbc(UserModel $user, $a, $b)
    {

    }

}

$person = new Person();

$ref = new ReflectionMethod($person, 'getAbc');

$params = $ref->getParameters();

$paramsReflection = $params[0]->getClass();

$model = new $paramsReflection->name;
```

观察者设计模式的代码：

```

<?php
class AbstractAlly
{
    public $playersCollection;

    //将所有的对象添加进入$oberServerCollection集合中
    public function addOberServer($play)
    {
        $this->playersCollection[] = new OberServer($play);
    }

    public function notify($play)
    {
        //一旦攻击行为被触发，则其他人去帮助这个被攻击的人。
        foreach ($this->playersCollection as $obServer)
        {
            //如果帮助者， 不等于被帮助者的名字的时候。help
            if ($obServer->name != $play) {
                $obServer->help($play);
            }
        }
    }

    abstract public function attack($play);
}

class Ally extends AbstractAlly
{
    public function __construct($players)
    {
        foreach ($players as $play) {
            //将所有的对象添加进入$oberServerCollection集合中
            $this->addOberServer($play);
        }
    }

    //某一个人被攻击了
    public function attack($play)
    {
        //其他人都在监听是否盟友被攻击了
        $this->notify($play);
    }
}

```

```
interface IOberServer
{
    public function help($play);
}

class OberServer implements IOberServer
{
    public $name;

    public function __construct($name)
    {
        $this->name = $name;
    }

    //执行真正的帮助行为
    public function help($play)
    {
        echo $this->name . '帮助' . $play . '<br />';
    }
}

//定义所有的玩家
$allPlayer = ['子旋', '刘国华', '王浩伟', '齐娜'];

//让所有的玩家结盟，并切加入到监听里
$ally = new Ally($allPlayer);

$ally->attack('子旋');
```

适配器模式的代码：


```

class Busi
{
    public $blood;

    public function restoreBlood()
    {
        echo '自动的在回血<br />';
    }
}

class Dirty
{
    public function dirtySelf()
    {
        echo '把自己搞得脏xixi的';
    }
}

class Dog extends Busi
{
    public $dirty;

    public function __construct()
    {
        $this->dirty = new Dirty();
    }

    public function dagun()
    {
        $this->dirty->dirtySelf();

        echo '在地上打滚<br />';
    }
}

$dog = new Dog();

$dog->dagun();

```

21. 什么是面向对象？主要特征是什么？几大原则是什么？

面向对象是程序的一种设计模式，它利于提高程序的重用性，使程序机构更加清晰。主要特征是：封装、继承、多态。五大基本原则：单一职责原则；开放封闭原则；替换原则；依赖原则；接口分离原则。

MySQL 数据库

1. 写出下面 2 个 PHP 操作 Mysql 函数的作用和区别（新浪网技术部）

`mysqli_num_rows()`

`mysqli_affected_rows()`

这两个函数都作用于 `mysqli_query($query)` 操作的结果，`mysqli_num_rows()` 返回结果集中行的数目。`mysqli_affected_rows()` 取得前一次 MySQL 操作所影响的记录行数。`mysqli_num_rows()` 仅对 SELECT 语句有效，要取得被 INSERT, UPDATE 或者 DELETE 查询所影响到的行的数目，用 `mysqli_affected_rows()`。

2. sql 语句应该考虑哪些安全性？（新浪网技术部）

防止 Sql 注入，对特殊字符进行转义、过滤或者使用预编译的 sql 语句绑定变量。

最小权限原则，特别是不要用 root 账户，为不同的类型的动作或者组建使用不同的账户。

当 sql 运行出错时，不要把数据库返回的错误信息全部显示给用户，以防止泄露服务器和数据库相关信息。

3. 简单描述 mysql 中，索引，主键，唯一索引，联合索引的区别，对数据库的性能有什么影响（从读写两方面）（新浪网技术部）

索引是一种特殊的文件(InnoDB 数据表上的索引是表空间的一个组成部分)，它们包含着对数据表里所有记录的引用指针。

普通索引(由关键字 KEY 或 INDEX 定义的索引)的唯一任务是加快对数据的访问速度。

普通索引允许被索引的数据列包含重复的值。如果能确定某个数据列将只包含彼此各不相同的值，在为这个数据列创建索引的时候就应该用关键字 UNIQUE 把它定义为一个唯一索引。也就是说，唯一索引可以保证数据记录的唯一性。

主键，是一种特殊的唯一索引，在一张表中只能定义一个主键索引，主键用于唯一标识一条记录，使用关键字 PRIMARY KEY 来创建。

索引可以覆盖多个数据列，如像 `INDEX(columnA, columnB)` 索引，这就是联合索引。

索引可以极大的提高数据的查询速度，但是会降低插入、删除、更新表的速度，因为在执行这些写操作时，还要操作索引文件。

4. 有一个留言板，用 mysql 做数据库，用户信息包括：用户名，密码，email，留言内容包括：留言 ID，标题，内容，发表时间，状态（审核，未审核）（新浪网技术部）请实现下列需求：

(1). 数据库结构。无需写建表语句，用类似下面的表格，描述清楚即可，注意，要在索引栏中注明是否需要创建索引，以及要创建的索引的类型

表名	table_aaa		
字段名	字段说明	字段类型	索引
name	姓名	varchar (64)	唯一索引
gender	性别	enum ('M', 'F')	

(2). 用一个 sql 语句查询出发表留言数量大于 10 条的用户名及其留言数量，查询结果按文章数量降序排列
 参考答案：用户表结构如下：

表名	user		
字段名	字段说明	字段类型	索引
user_id	用户编号	int unsigned	主键
name	用户名	varchar(30)	
password	密码	char (32)	
email	邮箱	varchar (50)	

留言表结构如下：

表名	message		
字段名	字段说明	字段类型	索引
message_id	留言编号	int unsigned	主键
title	标题	varchr(100)	
content	内容	text	
user_id	用户 id	int unsigned	普通索引
pubtime	发表时间	int unsigned	
state	状态	tinyint 0 未审核 1 审核	

查询语句如下：

