## 一、HTML5BASIC

### Day01

1、HTML的页面结构?

难度等级：【简单】

**答案：**

**<!DOCTYPE HTML>**

**<html>**

**<head>**

**<title> New Document </title>**

**<meta charset="utf-8">**

**<style></style>**

**<script></script>**

**</head>**

**<body>**

**</body>**

**</html>**

2.html是什么语言？

难度等级：【简单】

**答案：**

**HTML是开发Web网页程序的一种语言，全称为：Hyper Text Markup Language，即超文本标 记语言，由HTML编写的文本最终是以 .html 或 .htm作为结尾的文件 , 并且由浏览器解释运行。**

3.前端页面有哪三层构成，分别是什么？作用是什么？

难度等级：【简单】

**答案：**

**分成：结构层、表示层、行为层。**

**结构层（structural layer）**

**由 HTML 或 XHTML之类的标记语言负责创建。标签，也就是那些出现在尖括号里的单词，对网 页内容的语义含义做出了描述，但这些标签不包含任何关于如何显示有关内容的信息。例如，P标签 表达了这样一种语义：“这是一个文本段。”**

**表示层（presentation layer）**

**由 CSS 负责创建。 CSS对“如何显示有关内容”的问题做出了回答。**

**行为层（behaviorlayer）**

**负责回答“内容应该如何对事件做出反应”这一问题。这是 Javascript 语言和 DOM主宰的领域。**

1. doctype的作用是什么？存在有什么意义？

难度等级：【简单】

**答案：**

**DOCTYPE描述了html文档的类型，对不同的DOCTYPE类型，浏览器会使用不同的方法来解析。**

### Day02

1.在html中的a标签中，实现在新窗口中打开链接的标签属性是：\_\_\_\_\_\_\_?

难度等级：【中等】

**答案：**

**target属性。当它的值为\_blank的时候就会实现在新窗口中打开，它的值还可以是：\_self（当前窗 口中打开），\_parent(父frame中打开），\_top（整个窗口中打开）。**

2、img的alt与title有何异同？

难度等级：【简单】

**答案：**

**title:该属性为设置该属性的元素提供建议性的信息。alt属性用来指定替换文字。**

### Day03

1. xhtml和html有什么区别？html语义化有何意义？

难度等级：【中等】

**答案：**

**HTML与XHTML之间的差别，粗略可以分为两大类比较：一个是功能上的差别，另外是书写习惯 的差别。关于功能上的差别，主要是XHTML可兼容各大浏览器、手机以及PDA，并且浏览器也能 快速正确地编译网页。因为XHTML的语法较为严谨，所以如果你是习惯松散结构的HTML编写者， 那需要特别注意XHTML的规则。**

**为了在没有CSS的情况下，页面也能呈现出很好地内容结构、代码结构:为了裸奔时好看；**

**用户体验：例如title、alt用于解释名词或解释图片信息、label标签的活用；**

**有利于**[**SEO**](http://baike.baidu.com/view/1047.htm)**：和搜索引擎建立良好沟通，有助于爬虫抓取更多的有效信息：**[**爬虫**](http://baike.baidu.com/view/998403.htm)**依赖于标签来确定上 下文和各个关键字的权重；**

**方便其他设备解析（如屏幕阅读器、盲人阅读器、移动设备）以意义的方式来渲染网页；**

**便于团队开发和维护，语义化更具可读性，是下一步吧网页的重要动向，遵循W3C标准的团队都遵 循这个标准，可以减少差异化。**

1. 什么是块级元素和行内元素，分别写三个例子？

难度等级：【简单】

**答案：**

**块级元素独占一行，可以设置宽和高。比如：div h1 ul**

**行内元素在一行内排列，且不可设置宽和高。比如：span a b**

3、input标签如何设置成按钮？

难度等级：【简单】

**答案：**

**<input type=”button” />**

4、table合并单元格的属性是什么?

难度等级：【简单】

**答案：**

**colspan合并列，rowspan合并行，取值都为数字。**

5什么是语义化的HTML?

难度等级：【简单】

**答案：**

**语义化的HTML就是写出的HTML代码，符合内容的结构化（内容语义化），选择合适的标签（代 码语义化），能够便于开发者阅读和写出更优雅的代码的同时让浏览器的爬虫和机器很好地解析。**

**1.语义化有利于SEO，有利于搜索引擎爬虫更好的理解我们的网页，从而获取更多的有效信息，提升 网页的权重。**

**2.在没有CSS的时候能够清晰的看出网页的结构，增强可读性。**

**3.便于团队开发和维护，语义化的HTML可以让开发者更容易的看明白，从而提高团队的效率和协 调能力。**

## 二、CSS3 BASIC

### Day01

1、CSS样式有哪几种？引入方式中link和@import的区别是？

难度等级：【中等】

**答案：**

**内联样式、内部样式表、外部样式表**

**本质上，这两种方式都是为了加载CSS文件，但还是存在着细微的差别。  
　　（1）老祖宗的差别。link属于XHTML标签，而@import完全是CSS提供的一种方式。 link标 签除了可以加载CSS外，还可以做很多其它的事情，比如定义RSS，定义rel连接属性等，@import 就只能加载CSS了。  
　　（2）加载顺序的差别。当一个页面被加载的时候（就是被浏览者浏览的时候），link引用的CSS会同 时被加载，而@import引用的CSS会等到页面全部被下载完再被加载。所以有时候浏览@import 加载CSS的页面时开始会没有样式（就是闪烁），网速慢的时候还挺明显。  
　 （3）** [**兼容性的差别。由于@import是CSS2.1提出的所以老的浏览器不支持，@import只有在IE5**](mailto:兼容性的差别。由于@import是CSS2.1提出的所以老的浏览器不支持，@import只有在IE5) **以上的才能识别，而link标签无此问题。  
　 （4） 使用dom控制样式时的差别。当使用javascript控制dom去改变样式的时候，只能使用link 标签，因为@import不是dom可以控制的。  
　　@import可以在css中再次引入其他样式表，比如可以创建一个主样式表，在主样式表中再引入其 他的样式表，如：　  
 main.css ———————- @import “sub1.css”; @import “sub2.css”; sub1.css ———————- p {color:red;} sub2.css ———————- .myclass {color:blue}　　这样更利于修改和扩展。  
　　大致就这几种差别了，其它的都一样，从上面的分析来看，还是使用link标签比较好。标准网页制作 加载CSS文件时，还应该选定要加载的媒体（media），比如screen，print，或者全部all等。**

2、如何为盒子设置圆角和阴影效果？

难度等级：【简单】

**答案：**

**使用border-radius属性为盒子设置圆角：**

**如：border-radius:50%;**

**border-radius:10px 20px 30px 40px; border-top-left-radius:5px;**

**使用box-shadow属性为盒子设置阴影效果：box-shadow: h-shadow v-shadow blur spread color inset;**

**如：box-shadow:5px 5px 10px 20px red inset;**

3、如何让元素居中对齐？

难度等级：【简单】

**答案：text-align:center;**

4、px和em的区别?

难度等级：【简单】

**答案：**

**px和em都是长度单位，区别是，px的值是固定的，指定是多少就是多少，计算比较容易。**

**em得值不是固定的，并且em会继承父级元素的字体大小。**

5、内联样式表的权值为\_\_\_\_\_；ID 选择器的权值为\_\_\_\_；

Class 类选择器的权值为\_\_\_\_\_；HTML 标签选择器的权值为\_\_\_\_**。**

难度等级：【简单】

**答案：**

**1000 100 10 1**

6、浏览器的前缀分别是什么？

难度等级：【中等】

**答案：火狐 -moz- 谷歌 --webkit- Opera -0- IE -ms-**

7、1em等于多少px?