```
SELECT u.name, COUNT(*) AS total
FROM user AS u INNER JOIN message AS m
```

```
ON u.user_id = m.user_id
GROUP BY u.name
HAVING total > 10
ORDER BY total DESC
```

5. 如何用命令把 mysql 里的数据备份出来（酷讯 PHP 工程师笔试题）

- (1). 导出一张表 `mysqldump -u 用户名 -p 密码 库名 表名 > 文件名(如 D:/a.sql)`
- (2). 导出多张表 `mysqldump -u 用户名 -p 密码 库名 表名 1 表名 2 表名 3 > 文件名(如 D:/a.sql)`
- (3). 导出所有表 `mysqldump -u 用户名 -p 密码 库名 > 文件名(如 D:/a.sql)`
- (4). 导出一个库 `mysqldump -u 用户名 -p 密码 -B 库名 > 文件名(如 D:/a.sql)`

6. 两张表 city 表和 province 表。分别为城市与省份的关系表。

city:

id	city	provinceid
1	广州	1
2	深圳	1
3	惠州	1
4	长沙	2
5	武汉	3

province:

id	province
1	广东
2	湖南
3	湖北

- (1). 写一条 sql 语句关系两个表，实现：显示城市的基本信息。

显示字段：城市 id ， 城市名， 所属省份 。如：

id (城市 id) cityname(城市名) privence(所属省份)

.....

```
SELECT c.id AS id,c.city AS cityname,p.province
FROM city c LEFT JOIN province p ON c.provinceid=p.id
```

(2). 如果要统计每个省份有多少个城市，请用 group by 查询出来。
显示字段：省份 id ， 省份名， 包含多少个城市。

```
SELECT p.id,p.province,count(c.id) AS num
FROM province p LEFT JOIN city c ON p.id = c.provinceid
GROUP BY p.id;
```

7. MySQL 数据库中的字段类型 varchar 和 char 的主要区别是什么？哪种字段的查找效率要高，为什么？

区别一，定长和变长

char 表示定长，长度固定，varchar 表示变长，即长度可变

当所插入的字符串超出它们的长度时，视情况来处理，如果是严格模式，则会拒绝插入并提示错误信息，如果是宽松模式，则会截取然后插入。如果插入的字符串长度小于定义长度时，则会以不同的方式来处理，如 char (10)，表示存储的是 10 个字符，无论你插入的是多少，都是 10 个，如果少于 10 个，则用空格填满。而 varchar (10)，小于 10 个的话，则插入多少个字符就存多少个。varchar 怎么知道所存储字符串的长度呢？实际上，对于 varchar 字段来说，需要使用一个（如果字符串长度小于 255）或两个字节（长度大于 255）来存储字符串的长度。

区别之二，存储的容量不同对 char 来说，最多能存放的字符个数 255，和编码无关。

而 varchar 呢，最多能存放 65532 个字符。

VARCHAR 的最大有效长度由最大行大小和使用的字符集确定。整体最大长度是 65,532 字节

最大有效长度是 65532 字节，在 varchar 存字符串的时候，第一个字节是空的，不存任何的数据，然后还需要两个字节来存放字符串的长度。所以有效长度就是 $65535 - 1 - 2 = 65532$

由字符集来确定，字符集分单字节和多字节

Latin1 一个字符占一个字节，最多能存放 65532 个字符

GBK 一个字符占两个字节，最多能存 32766 个字符 UTF8 一个字符占三个字节，最多能存 21844 个字符**注意，char 和 varchar 后面的长度表示的是字符的个数，而不是字节数。**

两相比较，char 的效率高，没有碎片，尤其更新比较频繁的时候，方便数据文件指针

的操作。但不够灵活，在实际使用时，应根据实际需求来选用合适的数据类型。

相关题目：若一个表定义为 `create table t1(c int, c2 char(30), c3 varchar(N)) charset=utf8;` 问 N 的最大值又是多少？（65535 - 1 - 2 - 4 - 30 * 3）/3

8. IP 该如何保存?

最简单的办法是使用字符串（varchar）来保存，如果从效率考虑的话，可以将 ip 保存为整型（unsigned int），使用 php 或 mysql 提供的函数将 ip 转换为整型，然后存储即可。

PHP 函数：long2ip() 和 ip2long()

MySQL 函数：inet_ntop() 和 inet_pton

9. 设有成绩表如下所示，查询两门及两门以上不及格同学的平均分。

编号	姓名	科目	分数
1	张三	数学	90
2	张三	语文	50
3	张三	地理	40
4	李四	语文	55
5	李四	政治	45
6	王五	政治	30

#创建一个成绩表