难度等级：【中等】

**答案：**

**em是相对于当前浏览器默认字体大小来说的，任意浏览器的默认字体高都是16px。所有未经调整 的浏览器都符合：1em=16px。**

### Day02

1. rgba()和opacity的透明效果有什么不同？

**难度等级：【中等】**

**答案：**

**rgba()和opacity都能实现透明效果，但最大的不同是opacity作用于元素，以及元素内的所有内 容的透明度，**

**而rgba()只作用于元素的颜色或其背景色。（设置rgba透明的元素的子元素不会继承透明效果！）**

1. 什么是盒子模型？

**难度等级：【中等】**

**答案：**

**在网页中，一个元素占有空间的大小由几个部分构成，其中包括元素的内容（content），元素的内 边距（padding），元素的边框（border），元素的外边距（margin）四个部分。**

**这四个部分占有的空间中，有的部分可以显示相应的内容，而有的部分只用来分隔相邻的区域或区域。 4个部分一起构成了css中元素的盒模型。**

### Day03

1、哪些css属性可以继承？

难度等级：【简单】

**答案：**

**不可继承的：display、margin、border、padding、background、height、min-height、 max-height、width、min-width、max-width、overflow、position、left、right、top、bottom、 z-index、float、clear、table-layout、vertical-align、page-break-after、page-bread-before 和unicode-bidi。**

**所有元素可继承：visibility和cursor。**

**内联元素可继承：letter-spacing、word-spacing、white-space、line-height、color、font、 font-family、font-size、font-style、font-variant、font-weight、text-decoration、 text-transform、direction。**

**终端块状元素可继承：text-indent和text-align。**

**列表元素可继承：list-style、list-style-type、list-style-position、list-style-image。**

2、火狐下怎么撑开容器高度？

难度等级：【简单】

**答案：**

**标准浏览器中固定高度值的容器是不会象IE6里那样被撑开的，但如果又想固定高度，又想能被撑开 需要怎样设置呢？同时为了照顾不认识min-height的IE6 ，代码可以这样定义：**

**div{**

**height:auto!important;**

**height:200px;**

**width:300px;**

**min-height:200px;**

**}**

3、如何垂直居中一个浮动元素？

难度等级：【简单】

**答案：**

**方法一：已知元素的高宽**

**#div1{**

**background-color:#6699FF;**

**width:200px;height:200px;**

**position: absolute; //父元素需要相对定位**

**top: 50%; left: 50%;**

**margin-top:-100px ; //二分之一的height，width**

**margin-left: -100px;**

**}**

**//方法二:未知元素的高宽**

**#div1{**

**width: 200px;**

**height: 200px; //宽高就可以随意设置了，无需在设置其他的代码，永运句中显示**

**background-color: #6699FF;**

**margin:auto;**

**position: absolute; //父元素需要相对定位**

**left: 0;top: 0;**

**right: 0;**

**bottom: 0;**

**}**

4、如何居中div，如何居中一个浮动元素？

难度等级：【中等】

答案：

（1）给div设置一个宽度，然后添加margin:0 auto属性

div{

width:200px;

margin:0 auto;

}

（2）居中一个浮动元素

确定容器的宽高，如宽500、高 300的层，设置层的外边距：

.div {

Width:500px ; height:300px;//高度可以不设

Margin: -150px 0 0 -250px;

position:relative;相对定位

background-color:pink;//方便看效果

left:50%;

top:50%;

}

### Day04

1、css实现左边固定，右边自适应容器宽度的两栏布局

**难度等级：【简单】**

**答案：**

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8">**

**<title>Document</title>**

**<style>**

**div{**

**height: 200px;**

**}**

**#d1{**

**width: 200px;**

**background-color: red;**

**float: left;**

**}**

**#d2{**

**background-color: green;**

**margin-left: 200px;}**

**</style>**

**</head>**

**<body>**

**<div id="d1"></div>**

**<div id="d2"></div>**

**</body>**

**</html>**

2、尽你最大能力，想出3种隐藏元素的样式？

**难度等级：【简单】**

**答案：**

**若使用display:none来隐藏元素的话，修改为：display:block;**

**若使用visibility:hidden来隐藏元素的话，修改为：visbility:visible;**

**若使用overflow:hidden隐藏溢出内容的话，修改为：overflow:visible;**

3、display:none与visibility:hidden的区别是什么？

**难度等级：【中等】**

**答案：**

**display : 隐藏对应的元素并且不占位置**

**visibility: 隐藏对应的元素占位置。**

4、如何让一个div布满整个屏幕。

**难度等级：【简单】**

**答案：**

**<!DOCTYPE HTML>**

**<html>**

**<head>**

**<style>**

**html,body{ margin:0px; height:100%; }**

**#bg{ width:100%; height:100%; MARGIN: 0px auto; background-color:blue; }**

**</style>**

**</head>**

**<body bgcolor="red">**

**<div id="bg"> </div>**

**</body>**

**</html>**

5、什么要初始化CSS样式？

**难度等级：【简单】**

**答案：**

**因为浏览器的兼容问题，不同浏览器对有些标签的默认值是不同的，如果没对CSS初始化往往会出 现浏览器之间的页面显示差异。**

6、列出display的值，说明他们的作用?

**难度等级：【中等】**

**答案：**

**常用的有：**

**none 缺省值，像行内元素类型一样显示，彻底的隐藏元素包括位置也不再占据；**

**block 像块级元素一样显示；**

**inline-block 像行内元素一样显示，但其内容像块级元素一样可以设置宽高；**

**table 像表格一样显示。**

7、position的值中，relative和absolute定位原点是？

**难度等级：【中等】**

**答案：**

**absolute 父元素设置了定位方式的，基于父元素定位；没有设置的，基于body定位；**

**relative 基于本身进行定位。**

8、如何设计一个满屏“品”字布局？

**难度等级：【中等】**

**答案：**

**div{**

**border:1px solid red;**

**width:10%;**

**height:100px;**

**}**

**#d1{**

**margin:0 auto;**

**}**

**#d2{**

**margin-left:40%;**

**}**

**#d2,#d3{**

**float:left;**

**}**

**<div id="d1"></div>**

**<div id="d2"></div>**

**<div id="d3"></div>**

9、为以下页面提供出相应的布局方式（方式不限，可提供多种）

**难度等级：【中等】**

****

**答案：**

**1、选用默认布局，先设置上方一个div宽百分百，高固定；然后下方一分为二，分别浮动左右，然 后设置每个小div的高，这样就会依次上下排列显示。**

**2、选用弹性布局。**

**3、使用bootstrap框架的栅格布局。**

## 三、CSS3 CORE

### Day01

1. 读代码，回答？处问题。请写出其他4种方法实现如下效果

**<style>  
 .d1{  
 font-size:0; ？为什么要在p父元素中设置字体大小为0  
 }  
 p{  
 display: inline-block;  
 width: 50%;  
 height: 20px;  
 font-size: 14px;  
 background: #f00;  
 box-sizing: border-box;  
 }  
</style>  
<body>  
 <div class="d1">  
 <p>123</p>  
 <p>456</p>  
 </div>  
</body>**

**难度等级：【中等】**

**答案：**

**因为行内块元素之间有间隙，给父元素设置字体大小为0可以清除间隙，让两个P元素在同一行显示。**

**1.浮动法**

**<style>  
 \*{margin:0;padding:0;}  
 .box{width:600px;height:400px;border:1px solid red;}  
 .one{width:100px;height:100px;border:1px solid #000;float:left;}  
</style>**

**<div class="box">  
 <div class="one">第一个</div>  
 <div class="one">第二个</div>  
 <div class="one">第三个</div>  
</div>**

1. **转行块元素**
2. **<style>  
    \*{margin:0;padding:0;}  
    .box{width:600px;height:400px;border:1px solid red;}  
    .one{width:100px;height:100px;border:1px solid #000;display:inline-block;}  
   </style>**

**外间距解决办法.box{font-size:0;}  
把字体再改回来.one{font-size:14px;}**

**3.定位**

**4.弹性布局display:flex**

**<style>  
 \*{margin:0;padding:0;}  
 .box{width:600px;height:400px;**

**border:1px solid red;display:flex;flex-direction:row;}  
 .one{width:100px;height:100px;border:1px solid #000;flex:1;}  
</style>**

**5.转单元格display:table-cell**

**<style>  
 \*{margin:0;padding:0;}  
 .box{width:600px;height:400px;border:1px solid red;}  
 .one{width:100px;height:100px;border:1px solid #000;display:table-cell;}  
</style>**

1. 清除浮动有哪几种方式?（至少写出5种）

**难度等级：【简单】**

**答案：**

**1. 父级div定义height**

**2. 结尾处加空div标签 clear:both**

**3. 父级div定义伪类：after和zoom**

**Div:after{**

**Content:””;**

**Display:table;**

**Clear:both;**

**}**

**4. 父级div定义overflow:hidden**

**5. 父级div定义overflow:auto**

**6. 父级div也浮动，需要定义宽度**

**7. 父级div定义display:table**

**8. 结尾处加br标签clear:both**

1. 什么是Css Hack？ie6,7,8的hack分别是什么？

**难度等级：【中等】**

**答案：**

**浏览器的类型及版本的不同会造成CSS效果不尽相同，因此需要针对不同的浏览器编写不同的CSS，来实现浏览器兼容，也可以为不同版本的浏览器定制编写不同的CSS效果。针对不同的浏览器写不同的CSS code的过程，就是CSS hack。**

**示例如下：**

**#test {**

**width:300px;**

**height:300px;**

**background-color:blue; /\*firefox\*/**

**background-color:red\9; /\*all ie\*/**

**background-color:yellow; /\*ie8\*/**

**+background-color:pink; /\*ie7\*/**

**\_background-color:orange; /\*ie6\*/ }**

**:root #test { background-color:purple\9; } /\*ie9\*/**

**@media all and (min-width:0px)**

**{ #test {background-color:black;} } /\*opera\*/**

**@media screen and (-webkit-min-device-pixel-ratio:0)**

**{ #test {background-color:gray;} } /\*chrome and safari\*/**

1. 浏览器的内核分别是什么?

**难度等级：【中等】**

**答案：**

**IE: trident内核**

**Firefox：gecko内核**

**Safari：webkit内核**

**Opera：以前是presto内核，Opera现已改用Google Chrome的Blink内核**

**Chrome：Blink(基于webkit，Google与Opera Software共同开发)**

1. 描述css reset的作用和用途。

**难度等级：【简单】**

**答案：**

**reset重置浏览器的css默认属性，因为浏览器的品种不同，样式也不同，所以先重置让他们统一，然后再进行相关的设置。**

### Day02

1、如何让元素顺时针旋转60度？

**难度等级：【简单】**

**答案：**

**transform: rotate(60deg) ;**

2、哪些css属性可以继承？

**难度等级：【中等】**

**答案：**

**不可继承的：display、margin、border、padding、background、height、min-height、 max-height、width、min-width、max-width、overflow、position、left、right、top、bottom、 z-index、float、clear、table-layout、vertical-align、page-break-after、page-bread-before 和unicode-bidi。**

**所有元素可继承：visibility和cursor。**

**内联元素可继承：letter-spacing、word-spacing、white-space、line-height、color、font、 font-family、font-size、font-style、font-variant、font-weight、text-decoration、 text-transform、direction。**

**终端块状元素可继承：text-indent和text-align。**

**列表元素可继承：list-style、list-style-type、list-style-position、list-style-image。**

### Day03

1. CSS都有哪些选择器？

**难度等级：【简单】**

**答案：**

**元素选择器**

**id选择器（用DOM的ID申明）**

**类选择器（用一个样式类名申明）**

**属性选择器（用DOM的属性申明，属于CSS2，IE6不支持，不常用，不知道就算了）**

**除了前3种基本选择器，还有一些扩展选择器，包括**

**后代选择器（利用空格间隔，比如div .a{ }）**

**子代选择器：（div > p选择div下的第一个子元素p标签）**

**群组选择器（利用逗号间隔，比如p,div,#a{ }）**

## 四、JSBASIC

### Day01

1. JavaScript有哪些数据类型？分别是什么？

**难度等级：【简单】**

**答案：**

**基本数据类型有：**

**字符串 String**

**数字 Number**

**布尔Boolean**

**复合数据类型有：**

**数组 Array**

**对象 Object**

**特殊数据类型：**

**Null 空对象**

**Undefined 未定义**

1. var a,b,c=10;console.log(a);console.log(b);console.log(c);a，b，c分别输出的是什么？

**难度等级：【简单】**

**答案：Js不支持一次设置多个变量，所以会输出undefined undefined 10。**

1. 页面加载过程。

**难度等级：【简单】**

**答案：**

**Js不支持一次设置多个变量，所以会输出undefined undefined 10。**

**难度等级：【中等】**

**答案：**

**输入网址，浏览器向服务器发出请求，服务器返回html文件**

**浏览器载入html代码，发现 head 标签中有一个 link 引用了外部css文件**

**浏览器又发起css文件请求，服务器返回这个css文件**

**浏览器继续载入 body 部分的代码，而且css文件已经拿到，可以开始渲染页面了**

**浏览器在代码中发现一个 img 标签引用了张图片，就开始向服务器发出请求，此时浏览器不会等到 图片下载完毕，而是继续渲染后面的代码**

**服务器返回图片文件，由于图片占用了一定的位置，影响了后面的排版布局，因此浏览器需要返回来 重新渲染这部分代码**

**浏览器发现了 javascript 标签，赶快运行发现需要引用外部js文件，浏览器发起请求，服务器返回 浏览器开始解析js代码，发现js中需要浏览器隐藏掉代码中某个div块，突然接到这个命令浏览器 不得不重新渲染这部分代码**

**要是在页面上用户操作让浏览器更改页面布局或颜色搭配，浏览器就得重新请求相应的css文件**

**浏览器只得召集在座的 div、span、ul、h1告诉它们按照刚刚的流程再来一次。**

### Day02

1、以下代码输出结果是什么？

**难度等级：【中等】**

**var a=3,b=4;**

**console.log(a++ + ++b + ++a);**

**console.log(b++ - ++a + b--);**

**答案：15 3**

### Day03

1. 下面代码哪个结果会先弹出来？会弹出什么结果？

**难度等级：【简单】**

**var x=1;**

**var y=1;**

**mothed(8);//①**

**alert(x+y);//②**

**function mothed(y){**

**x+=3;**

**y+=3;**

**alert(x+y);**

**}**

**答案：①15 ② 5**

2、将数字5、9、3按从大到小的顺序排序（代码！**）**

**难度等级：【简单】**

**答案：**

**<script>**

**function sortNumber(a,b)**

**{**

**return b-a;**

**}**

**var arr = [5,9,3];**

**document.write(arr + "<br>")**

**document.write(arr.sort(sortNumber));**

**</script>**

3、如何获取一个随机数

**难度等级：【简单】**

**答案：**

**Math.floor(Math.random()\*(max-min+1)+min)**

### Day04

1、for(i=0,j=0;i<10,j<7;i++,j++){

k=i+j;

console.log(k);