```
CREATE TABLE grade( id INT UNSIGNED NOT NULL
    AUTO_INCREMENT PRIMARY KEY, name VARCHAR(10) NOT
    NULL, subject VARCHAR(10) NOT NULL, score TINYINT
    UNSIGNED NOT NULL
);
```

#插入记录

```
INSERT INTO grade(name,subject,score) VALUES('张三','数学',90);
INSERT INTO grade(name,subject,score) VALUES('张三','语文',50);
INSERT INTO grade(name,subject,score) VALUES('张三','地理',40);
INSERT INTO grade(name,subject,score) VALUES('李四','语文',55);
INSERT INTO grade(name,subject,score) VALUES('李四','政治',45);
INSERT INTO grade(name,subject,score) VALUES('王五','政治',30);
```

#查询语句

```
SELECT name,AVG(score),SUM(score<60) AS gk
FROM grade
```

```
GROUP BY name
HAVING gk>=2;
```

10. 为了记录足球比赛的结果，设计表如下，

team: 参赛队伍表

字段名称	类型	描述
teamID	int	主键
teamname	varchar(20)	队伍名称

match: 赛程表

字段名称	类型	描述
matchID	int	主键
hostTeamID	int	主队的 ID
gusetTeamID	int	客队的 ID
marchResult	varchar(20)	比赛结果
matchTime	date	比赛日期

其中，match 赛程表中的 hostTeamID 与 guestTeamID 都和 team 表中的 teamID 关联，查出

2006-6-1 到 2006-7-1 之间举行的所有比赛，并且用以下形式列出：拜仁 2:0 不莱梅 2006-6-21

#创建参赛队伍表

```
CREATE TABLE team( teamID INT UNSIGNED NOT NULL
    AUTO_INCREMENT PRIMARY KEY, teamName VARCHAR(30) NOT
    NULL
);
```

#向参赛队伍表中插入记录

```
INSERT INTO team(teamName) VALUES('拜仁');
INSERT INTO team(teamName) VALUES('不莱梅');
INSERT INTO team(teamName) VALUES('皇家马德里');
INSERT INTO team(teamName) VALUES('巴塞罗那');
INSERT INTO team(teamName) VALUES('切尔西');
INSERT INTO team(teamName) VALUES('曼联');
INSERT INTO team(teamName) VALUES('AC 米兰');
INSERT INTO team(teamName) VALUES('国际米兰');
```

#创建 match 赛程表

```
CREATE TABLE mat( matchID INT UNSIGNED NOT NULL
    AUTO_INCREMENT PRIMARY KEY, hostTeamID INT UNSIGNED NOT
    NULL, guestTeamID INT UNSIGNED NOT NULL, matchResult
    VARCHAR(20) NOT NULL, matchTime DATE NOT NULL
);
```

#向赛程表中插入几条记录

```
INSERT INTO mat(hostTeamID,guestTeamID,matchResult,matchTime)
VALUES(1,2,'3:1','2006-6-15');
INSERT INTO mat(hostTeamID,guestTeamID,matchResult,matchTime)
VALUES(3,4,'2:2','2006-6-28');
INSERT INTO mat(hostTeamID,guestTeamID,matchResult,matchTime)
VALUES(5,6,'0:2','2006-7-10');
INSERT INTO mat(hostTeamID,guestTeamID,matchResult,matchTime)
VALUES(7,8,'5:3','2006-5-30');
```

#查询语句

```
SELECT t1.teamName,matchResult,t2.teamName,matchTime FROM mat
LEFT JOIN team AS t1
ON hostTeamID = t1.teamID
LEFT JOIN team AS t2
ON guestTeamID = t2.teamID
WHERE matchTime BETWEEN '2006-6-1' AND '2006-7-1';
```

12、写出发贴数最多的十个人名字的 SQL，利用下表：

```
members(id,username,posts,pass,email)
SELECT members.username
FROM members
ORDER BY posts DESC
LIMIT 10;
```

12. 请简述项目中优化 sql 语句执行效率的方法,从哪些方面,sql 语句性能如何分析?

- 1) 尽量选择较小的列
- 2) 将 where 中用的比较频繁的字段建立索引
- 3) select 子句中避免使用 ‘*’
- 4) 避免在索引列上使用计算、not in 和 <> 等操作
- 5) 当只需要一行数据的时候使用 limit 1
- 6) 保证单表数据不超过 200W，适时分割表。针对查询较慢的语句，可以使用 explain 来分析该语句具体的执行情况。

15. SQL 中 LEFT JOIN 的含义是_____，如果 tbl_user 记录了学生的姓名(name)和学号(ID)，tbl_score 记录了学生（有的学生考试以后被开除了，没有其记录）的学号(ID)和考试成绩(score)以及考试科目(subject)，要想打印出各个学生姓名及对应的各科总成绩，则可以用 SQL 语句_____。

left join 表示左外连接，以左表为准，左表中的记录都会出现在查询结果中，如果对应的记录在右表中没有匹配的记录，则右表的字段值以 NULL 填充。

#创建表 tbl_user

```
CREATE TABLE tbl_user( id INT NOT
    NULL, name VARCHAR(50) NOT
    NULL, PRIMARY KEY (id)
```

);

#创建表 tbl_score

```
CREATE TABLE tbl_score( id INT NOT
    NULL, score DEC(6,2) NOT NULL,
    subject VARCHAR(20) NOT NULL
```

);

#插入记录

```
INSERT INTO tbl_user (id, name) VALUES (1, 'beimu');
```

```
INSERT INTO tbl_user (id, name) VALUES (2, 'aihui');
```

```
INSERT INTO tbl_score (id, score, subject) VALUES (1, 90, '语文');
```

```
INSERT INTO tbl_score (id, score, subject) VALUES (1, 80, '数学');
```

```
INSERT INTO tbl_score (id, score, subject) VALUES (2, 86, '数学');
```

```
INSERT INTO tbl_score (id, score, subject) VALUES (2, 96, '语文');
```

#查询语句

```
SELECT a.id,SUM(b.score) AS sumscore
```

```
FROM tbl_user a LEFT JOIN tbl_score b
```

```
ON a.id=b.id
```

```
GROUP BY a.id;
```

16. 使用 php 写一段简单查询，查出所有姓名为“张三”的内容并打印出来

表 user

name	tel	content	date
张三	13333663366	大专毕业	2006-10-11
张三	13612312331	本科毕业	2006-10-15
张四	021-55665566	中专毕业	2006-10-15

请根据上面的题目完成代码：

```
$mysql_db=mysql_connect("local","root","pass");
mysql_select_db("DB",$mysql_db);
$result=mysql_query("select * from user where name='张三'");
while($row=mysql_fetch_array($result)){ echo $row['name'].
$row['tel']. $row['content']. $row['date']; echo "<br>";
}
```

17. 写出 SQL 语句的格式：插入，更新，删除（卓望）

表名 user

name	tel	content	date
张三	13333663366	大专毕业	2006-10-11
	13612312331	本科毕业	2006-10-15
	021-55665566	中专毕业	2006-10-15

(a). 有一新记录(小王 13254748547 高中毕业 2007-05-06)请用 SQL 语句新增至表中

(b). 请用 sql 语句把张三的时间更新成为当前系统时间 (c). 请写出删除名为张四的全部记录

```
INSERT INTO user(name, tel, content, date) VALUES('小王', '13254748547', '高中毕业', '2007-05-06');
```

```
UPDATE user SET date = date_format(now(), '%Y-%m-%d') WHERE name = '张三'
```

```
DELETE FROM user WHERE name = '张四'
```

18. MySQL 自增类型(通常为表 ID 字段)必需将其设为() 字段整型，并设置为 AUTO_INCREMENT

19. 数据库中的事务是什么？

事务 (transaction) 是作为一个单元的一组有序的数据库操作。如果组中的所有操作都成功, 则认为事务成功, 即使只有一个操作失败, 事务也不成功。如果所有操作完成, 事务则提交, 其修改将作用于所有其他数据库进程。如果一个操作失败, 则事务将回滚, 该事务所有操作的影响都将取消。ACID 四大特性, 原子性、隔离性、一致性、持久性。

20. What's the difference between `mysqli_fetch_row()` and `mysqli_fetch_array()`? (Yahoo)

`mysqli_fetch_row()` 从和指定的结果标识关联的结果集中取得一行数据并作为数组返回。每个结果的列储存在一个数组的单元中, 偏移量从 0 开始。

`mysqli_fetch_array()` 是 `mysqli_fetch_row()` 的扩展版本。除了将数据以数字索引方式储存在数组中之外, 还可以将数据作为关联索引储存, 用字段名作为键名。

`mysqli_fetch_array()` 中可选的第二个参数 `result_type` 是一个常量, 可以接受以下值: `MYSQL_ASSOC`, `MYSQL_NUM` 和 `MYSQL_BOTH`。其默认值是 `MYSQL_BOTH`。

如果用了 `MYSQL_BOTH`, 将得到一个同时包含关联和数字索引的数组。用 `MYSQL_ASSOC` 只得到关联索引 (如同 `mysqli_fetch_assoc()` 那样), 用 `MYSQL_NUM` 只得到数字索引 (如同 `mysqli_fetch_row()` 那样)。

21. 请写出 php 连 mysql 连接中, 获取下一个自增长 id 值的方法, 可以写多个 (酷讯)

方法一, 使用 `show table status`, 然后获取 `auto_increment` 的值
方法二, 使用 `select max(id) + 1 from table`
方法三, 如果是刚插入记录, 可以使用 `last_insert_id() + 1` 获得

22. 从表 login 中选出 name 字段包含 admin 的前 10 条结果所有信息的 sql 语句 (酷讯)

```
SELECT * FROM login
WHERE name LIKE '%admin%'
LIMIT 10;
```

23. 表中有 A B C 三列, 用 SQL 语句实现: 当 A 列大于 B 列时选择 A 列, 否则选择 B 列, 当 B 列大于 C 列时选择 B 列否则选择 C 列。

使用 case 语句，如下：