}

输出结果为？

**难度等级：【中等】**

**答案：0 2 4 6 8 10 12**

2、按html和按selector查找的区别。

**难度等级：【中等】**

**答案：**

1. **从返回值来看：**
   1. **按HTML查找: 返回动态集合**

**优点是：率高！只需返回需要的数据即可，不需要返回完整数据；**

**缺点是：会造成反复查找；**

* 1. **按选择器查找: 返回非动态集合**

**优点是：不会反复查找；**

**缺点是：首次执行效率低；**

1. **从易用性来看，当查找条件复杂时：**
   1. **按HTML查找: 繁琐**
   2. **按选择器查找: 简洁**

**所以，如果根据一个条件就可获得想要的元素，首选按HTML查找；如果查找条件复杂时，首选按选择器查找。**

3、如何解决声明提前所带来的问题？

**难度等级：【中等】**

**答案：**

**在程序开始执行前，都会将var声明的变量和function声明的函数，提前到当前作用域的顶部，集 中创建。赋值留在原地。再开始执行程序。但是存在的问题是，这破坏了程序默认的执行顺序，那么 怎样解决呢？**

**1. 建议: 将所有的变量和函数声明都集中在当前作用域顶部。**

**2. ES6: let 代替var**

**let禁止在声明之前提前使用变量**

**强制将声明集中在当前作用域顶部创建**

**限制: 不能用let反复创建相同名称的变量**

**3. 函数: 第2种创建函数的方式：**

**var函数名=function (参数变量列表){... ...}**

**用以上方式创建的函数不会被声明提前**

### Day05

1. 数组去重 例如arr=[5,2,3,5,1,4,3,6,6] 如何得到 arr=[1,2,3,4,5,6]

**难度等级：【中等】**

**答案：**

**var arr=[5,2,3,5,1,4,3,6,6];  
 var arr1=[];  
 for(var i=0;i<arr.length;i++){  
 if(arr1.indexOf(arr[i]) == -1){  
 arr1.push(arr[i]);  
 }  
 }  
 arr1.sort();  
 function arr1(arr){  
 arr.sort();  
 for(var i = 0; i<arr.length;i++){  
 if(arr[i]!=arr[i+1]){  
 arr1.push(arr[i]);  
 }**

**}  
 return arr1;  
 }**

1. 如何判断是否为闰年？(代码！！)

**难度等级：【简单】**

**答案：**

**<script>**

**//定义函数检查一个年份是否为闰年**

**function isLeap(year){**

**//如果year是闰年，就输出year是闰年**

**//否则,就输出year是平年**

**document.write(year+**

**(year%4==0&&year%100!=0||year%400==0?**

**"是闰年":"是平年")**

**);**

**}//闰年: year可被4整除且不能被100整除**

**//或**

**// 可直接被400整除**

**//测试:**

**isLeap(prompt("请输入年份:"));**

**</script>**

1. 编写函数，实现冒泡排序。

**难度等级：【中等】**

**答案：**

**<script>**

**var array = [5, 4, 3, 2, 1];**

**var temp = 0;**

**for (var i = 0; i < array.length; i++){**

**for (var j = 0; j < array.length - i; j++){**

**if (array[j] > array[j + 1]){**

**temp = array[j + 1];**

**array[j + 1] = array[j];**

**array[j] = temp;**

**}**

**}**

**}**

**console.log(array);**

**</script>**

### Day06

1、分别写出下列情况的正则验证表达式。

身份证号码验证

电子邮箱验证

手机号码验证

**难度等级：【中等】**

**答案：**

1. **身份证号码验证**

**/(^\d{15}$)|(^\d{18}$)|(^\d{17}(\d|X|x)$)**

**b) 电子邮箱验证**

**/^([a-zA-Z0-9\_-])+@([a-zA-Z0-9\_-])+(.[a-zA-Z0-9\_-])+/**

**/^([a-zA-Z0-9\_\.\-])+\@(([a-zA-Z0-9\-])+\.)+([a-zA-Z0-9]{2,4})+$/**

**c) 手机号码验证**

**(\+86|0086)?\s\*1[34578]\d{9}**

### Day07

1. JS的typeof返回哪些数据类型?

**难度等级：【简单】**

**答案：**

**“undefined” ——未定义  
 “boolean”——**[**布尔值**](https://www.baidu.com/s?wd=%E5%B8%83%E5%B0%94%E5%80%BC&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YvnWcvrHD3n1whmhRknWmz0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnH6znHc4nH6LnjDknW03n1mvr0) **“string”——字符串  
 “number”——数值  
 “object”——对象或者null;  
 “function”——函数**

1. 输出今天的日期，以YYYY-MM-DD的方式，例如今天是2017年5月11日，输出后为2017-05-11。

**难度等级：【中等】**

**答案：**

**var o = {**

**outputDate:function(){**

**var arrDate = [];**

**var nowDate = new Date();**

**var year = nowDate.getFullYear();**

**var month = nowDate.getMonth() + 1;**

**month = month<10 ? '0'+month : month;**

**var day = nowDate.getDate();**

**day = day<10 ? '0'+day : day;**

**console.log(year+'-'+month + '-' +day);**

**}**

**};**

**o.outputDate();**

1. 从0-9中随机选取5个数，这5个数不能重复，并且按大到小排列。

**难度等级：【中等】**

**答案：**

**function rn(min,max){**

**return Math.floor(Math.random()\*(max-min)+min);**

**}**

**var arr=[];**

**for (var i = 0; i < 5; i++) {**

**arr.push(rn(0,9));**

**}**

**console.log(arr.sort());**

1. 此段代码执行结果是什么？

(function(){

var x = foo( );

var foo = function foo( ){

return "foobar"

};

return x;

})( );

**难度等级：【中等】**

**答案：**

**代码报错，错误为Uncaught TypeError: foo is not a function(…)**

1. 简述内置对象。

**难度等级：【中等】**

**答案：**

**内置对象就是指这个语言自带的一些对象，供开发者使用，这些对象提供了一些常用的或是最基本而 必要的功能，这些对象你可以使用，但是通常不建议修改。Js中的内置对象包括：String，Number， Boolean，Array，Date，Math，Function，Error，Object，Global。**

### Day08

1. 计算出距离中午下课还有多长时间(中午12:00下课)

**难度等级：【中等】**

**答案：**

**<script>**

**var end=**

**new Date("2017/05/23 12:00:00");**

**var now=new Date();**

**var s=parseInt((end-now)/1000);**

**var d=parseInt(s/3600/24);**

**//s/3600/24,再下取整**

**var h=parseInt(s%(3600\*24)/3600);**

**//s/(3600\*24)的余数,再/3600,再下去整**

**var m=parseInt(s%3600/60);**

**//s/3600的余数,再/60，再下取整**

**s%=60;//s/60的余数**

**//距离清明放假还有: ?天?小时?分?秒**

**document.write(**

**d+"天"+h+"小时"+m+"分"+s+"秒");**

**</script>**

## 五、DOM

### Day01

1、题目：用节点间关系，遍历指定父节点下所有后代节点

**难度等级：【中等】**

**答案：**

**1. 递归遍历: 2步:**

**Step1: 先仅查找直接子节点**

**Step2: 对每个直接子节点，调用和父节点完全相同的函数**

**算法: 深度优先遍历: 每个节点都优先遍历子节点，子节点遍历完，才遍历兄弟节点。**

**问题: 递归时，函数内的函数名不能写死**

**解决: 用arguments.callee指代当前函数自己。**

**2. 用循环代替递归:**

**节点迭代器对象: 依次遍历，并返回每个节点对象的小工具——内置深度优先算法**

**如何: 2步:**

**1. 创建节点迭代器对象:**

**var iterator=document.createNodeIterator(**

**parent, NodeFilter.SHOW\_ALL, null, false**

**SHOW\_ELEMENT**

**);**

**2. 循环调用迭代器的nextNode()，调到下一个节点。直到返回null结束。**

**do{**

**var node=iterator.nextNode();**

**if(node!=null){**

**//输出node**

**}else break;**

**}while(true);**

### Day02

1、如何在页面中添加一个新的元素？

**难度等级：【简单】**

**答案：**

**方法一、**

**将需要创建的元素，以字符串的形式拼接；找到父级元素，直接对父级元素的innnerHTML进行赋值。**

**方法二、**

**使用Document、Element对象自带的一些函数，来实现动态创建元素（创建元素 => 找到父级元素 => 在指定位置插入元素）**

**解决思路分三步：**

**1、创建元素：Document.createElement()**

**2、找到父级元素 ：可以通过Id、name、标签名、class、匹配指定的css选择器**

**3、在指定位置插入元素：element.appendChild()、element.insertBefore()**

### Day03

1. 定时器有哪几种以及有何区别，并且写出下题中输出的结果**。**

**function test() {  
for(var i = 0; i < 3; i ++)  
 setTimeout(function(){  
 console.log(i);   
 },0);  
}  
test();**

**难度等级：【简单】**

**答案：**

**定时器有：一次性定时器和周期性定时器。区别：一次性定时器只执行一次就结束，周期性的会反复执行。**

**3个3**

2、写一段代码，使某个元素的背景颜色在红黄蓝之间交替，间隔时间为10s。

**难度等级：【简单】**

**答案：**

**Css：**

**div{**

**width: 200px;**

**height: 200px;**

**}**

**Html：**

**<div id="d1"></div>**

**Js：**

**<script>**

**var arr=['red','yellow','blue'],i=0;**

**setInterval(function(){**

**d1.style.backgroundColor=arr[i];**

**i++;**

**if(i>2){**

**i=0;**

**}**

**},10000)**

**</script>**

3、for (var i = 0; i < 5; i++) {

setTimeout(function() {

console.log(i);

}, 1000 \* i);

}

改写以上代码，使其能够每隔1000ms递增输出以下内容：0 1 2 3 4 5

**难度等级：【中等】**

**答案：**

**for(var i=0;i<5;i++){**

**setTimeout(function(i){**

**return function(){**

**console.log(i);**

**}**

**}(i),1000\*i)**

**}**

**Day04**

1、绑定事件处理函数有几种方式？具体是什么？

**难度等级：【中等】**

**答案：**

**在JavaScript中，有三种常用的绑定事件的方法：**

**1、在DOM元素中直接绑定：语法为：onXXX="JavaScript Code"；**

**2、在JavaScript代码中绑定：语法为：**

**elementObject.onXXX=function(){**

**// 事件处理代码**

**}**

**3、 绑定事件监听函数，用addEventListener() 或 attachEvent()。**

**addEventListener()函数语法：**

**elementObject.addEventListener(eventName,handle,useCapture);**

**attachEvent()函数语法：**

**elementObject.attachEvent(eventName,handle);**

**六、JAVASCRIPT CORE**

**Day01**

1. 打印输出结果为？

function a(xx){

this.x=xx;

return this;};

var x=a(5);

var y=a(6);

console.log(x.x);

console.log(y.x);

**难度等级：【中等】**

**答案：undefined 6**

1. 请简述你知道的错误类型。

**难度等级：【中等】**

**答案：**

**1.SyntaxError（语法错误）**

**解析代码时发生的语法错误**

**eg:var 1a;**

**Uncaught SyntaxError: Unexpected number**

**2.ReferenceError（引用错误）**

**a.引用了一个不存在的变量**

**eg: console.log(a);**

**Uncaught ReferenceError: a is not defined**

**b.将变量赋值给一个无法被赋值的对象**

**eg:console.log()= 1;**

**Uncaught ReferenceError: Invalid left-hand side in assignment**

**3.RangeError（范围错误）**

**超出有效范围**

**eg:var a= new Array(-1);**

**Uncaught RangeError: Invalid array length**

**4.TypeError（类型错误）**

**a.变量或参数不是预期类型，比如，对字符串、布尔值、数值等原始类型的值使用new命令，就会抛出这种错误，因为new命令的参数应该是一个构造函数。**

**eg: var a= new 123;**

**Uncaught TypeError: 123 is not a function**

**b.调用对象不存在的方法**

**eg:var a;a.aa();**

**Uncaught TypeError: Cannot read property 'aa' of undefined**

**5.URLError（URL错误）**

**与url相关函数参数不正确，主要是encodeURI()、decodeURI()、encodeURIComponent()、decodeURIComponent()、escape()和unescape()这六个函数。**

**eg: decodeURI('%2')**

**Uncaught URIError: URI malformed**

**6.EvalError（eval错误）**

**eval函数没有被正确执行**

3、闭包有什么特点？

**难度等级：【中等】**

**答案：在JavaScript中，有三种常用的绑定事件的方法：**

4、绑定事件处理函数有几种方式？具体是什么？

**难度等级：【中等】**

**答案：**

**①.封闭性：外界无法访问闭包内部的数据，如果在闭包内声明变量，外界是无法访问的，除非闭包主动向外界提供访问接口；**

**②.持久性：一般的函数，调用完毕之后，系统自动注销函数，而对于闭包来说，在外部函数被调用之后，闭包结构依然保存在**

**系统中，闭包中的数据依然存在，从而实现对数据的持久使用。**

**优点：**

**① 减少全局变量。**

**② 减少传递函数的参数量**

**③ 封装；**

**缺点：使用闭包会占有内存资源，过多的使用闭包会导致内存溢出等.**

**Day02**

1. http://item.taobao.com/item.htm?a=1&b=2&c=&d=xxx&e，请写一段JS程序提取URL中的各个GET参数(参数名和参数个数不确定)，将其按key-value形式返回到一个json结构中，如{a:’1′, b:’2′, c:”, d:’xxx’, e:undefined}。