```
SELECT CASE WHEN A > B THEN A ELSE B END,  
CASE WHEN B > C THEN B ELSE C END  
FROM table
```

24. 写出三种以上 MySQL 数据库存储引擎的名称（提示：不区分大小写）

MyISAM、InnoDB、BDB (BerkeleyDB)、Merge、Memory (Heap)、Example、Federated、Archive、CSV、Blackhole、MaxDB 等等十几个引擎。

25. 请简述数据库设计的范式及应用。

一般第 3 范式就足以，用于表结构的优化，这样做既可以避免应用程序过于复杂同时也避免了 SQL 语句过于庞大所造成系统效率低下。

第一范式：若关系模式 R 的每一个属性是不可再分解的，且有主键，则属于第一范式。

第二范式：若 R 属于第一范式，且所有的非主键属性都完全函数依赖于主键属性，则满足第二范式。

第三范式：若 R 属于第二范式，且所有的非主键属性没有一个是传递函数依赖于候选主键属性，则满足第三范式。

在实际使用中，可以根据需求适当的逆范式。

26. 取得最新一次添加记录（假设 id 为主键，并且是自增类型）所产生的 id 的函数是什么？

`mysqli_insert_id()`；如果上一查询没有产生 `AUTO_INCREMENT` 的值，则 `mysqli_insert_id()` 返回 0。

27. php 访问数据库有哪几步？

主要有以下几个步骤：

- 1) 连接数据库服务器：`mysqli_connect('host','user','password');`
- 2) 判断是不是连接成功了，是不是有错误 `mysqli_errno, mysqli_error.`
- 3) 选择数据库：`mysqli_select_db(数据库名);`
- 4) 设置从数据库提取数据的字符集：`mysqli_set_charset;`
- 5) 准备 sql 语句（字符串变量）`$sql = “ ” ;`
- 6) 发送 sql 语句 `mysqli_query`

- 7) 判断结果（如果是查询类的语句需要显示输出结果） `Mysqli_affected_rows`
- 8) 清空资源关闭数据库连接 `mysqli_close`.

29. 在平常 mysql 优化方面，最基本的也是最重要的优化是（）。（奇矩互动）

查询优化

30. 解释 MySQL 外连接、内连接与自连接的区别（小米）

先说什么是交叉连接，交叉连接又叫笛卡尔积，它是指不使用任何条件，直接将一个表的所有记录和另一个表中的所有记录一一匹配。内连接则是只有条件的交叉连接，根据某个条件筛选出符合条件的记录，不符合条件的记录不会出现在结果集中，即内连接只连接匹配的行。

而外连接其结果集中不仅包含符合连接条件的行，而且还会包括左表、右表或两个表中的所有数据行，这三种情况依次称之为左外连接，右外连接，和全外连接。

左外连接，也称左连接，左表为主表，左表中的所有记录都会出现在结果集中，对于那些在右表中并没有匹配的记录，仍然要显示，右边对应的那些字段值以 NULL 来填充。右外连接，也称右连接，右表为主表，右表中的所有记录都会出现在结果集中。左连接和右连接可以互换，MySQL 目前还不支持全外连接。

31. 两条查询语句，一条是 `select * from table1 where id > 10 and id < 100`，一条是 `select * from table2 where id > 200 and id < 500`，请尝试只写一条 SQL 语句，完成相应的查询任务。（鑫众人云）

使用 `union` 语句，但有一个前提，即 `table1` 和 `table2` 的结构相同

```
SELECT * FROM table1 WHERE id > 10 AND id < 100
```

```
UNION ALL
```

```
SELECT * FROM table2 WHERE id > 200 AND id < 500
```

32. 写出你所知道的数据库。（亿邮）

MySQL, SQL Server, Oracle, Sybase, informix, DB2 等

33. 用户互为好友的 SNS 存储结构怎么设计。（亿邮）

首先是有用户表，如下：

```
CREATE TABLE user( id INT UNSIGNED NOT NULL PRIMARY KEY
    AUTO_INCREMENT, username VARCHAR(30) NOT NULL DEFAULT
    " COMMENT '用户名', email VARCHAR(50) NOT NULL DEFAULT "
    COMMENT '邮箱', password CHAR(32) NOT NULL DEFAULT "
    COMMENT '密码'
```

```
)engine=MyISAM charset=utf8 comment='用户表';
```

其次是用户间的关系，如下：

```
CREATE TABLE relation( rel_id INT UNSIGNED NOT NULL PRIMARY KEY
    AUTO_INCREMENT, fuid INT UNSIGNED NOT NULL COMMENT '关注人的 id',
    suid INT UNSIGNED NOT NULL COMMENT '被关注人的 id', relation_type
    ENUM('S','D') NOT NULL DEFAULT 'S' COMMENT '关系，s 为关注，
    d 表示为好友'
```

```
)engine=MyISAM charset=utf8 comment='用户关系表';
```

36. 假设现在有一个数据库服务器，服务器地址为 **192.168.0.110**，用户名为 **root** 密码为 **password** 请使用 **PHP** 编写一个面向过程化的连接该数据库的脚本代码（亿邮）