**难度等级：【中等】**

**答案：**

**function getData(url){**

**var result ={};**

**var temp = url.split('?')[1].split('&');**

**for(i in temp){**

**var s = temp[i].split('=');**

**result[s[0]] = s[1];**

**}**

**return result;**

**}**

1. JS中如何实现继承?

**难度等级：【中等】**

**答案：**

**（1）原型链继承**

**核心： 将父类的实例作为子类的原型；**

**（2）构造继承**

**核心：使用父类的构造函数来增强子类实例，等于是复制父类的实例属性给子类（没用到原型）；**

**（3）实例继承**

**核心：为父类实例添加新特性，作为子类实例返回；**

**（4）拷贝继承**

**（5）组合继承**

**核心：通过调用父类构造，继承父类的属性并保留传参的优点，然后通过将父类实例作为子类原型，实现函数复用；**

**（6）寄生组合继承**

**核心：通过寄生方式，砍掉父类的实例属性，这样，在调用两次父类的构造的时候，就不会初始化两次实例方法/属性，避免的组合继承的缺点。**3、绑定事件处理函数有几种方式？具体是什么？

**难度等级：【中等】**

**答案：**

**在JavaScript中，有三种常用的绑定事件的方法：**

3、function Foo() {

getName = function(){

console.log(1);

};

return this;

}

Foo.getName = function () { console.log(2);};

Foo.prototype.getName = function () { console.log(3);};

var getName= function () { console.log(4);};

function getName() {console.log(5);}

Foo.getName();

getName();

Foo().getName();

getName();

new Foo.getName();

new Foo().getName();

**难度等级：【难】**

**答案：**

**Foo.getName();//2**

**getName();//4**

**Foo().getName();//1**

**getName();//1**

**new Foo.getName();//2**

**new Foo().getName();//3 (new Foo()).getName() //实例化Foo() Foo()的getName 因为Foo()没有getName，继续往上找,找构造函数的prototype // var a=new A() A.prototype.x=xxxx;**

**new new Foo().getName();**

**//new ((new Foo()).getName)();//3**

4、function add(a){

var sum =a;

function f(b){

sum+=b;

return f;

}

f.toString=function(){

return sum;

}

return f;

}

console.log(add(1)(2));

var acc=add(5)(2);

console.log(acc);

console.log(acc(10));

console.log(acc(10)(30));

console.log(acc);

**难度等级：【中等】**

**答案：**

**console.log(add(1)(2));//3**

**var acc=add(5)(2);//f**

**console.log(acc);//7**

**console.log(acc(10));//17**

**console.log(acc(10)(30));//57**

**console.log(acc);//57**

5、var length=10;

function fn(){

console.log(this.length);

}

var obj={

length:5,

method:function(a){

a();

arguments[0]();

}

}

obj.method(fn);

**难度等级：【中等】**

**答案： 10 1**

6、补全\_\_\_\_\_处代码。

function FUN(){

this.name=\_\_\_\_\_;

this.age=\_\_\_\_\_;

}

var obj=new FUN('小哪吒',15);

console.log(

obj.name,obj.age

);

**难度等级：【中等】**

**答案： aguments[0] aguments[1]**

7、补全？？？？处代码。

<script type="text/javascript">

function Animal(){

this.name = "Animal";

this.showName = function(){

console.log(this.name);

}

}

function Cat(){

this.name = "Cat";

？？？？

this.showName1 = function(){

console.log(this.name);

}

this.showName2 = function(){

console.log(this.name);

}

this.showName3 = function(){

console.log(this.\_super.name + "=>" + this.name);

}

}

console.log(Cat.prototype);

Animal.call(this);

？？？？

var cat = new Cat();

console.log(cat instanceof Animal); //true

cat.showName1(); //"Cat"

cat.showName2.call(Cat.prototype); //"Animal"

cat.showName3(); //"Animal" => "Cat"

</script>

<script type="text/javascript">

function Animal(){

this.name = "Animal";

this.showName = function(){

console.log(this.name);

}

}

function Cat(){

this.name = "Cat";

this.\_super = Cat.prototype;

this.showName1 = function(){

console.log(this.name);

}

this.showName2 = function(){

console.log(this.name);

}

this.showName3 = function(){

console.log(this.\_super.name + "=>" + this.name);

}

}

console.log(Cat.prototype);

Animal.call(this);

Cat.prototype = new Animal();

var cat = new Cat();

console.log(cat instanceof Animal); //true

cat.showName1(); //"Cat"

cat.showName2.call(Cat.prototype); //"Animal"

cat.showName3(); //"Animal" => "Cat"

</script>

**难度等级：【中等】**

**答案：this.\_super = Cat.prototype; Cat.prototype = new Animal();**

8、谈谈对new的理解。

**难度等级：【中等】**

**答案：**

**new操作符产生的动作有：**

1. **开辟一块内存，标识为instance；**
2. **把SubType本身（不包括原型对象）包含的属性和值注入到instance里面；**

**3、 添加\_\_proto\_\_属性，并且把值设置为SubType的prototype属性指向的那个原型对象。**

9、创建一个飞机构造函数Flyer其具有坐标x和y，速度speed属性，图片地址src属性。

声明一个英雄机hero对象其继承飞机对象Flyer所有的属性

其中x=100 y=200 speed=100 src=hero.jpg

加分项：

请用ES6语法完成以上代码。

当鼠标在游戏面板中移动时，hero英雄机跟随鼠标位置进行移动

html结构为：

<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Document</title>

<style>

#game{

width:350px;

height:600px;

border:1px solid #CCC;

margin:0 auto;

}

</style>

</head>

<body>

<div id="game"></div>

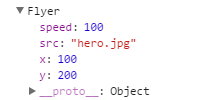
</body>

<script>

//请用面向对象的思想实现这段代码

//创建一个飞机构造函数Flyer，包含属性x,y,speed,src

//实例化一个hero对象，继承Flyer全部属性，输出此对象，正确结构应为



</script>

</html>

**难度等级：【难】**

**答案：**

**Css：**

**#game{**

**width:350px;**

**height:600px;**

**border:1px solid #CCC;**

**margin:0 auto;**

**position:relative;**

**}**

**img{**

**position:absolute;**

**}**

**Html：**

**<div id="game"></div>**

**Js：**

**<script>**

**/\*第一种：\*/**

**/\*function Flyer(x,y,speed,src){**

**this.x=x;**

**this.y=y;**

**this.speed=speed;**

**this.src=src;**

**}**

**var hero=new Flyer(100,200,100,"hero.jpg");**

**console.dir(hero);\*/**

**class Flyer{**

**constructor(x,y,speed,src){**

**this.x=x;**

**this.y=y;**

**this.speed=speed;**

**this.src=src;**

**}**

**}**

**/\*class Hero extends Flyer**

**{**

**constructor(){**

**super(x,y,speed,src);**

**}**

**}\*/**

**var hero=new Flyer(100,200,100,'img/hero.png');**

**var gamePanel=document.getElementById("game");**

**var frag=document.createDocumentFragment();**

**var img=new Image();**

**img.style.cssText=`top:${hero.y}px;left:${hero.x}px`;**

**img.src=hero.src;**

**frag.appendChild(img);**

**gamePanel.appendChild(frag);**

**var game={**

**start(){**

**gamePanel.onmousemove=function(e){**

**var x=e.offsetX;**

**var y=e.offsetY;**

**this.paint(x,y);**

**}.bind(this);**

**},**

**paint(x,y){**

**/\*gamePanel.innerHTML=**

**gamePanel.innerHTML.replace(**

**/<img .\*>/g,""**

**);\*/**

**this.paintHero(x,y);**

**},**

**paintHero(x,y){**

**//speed暂时无用**

**img.style.cssText=`top:${y}px;left:${x}px`;**

**},**

**move(){**

**}**

**}**

**game.start();**

**</script>**

10、 请编写一个JavaScript函数 parseQueryString，它的用途是把URL参数解析为一个对象，如：var url = “http://www.test.cn/index.php?key0=0&key1=1&key2=2″;

最后函数返回一个对象,其内容为

var params={

"key0":"0",

"key1":"1",

"key2":"2"

}

说明：上机操作题，代码成功返回所需对象内容后将代码写在答案内。无法成功的此题不用填写。

**难度等级：【难】**

**答案：**

**function parseQueryString(url){**

**var params = {};**

**var arr = url.split("?");**

**if (arr.length <= 1)**

**return params;**

**arr = arr[1].split("&");**

**for(var i=0, l=arr.length; i<l; i++){**

**var a = arr[i].split("=");**

**params[a[0]] = a[1];**

**}**

**return params;**

**}**

**var url = "http://www.test.cn/index.php?key0=0&key1=1&key2=2";**

**var obj = parseQueryString(url);**

**Day03**

1、function step(a){

return function(x){

return x+a++;

}

}

var a = step(10); var b = step(20);

alert(a(10)); alert(a(20)); alert(a(10));

alert(b(10)); alert(b(20));

**难度等级：【中等】**

**答案： 20 31 22 30 41**

2、谈谈你对访问器属性的理解。

**难度等级：【中等】**

**答案：**

**访问器属性不包含数据值。它包含一对getter和setter函数。当读取访问器属性时，会调用getter函数并返回有效值；当写入访问器属性时，会调用setter函数并传入新值，setter函数负责处理数据。该属性有四个特性：**

**1.[[Configurable]]：默认为true。表示能否通过delete删除属性从而重新定义属性，能否修改属性特性，或者能否把属性修改为访问器属性；**

**2.[[Enumerable]]：默认为true。表示能否通过for-in循环返回属性；**

**3.[[Get]]：读取属性时调用的函数，默认为undefined；**

**4.[[Set]]：写入属性时调用的函数，默认为undefined。**

**访问器属性不能直接定义，必须通过Object.defineProperty()函数定义。**

3、使用原生JS编写对象深度Clone方法?

**难度等级：【难】**

**答案：**

**var s = {"a": "aaaaa"},**

**lightClone = {},**

**deepClone = {};**

**//浅clone**

**lightClone = s;**

**lightClone.a = 'lightClone';**

**console.log(lightClone.a);**

**console.log(s.a);**

**//深clone**

**s = {"a": "aaaa"};**

**for(var k in s) {**

**deepClone[k] = s[k];**

**}**

**deepClone.a = "deepClone";**

**console.log(deepClone.a);**

**console.log(s.a);**

**Day04**

1、写出程序的运行结果。

for(i=0, j=0; i<10, j<6; i++, j++){

k = i + j;

}

console.log(i,j,k);

**难度等级：【中等】**

**答案：6 6 10**

2、for of有哪些局限性？

**难度等级：【中等】**

**答案：**

**局限: 1. 只能遍历索引数组和类数组对象**

**不能遍历关联数组和对象: for in**

**2. 无法获得房间号**

**3. 只能逐个遍历所有，不能控制循环的开始和结束以及步调。**

**4. 如果数组中保存的是原始类型的值，那么，修改val，不会影响数组元素——按值传递:其实val只是数组元素值的副本。**

**何时: 只要仅遍历元素值，不关心下标时，才可用for of简写。**

3、浅谈class对面向对象的简化。

**难度等级：【中等】**

**答案：**

**1. 用class结构包裹构造函数和原型对象方法:**

**2. 构造函数名提升为类名,构造函数统一命名为constructor**

**3. 直接定义在class中的函数直接量会自动保存在当前类型的原型对象中**

**七、jQuery**

**Day01**

1、将jquery对象转化成dom对象的方法？例如：

<button class=”btn”></button>

$(“.btn”) 选取了一个按钮 button，现将其转化为DOM对象，请写出你的方法？

**难度等级：【简单】**

**答案：**

**jquery对象->dom对象**

**a:var dom对象 = $jquery对象.get(0);**

**b:var dom对象 = $jquery对象[0];**

**$(".btn")[0]**

**$(".btn").get(0)**

**Day02**

1、jQuery中访问标准属性，如何访问？修改呢？

**难度等级：【简单】**

**答案：**

**访问：$(ele).attr(“属性名”);**

**修改：$(ele).attr(“属性名”,属性值);**

2、创建数组对象。

1:var list = [{"id":1001,"name":"tom",salary:”3000”,”sex”:”男”},

{"id":1002,"name":"jerry",salary:”5000”,”sex”:”男”},

{"id":1003,"name":"kk",salary:”7000”,”sex”:”女”}];

2:依据数组创建对象jquery对象

3:结果是一个表格

****

**难度等级：【难】**

**答案：**

**<table id="tables">**

**<thead>**

**<tr>**

**<th>员工编号</th>**

**<th>员工姓名</th>**

**<th>员工薪酬</th>**

**<th>员工性别</th>**

**</tr>**

**</thead>**

**<tbody></tbody>**

**</table>**

**<script src="jquery-1.11.3.js"></script>**

**<script>**

**var list= [{"id":1001,"name":"tom","salary":3000,"sex":"男"},**

**{"id":1002,"name":"jerry","salary":5000,"sex":"男"},**

**{"id":1003,"name":"kk","salary":7000,"sex":"女"}**

**];**

**var html="";**

**var tds="";**

**for(var i=0;i<list.length;i++){**

**var arr=[];**

**for(var key in list[i]){**

**arr.push(list[i][key]);**

**tds+="<td>"+key+"</td>";**

**}**

**html+="<tr>"+tds+"</tr>";**

**}**

**var $html=$(html);**

**$("#tables").append($html);**

**</script>**

3、实现点击 span按钮为页面 div重新按照 data-id 顺序排列（用jQuery实现即可）。

<script src="jquery-1.11.3.js"></script>

<span id=”sort”>排序</span>

<div data-id="1">1111</div>

<div data-id="4">4444</div>

<div data-id="3">3333</div>

<div data-id="5">5555</div>

<div data-id="2">2222</div>

**难度等级：【难】**

**答案：**

**<!DOCTYPE html >**

**<html>**

**<head>**

**<title> Boye </title>**

**<meta charset="utf-8"/>**

**<script src="jquery-1.11.3.js"></script>**

**<script type="text/javascript">**

**$(function(){**

**var arr1=$("div").toArray();**

**var arr=arr1.slice(0,arr1.length);**

**var s=arr.sort(function(a,b){**

**return $(a).attr('data-id')-$(b).attr('data-id')**

**})**

**$("p").append(arr);**

**});**

**</script>**

**</head>**

**<body>**

**<span></span>**

**<div data-id="1">1111</div>**

**<div data-id="4">4444</div>**

**<div data-id="3">3333</div>**

**<div data-id="5">5555</div>**

**<div data-id="2">2222</div>**

**<p></p>**

**</body>**

**</html>**

1. **SERVER&HTTP**

**Day01**

1、请写出以下操作的sql语句。

表名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

（1）新增一行记录(小王 13254748547 高中毕业 2007-05-06)

（2）删除名为张四的全部记录

**难度等级：【简单】**

**答案：**

**(1)INSERT INTO user VALUES('小王','13254748547','高中毕业','2007-05-06');**

**(2)DELETE FROM user WHERE Name='张四';1、使用原生JS编写对象深度Clone方法?**

2、查询数据库book内所有类型为数学的图书信息，并且更新id为3的图书价格为56，表结构如下：

bid btype bname bprice

1 数学 高数 68

2 化学 高分子材料 40

3 数学 概率理论 32

4 汉语 古汉语学 65

5 语言 计算机英语 35

表名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

（1）新增一行记录(小王 13254748547 高中毕业 2007-05-06)