```
$conn = mysqli_connect('192.168.0.110','root','password') or die('数据库连接失败');
```

37. 简述在 MySQL 数据库中 MyISAM 和 InnoDB 的区别（亿邮）

区别于其他数据库的最重要的特点就是其插件式的表存储引擎。切记：存储引擎是基于表的，而不是数据库。

InnoDB 与 MyISAM 的区别：

InnoDB 存储引擎，主要面向 OLTP(Online Transaction Processing, 在线事务处理)方面的应用，是第一个完整支持 ACID 事务的存储引擎(BDB 第一个支持事务的存储引擎，已经停止开发)。

特点：

行锁设计、支持外键；

支持类似于 Oracle 风格的一致性非锁定读(即：默认情况下读取操作不会产生锁)；

InnoDB 将数据放在一个逻辑的表空间中，由 InnoDB 自身进行管理。从 MySQL4.1 版本开始，可以将每个 InnoDB 存储引擎的表单独存放到一个独立的 ibd 文件中；

InnoDB 通过使用 MVCC(多版本并发控制：读不会阻塞写，写也不会阻塞读)来

获得高并发性，并且实现了 SQL 标准的 4 种隔离级别(默认为 REPEATABLE 级别)；

InnoDB 还提供了插入缓冲(insert buffer)、二次写(double write)、自适应哈希索引(adaptive hash index)、预读(read ahead)等高性能和高可用的功能；

InnoDB 采用了聚集(clustered)的方式来存储表中的数据，每张表的存储都按主键的顺序存放(如果没有显式的在建表时指定主键，InnoDB 会为每一行生成一个 6 字节的 ROWID，并以此作为主键)；

InnoDB 表会有三个隐藏字段：除了上面提到了 6 字节的 DB_ROW_ID 外，还有 6 字节的 DB_TX_ID(事务 ID)和 7 字节的 DB_ROLL_PTR(指向对应回滚段的地址)。这个可以通过 innodb monitor 看到；

MyISAM 存储引擎是 MySQL 官方提供的存储引擎，主要面向 OLAP(Online Analytical Processing, 在线分析处理)方面的应用。

特点：

不支持事务，支持表所和全文索引。操作速度快；

MyISAM 存储引擎表由 MYD 和 MYI 组成，MYD 用来存放数据文件，MYI 用来存放索引文件。MySQL 数据库只缓存其索引文件，数据文件的缓存交给操作系统本身来完成；

MySQL5.0 版本开始，MyISAM 默认支持 256T 的单表数据；

38. 现在有下面一个查询语句 `select * from tablename where id=2 and password='abc'` 如何判断它是现在是最优的。(亿邮)

可以使用 `explain select * from tablename where id=2 and password='abc'` 来分析其执行情况。

39. 请问如何在 Mysql 操作中如何写入 utf8 格式数据 (亿邮)

首先确保数据库中的表是基于 utf8 编码的，其次 php 文件是 utf8 编码，在执行 mysql 操作之前，执行 `mysql_query('set names utf8')` 操作，对于要操作的文本如果是 utf8 编码，则可以直接操作，如果是其它编码，则可以使用 `iconv` 函数将其转化为 utf8 编码，然后写入。

40. mysql 中 varchar 的最大长度是多少？用什么类型的字段存储大文本？date 和 datetime 和 timestamp 什么区别？怎么看数据库中有哪些 sql 正在执行？ (卓望)

varchar 的最大有效长度由最大行大小和使用的字符集确定。整体最大长度是 65532 字节。在 varchar 存字符串的时候，第一个字节是空的，不存任何的数据，然后还需要两个字节来存放字符串的长度。所以有效长度就是 $65535 - 1 - 2 = 65532$ 。

由字符集来确定，字符集分单字节和多字节，如果是单字节，如 latin1，则最多可以存放 65532 个字符，如果是多字节，如 GBK 则可以存放 32766 个字符，UTF8 则可以存放 21844 个字符。

存储大文本可以使用 text 类型。

date 表示日期，其范围为 1000-01-01 ~ 9999-12-31

datetime 表示日期时间，其范围为 1000-01-01 00:00:00 ~ 9999-12-31 23:59:59

timestamp 是 unix 时间戳的日期时间表示方式，其范围较小为 1970-01-01 00:00:00~2038-01-19 03:14:07，timestamp 具备自动初始化和自动更新功能。

查看数据库中正在执行的 sql 语句可以使用日志，也可以使用 show processlist 命令。

41. 现在有一个 mysql 数据库表 visits 记录用户访问情况，

表结构如下： visits(

```
id int unsigned auto_increment,  
user_id int unsigned comment '本次访问页面数',  
visit_time timestamp comment '本次访问开始时间',  
primary key(id)
```

);

用户每访问过一次网站(从进入到离开),会增加一条记录。记录用户的 ID(user_id),以及访问的页面总数。比如:

1, 208, 2, //208 这个用户访问 2 个页面

2, 2073, 3,

3, 208, 1, //208 用户访问了 1 个页面

(1). 请写一个 SQL 语句挑出你是累计访问页面数最多的 10 个用户(user_id)和对应的访问页面数。

(2). 请写一个 SQL 语句，输出累计访问页面数分别等于 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 的唯一用户的数量, 如果某个数量对应的用户数为 0, 可以不输出。(嘀嗒团) 从题目的描述来看，表结构貌似有些问题，user_id 应为用户 ID，而不是访问页面数，

增加一个字段 pages 表示访问页面数。查询访问页面数最多的 10 个用户的查询语句如下：

```
SELECT user_id, sum(pages) as total  
FROM visits  
GROUP BY user_id  
ORDER BY total DESC LIMIT 10;
```


输出累计访问页面数分别等于 1,2,3,4,5,6,7,8,9,10 的唯一用户的数量的查询语句如下:

```
SELECT total, count(user_id)
FROM (
SELECT user_id, sum(pages) AS total
FROM visits
GROUP BY user_id) AS temp
WHERE total IN (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
GROUP BY total ;
```

42. 简述存储过程的适用情况当需要处理复杂的查询和运算时， 可以使用存储过程。

从应用分层的原则，大量使用存储过程导致业务逻辑分散在 DB 和应用服务器层，不利于维护和更新。总体来说，存储程序可以用，但要慎重，最好只用来维护，不用于业务逻辑和支撑高并发高性能的东西。

43. 对于关系型数据库而言，索引是相当重要的概念，请回答有关索引的几个问题：

a)、索引的目的是什么？

1. 快速访问数据表中的特定信息，提高检索速度
2. 创建唯一性索引，保证数据库表中每一行数据的唯一性。
3. 加速表和表之间的连接
4. 使用分组和排序子句进行数据检索时，可以显著减少查询中分组和排序的时间

b)、索引对数据库系统的负面影响是什么？

负面影响：创建索引和维护索引需要耗费时间，这个时间随着数据量的增加而增加；索引需要占用物理空间，不光是表需要占用数据空间，每个索引也需要占用物理空间；当对表进行增、删、改、的时候索引也要动态维护，这样就降低了数据的维护速度。

c)、为数据表建立索引的原则有哪些？

- 1、在最频繁使用的、用以缩小查询范围的字段上建立索引。
- 2、在频繁使用的、需要排序的字段上建立索引

d)、 什么情况下不宜建立索引？

- 1、对于查询中很少涉及的列或者重复值比较多的列，不宜建立索引。
- 2、对于一些特殊的数据类型，不宜建立索引，比如文本字段（text）等。

44、 web 应用中，数据库的读取频率远高于写入频率，如何优化 Mysql 而应对此种场景？

使用 memcache 缓存技术，将动态数据缓存到文件，访问动态页面时直接调用缓存文件，而不必重新访问数据库，这样就减少了查询数据库的次数。

如果网站的访问量很大，可以把数据库读写服务器分开，使用多态服务器去处理数据库查询，使用较少的服务器处理数据库的写入和修改。

45、 数据库设计时，常用到性能瓶颈有哪些，常用的解决方案？

瓶颈主要有：

- 1、磁盘检索 优化方法是：将数据分布在多个磁盘上
- 2、磁盘读/写 优化方法是：从多个磁盘并行读写。
- 3、CPU 周期 优化方法是： 扩充内存
- 4、内存带宽

46、请举例说明在你的开发中用什么方法来加快页面的加载速度？

要用到服务器资源时才打开，及时关闭服务器资源，数据库添加索引，页面可生成静态，图片等大文件单独服务器。使用代码优化工具。

47、如何设计或配置 Mysql，才能达到高效使用的目的？

- 1、数据库设计方面，设计结构良好的数据库，允许部分数据冗余。选取最适用的字段属性，尽可能把字段设置为 NOTNULL, 这样在查询的时候，数据库不用去比较 NULL 值。
- 2、系统架构设计方面，表散列，把海量数据散列到几个不同的表里面，集群，数据库查询和写入分开。

写高效的 sql 语句，以提高效率。

使用连接 (join) 来代替子查询。

使用联合 (union) 来代替手动创建的临时表。

所得皆必须，只从数据库取必须的数据。

必要的时候用不同的存储引擎，比如 Innodb 可以减少死锁，HEAP 可以提高一个数量级的查询速度。

使用事务

使用外键

使用索引。

48、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？

我们可以根据用户的活跃程度，把活跃的用户提取出来放到另外一张表里面，每次活跃的用户登录的时候就直接到活跃用户表中查询，这样就提高了数据库的查询速度。

49、了解 XSS 攻击吗？如何防止？

XSS 是跨站脚本攻击，首先是利用跨站脚本漏洞以一个特权模式去执行攻击者构造的脚本，然后利用不安全的 ActiveX 控件执行恶意的行为。

使用 `htmlspecialchars()` 函数对提交的内容进行过滤，使字符串里面的特殊符号实体化。

50、SQL 注入漏洞产生的原因？如何防止？

SQL 注入产生的原因：程序开发过程中不注意规范书写 sql 语句和对特殊字符进行过滤，导致客户端可以通过全局变量 POST 和 GET 提交一些 sql 语句正常执行。

防止 SQL 注入的方式：

- 1、开启配置文件中的 `magic_quotes_gpc` 和 `magic_quotes_runtime` 设置
- 2、执行 sql 语句时使用 `addslashes` 进行 sql 语句转换
- 3、Sql 语句书写尽量不要省略双引号和单引号。
- 4、过滤掉 sql 语句中的一些关键词：`update`、`insert`、`delete`、`select`、`*`。
- 5、提高数据库表和字段的命名技巧，对一些重要的字段根据程序的特点命名，取不易被猜到的。
- 6、Php 配置文件中设置 `register_globals` 为 `off`，关闭全局变量注册
- 7、控制错误信息，不要在浏览器上输出错误信息，将错误信息写到日志文件中。

51、PHP 如何抛出和接收错误？

使用 `try...catch`，异常的代码放在 `try` 代码块内，如果没有出发异常，则代码继续执行，如果异常被触发，就会抛出一个异常。`Catch` 代码块捕获异常，并创建一个包含异常信息的对象。`$e->getMessage()`，输出异常的错误信息。

52、PHP 网站的主要攻击方式有哪些？

- 1、命令注入(Command Injection)
- 2、eval 注入(Eval Injection)
- 3、客户端脚本攻击(Script Insertion)
- 4、跨网站脚本攻击(Cross Site Scripting, XSS)
- 5、SQL 注入攻击(SQL injection)
- 6、跨网站请求伪造攻击(Cross Site Request

Forgeries, CSRF)

- 7、Session 会话劫持(Session Hijacking)
- 8、Session 固定攻击(Session Fixation)
- 9、HTTP 响应拆分攻击(HTTP Response Splitting)
- 10、文件上传漏洞(File Upload Attack)
- 11、目录穿越漏洞(Directory Traversal)
- 12、远程文件包含攻击(Remote Inclusion)
- 13、动态函数注入攻击(Dynamic Variable Evaluation)
- 14、URL 攻击(URL attack)
- 15、表单提交欺骗攻击(Spoofed Form Submissions)
- 16、HTTP 请求欺骗攻击(Spoofed HTTP Requests)

53、你常用到的 mysql 命令？