（2）删除名为张四的全部记录

**难度等级：【中等】**

**答案：**

**select \* from book where btype=‘数学’;**

**Update book set bprice where id=3;**

**Day04**

1、解释以下HTTP状态码的含义，你还用过哪些？比如：200,400,401,403,404,500。

**难度等级：【简单】**

**答案：**

**200 服务器成功地接受了客户端请求**

**400 Bad Request 请求出现语法错误【服务器不理解请求的语法。】。**

**403 Forbidden 资源不可用。服务器理解客户的请求，但拒绝处理它（服务器拒绝请求）。通常由于服务器上文件或目录的权限设置导致。**

**404 Not Found 无法找到指定位置的资源。这也是一个常用的应答。**

**500 Internal Server Error 服务器遇到了错误，不能完成客户的请求。**

**例如：**

**304 Not Modified 客户端有缓冲的文档并发出了一个条件性的请求（一般是提供If-Modified-Since头表示客户只想比指定日期更新的文档）。服务器告诉客户，原来缓冲的文档还可以继续使用。**

**504 Gateway Timeout 网关超时，由作为代理或网关的服务器使用，表示不能及时地从远程服务器获得应答 。**

1. **AJAX**

**Day02**

1、AJAX请求时，如何解释json数据？

**难度等级：【简单】**

**答案：**

**eval（）**

**JSON.parse();**

**Day03**

1. 根据以下接口文档内容，实现一个jquery的Ajax请求。

接口功能描述：点击live.html页面，视频列表中的任意视频时

****

跳转至对应的页面live\_intro.html

****

获取URL地址，http://tmooc.cn/web/live\_v1/live\_intro.html?classId=28959599

并拿到classId的值 ，向以下接口发送请求。可返回对应的视频具体内容

请求地址：http://tmooc.cn/livevideo\_v1/findLiveVideoById

请求数据示例: {classID:20445756}

请求方式：POST

**难度等级：【简单】**

**答案：**

**<script>**

**function loadLive(){**

**var url=location.href;**

**var classId=url.split("?")[1].split("=")[1];**

**$.ajax({**

**url:'http://tmooc.cn/livevideo\_v1/findLiveVideoById',**

**type:'POST',**

**data:{classId:classId},**

**success:function(data){**

**console.log(data);**

**}**

**})**

**}**

**</script>**

2、在你使用ajax的过程中，你觉得ajax的优点有哪些？缺点呢？

**难度等级：【简单】**

**答案：**

**ajax的优点：**

**1、最大的一点是页面无刷新，用户的体验非常好。**

**2、使用异步方式与服务器通信，具有更加迅速的响应能力。**

**3、可以把以前一些服务器负担的工作转嫁到客户端，利用客户端闲置的能力来处理，减轻服务器和带宽的负担，节约空间和宽带租用成本。并且减轻服务器的负担，ajax的原则是“按需取数据”，可以最大程度的减少冗余请求，和响应对服务器造成的负担。**

**4、基于标准化的并被广泛支持的技术，不需要下载插件或者小程序。**

**ajax的缺点：**

**1、ajax不支持浏览器back按钮。**

**2、安全问题 AJAX暴露了与服务器交互的细节。**

**3、对搜索引擎的支持比较弱。**

**4、破坏了程序的异常机制。**

**5、不容易调试。**

1. **Node.js**

**Day01**

1、nodejs的模块有哪些？你熟悉的有哪些具体模块？

**难度等级：【简单】**

**答案：**

**（1）Express框架**

**（2）jade模块**

**（3）forever模块**

**（4）Socket.IO模块**

**Day02**

1、请问NodeJs中，怎样使用文件系统模块，读取以下内容：

（1）建立文件"story.txt“，如下内容（“一年前,学校广场上献血.200CC送一副修指甲的用具,400CC送个手表. 邻班一姑娘听说了感觉很幸福,跑过去问护士:"1000CC送什么?" 护士淡定的说:"送个棺材……"”）

（2）完成读取以上文件的readtext.js 方法。

**难度等级：【中等】**

**答案：**

**var fs = require("fs") ;**

**fs.readFile("story.txt","utf8",function (error,data){**

**if(error) throw error ;**

**console.log(data) ;**

**}) ;**

1. **HTML5CORE**

**Day01**

1、列举html5中10个新特性。

**难度等级：【简单】**

**答案：**

**(1)新的语义标签**

**(2)增强型表单(表单2.0)**

**(3)音频和视频**

**(4)Canvas绘图**

**(5)SVG绘图**

**(6)地理定位**

**(7)拖放API**

**(8)Web Worker**

**(9)Web Storage**

**(10)Web Socket**

**Day02**

1、使用canvas绘制动态增长的进度条（从0%增长到100%，前20%是红色，后80%部分是绿色）。

****

**难度等级：【简单】**

**答案：**

**<body>**

**<canvas width="100" height="20" id="c">**

**您的浏览器不支持Canvas标签**

**</canvas>**

**<script>**

**var ctx=c.getContext('2d');**

**ctx.strokeRect(0,0,100,20);**

**var x=0;**

**var t=setInterval(function(){**

**if(x<=20){**

**ctx.fillStyle='red';**

**ctx.fillRect(x,0,1,20);**

**x++;**

**}**

**if(x>20){**

**ctx.fillStyle='green';**

**ctx.fillRect(x,0,1,20);**

**x++;**

**}**

**if(x>100){**

**clearInterval(t);**

**}**

**},200)**

**</script>**

**</body>**

**Day03**

1、用svg实现如下的效果。

****

**难度等级：【简单】**

**答案：**

**<svg>**

**<circle id="c1" cx="250" cy="200" r="100" fill="#fff" stroke="green" stroke-width="3"/>**

**<polygon points="250,110 170,240 330,240" style="fill:transparent;stroke:green;stroke-width:3"/>**

**<polygon points="250,290 170,160 330,160" style="fill:transparent;stroke:green;stroke-width:3"/>**

**</svg>**

**Day05**

1、请描述一下cookies，sessionStorage和localStorage的区别？

**难度等级：【中等】**

**答案：**

**共同点：都是保存在浏览器端，且同源的。**

**区别：**

**(1)cookie数据始终在同源的http请求中携带（即使不需要），即cookie在浏览器和服务器间来回传递。而sessionStorage和localStorage不会自动把数据发给服务器，仅在本地保存。cookie数据还有路径（path）的概念，可以限制cookie只属于某个路径下。**

**(2)存储大小限制也不同，cookie数据不能超过4k，同时因为每次http请求都会携带cookie，所以cookie只适合保存很小的数据，如会话标识。sessionStorage和localStorage 虽然也有存储大小的限制，但比cookie大得多，可以达到5M或更大。**

**(3)数据有效期不同，sessionStorage：仅在当前浏览器窗口关闭前有效，自然也就不可能持久保持；localStorage：始终有效，窗口或浏览器关闭也一直保存，因此用作持久数据；cookie只在设置的cookie过期时间之前一直有效，即使窗口或浏览器关闭。**

**(4)作用域不同，sessionStorage不在不同的浏览器窗口中共享，即使是同一个页面；localStorage 在所有同源窗口中都是共享的；cookie也是在所有同源窗口中都是共享的。**

**(5)Web Storage 支持事件通知机制，可以将数据更新的通知发送给监听者。**

**(6)Web Storage 的 api 接口使用更方便。**