```
Show databases
Show tables
Insert into 表名() values()
Update 表名 set 字段=值 where ...
Delete from 表名 where ...
Select * from 表名 where 条件 order by ...
Desc/asc limit ... Group by ... Having ...
```

54、mysql 常用命令的应用？

进入 mysql 管理命令行的命令？

```
Mysql -uroot -p 回车 密码
```

show databases; 这个命令的作用？

显示当前 mysql 服务器中有哪些数据库

show create database mysql; 这个命令的作用？

显示创建数据库的 sql 语句

show create table user; 这个命令的作用？

显示创建表的 sql 语句

desc user; 这个命令的作用？

查询 user 表的结构

explain select * from user; 这个命令的作用？

获取 select 相关信息

`show processlist;` 这个命令的作用?

显示哪些线程正在运行

`SHOW VARIABLES;` 这个命令的作用?

显示系统变量和值

`SHOW VARIABLES like ' %conn% '` 这个命令的作用?

显示系统变量名包含 conn 的值

`LEFT JOIN` 写一个 SQL 语句?

```
SELECT A.id,A.class FROM A LEFT JOIN B ON A.cid=B.id
```

55、请列出 5 个常用的 PHP 操作 MySQL 的函数

- 1、`Mysqli_connect()`
- 2、`Mysqli_select_db()`
- 3、`Mysqli_query()`
- 4、`Mysqli_fetch_row()`
- 5、`Mysqli_close()`

56、优化 Mysql 数据库的方法 ?

- 1、选取最适用的字段属性, 尽可能减少定义字段长度, 尽量把字段设置 NOT NULL, 例如' 省份, 性别', 最好设置为 ENUM
- 2、使用连接 (JOIN) 来代替子查询:
- 3、使用联合 (UNION) 来代替手动创建的临时表
- 4、事务处理:
- 5、锁定表, 优化事务处理:
- 6、使用外键, 优化锁定表
- 7、建立索引:
- 8、优化查询语句

杂项

1. memcache 的默认端口是啥? Value 的最大长度是什么? 默认的缓存有效期是多久?

答案: 端口: 11211, value 最大是 1M, 有效期是 30 天,

2. redis 的最大的 value 值是多少? 存储有序和无序集合用哪种数据类型? 哪个命令发起事务? 哪两种持久化方式?

答案: value 最大 512M, 有序用 list 和 sorted set, 无序用 set, 发起事务用 multi; 持久化方式为内存快照和日志追加。

4、说说你对代码管理的了解?

通常一个项目是由一个团队去开发, 每个人将自己写好的代码提交到版本服务器, 由项目负责人按照版本进行管理, 方便版本的控制, 提高开发效率, 保证需要时可以回到旧版本。

5、windows 平台,, Apache Http Server 启动失败,, 排错思路是什么?

检查 apache 使用的 80 端口是否被占用, 如果被占用, 先停止占用 80 端口的服务, 然后启动 apache 服务器。

6、视频常用的格式是什么, 流媒体服务器都有哪些?

视频常用的格式 : a、MPEG/.MPG/.DAT b、AVI c、RA/RM/RAM d、MOV e、其他本地播放的视频格式

常见的流媒体服务器有:

- 1)、Flash 流媒体服务器 [Red5](#)
- 2)、 Darwin Streaming Server
- 3)、 流媒体服务器 Open Streaming Server
- 4)、 流媒体服务器 Yass
- 5)、 开源流媒体平台 Stream-2-Stream
- 6) Flumotion 流媒体服务器 9 icecast10 WEB 自动点唱机 netjukebox
- 7)、 流媒体解决方案 [live555](#)
- 8)、 Helix DNA Platform
- 9)、 电视流媒体服务器 Tvheadend14 OpenAMF15 【商业】流媒体

服务器 xmoovStream

10)、 Java 流媒体服务器 JRoar

11)、 webcamFLV

12)、 【商业】流媒体服务器软件 Helix Server

13)、 Free UPnP Entertainment Service

14)、 流媒体服务器 Tunapie

15)、 pulpTunes

16)、 流媒体服务器 TivoServer

7、框架中什么是单一入口和多入口，单一入口的优缺点？

1、多入口就是通过访问不同的文件来完成用户请求。单一入口只 web 程序所有的请求都指向一个脚本文件的。

2、单一入口更容易控制权限，方便对 http 请求可以进行安全性检查。

缺点：URL 看起来不那么美观，特别是对搜索引擎来说不友好。

8、常用的 PHP 包管理你知道有哪些吗？说几个。

1)、 phar 2)、 composer.

9、谈谈你对 composer 的理解？

composer 是 PHP 的一个依赖管理工具。你可以在自己的项目中使用 composer 管理依赖库。Composer 可以为你解决一以下问题：如果你的项目依赖于若干个库，这些库还依赖于其他库，你可以在 composer.json 中声明依赖库，compose 会自动下载安装所需要的的库（包）需要安装。

Composer 的运行原理如下：

