**中软国际考题-笔试**

1. 单选题

1、在html5标准中，为input标签的type属性增加了若干类型，如下不属于新增类型的是（B）

A.database B.number C.date D.color

2、不属于HTML5 废弃的标签选项是（B）

A. applet B.ol C. center D. frameset

3、下列标签中用于标记媒介元素的是（B）

A. <summary>标签 B. <source>标签 C. <video>标签 D. <progress>标签

4、canvas用于填充颜色的属性是（Ａ）

A.fillStyle B.fillRect C.lineWidth D.strokeRect

5、下列哪一项表示的不是按钮。（C）

A．type="submit" B．type="reset" C．type="image" D．type="button"

6,HTML5 中的 <canvas> 元素用于（C）

A、显示数据库记录 B、操作 MySQL 中的数据 C、绘制图形 D、创建可拖动的元素

7、HTML5中，那个元素用于组合标题元素（A）

A. <hgroup> B. <hgroups> C. <group> D. <groups>

8、下列哪一个属性规定用于验证input域的模式（D）

A. novalidate B. required C.placehodler D. pattern

9、以下是HTML5新增的标签是（A）

A. <aside> B. <isindex> C. <samp> D. <s>

10、关于HTML5说法正确的是（C）

A. HTML5只是对HTML4的一个简单升级

B. 所有主流浏览器都支持HTML5

C. HTML5新增了离线缓存机制

D. HTML5主要是针对移动端进行了优化

11、在HTML5中，哪个元素用于组合标题元素（D）

A. <group> B. <header> C. <headings> D. <hgroup>

12、在 HTML5 中，哪个元素用于组合标题元素？（D）

A. <group> B. <header> C. <headings> D. <hgroup>

13、HTML5 中不再支持下面哪个元素？（D）

A. <q> B. <ins> C. <menu> D. <font>

14、HTML5 中的 <canvas> 元素用于？（C）

A. 显示数据库记录

B. 操作 MySQL 中的数据

C. 绘制图形

D. 创建可拖动的元素

15、下列选项，那个属于HTML5的离线存储（B）

A.request B.sessionStorage C.cookie D.session

16、以下是HTML5新增的标签是(A)

    A、<aside>  B、<isindex>  C、<samp>    D、<s>

1. 多选题
2. 下面哪些项属于HTML扩展的功能？（A、B、C、E）
3. Canvas绘图 B. video C. 索引数据库
4. 在线地图工具 E. 离线存储

2、下面哪些HTML5新增的标签（A、B、C）

A. <mark> B. <time> C. <footer> D. <ul> E. <span>

3、以下哪些是HTML5特有的存储类型（　ＡＣＤ　）

A. localStorage B. cookie C. Application cache应用缓存 D. sessionStorage

4、以下TAG中合法的html元素有（A,B,C,D）

A〈body〉B〈a〉C〈html〉D〈title〉

5、下列属于HTML特有的存储方式有哪些？（ABC）

A. localStorage B. sessionStorage C. sessionStorage D. cookie

6、<video>元素支持哪几种视频格式？（BCD）

A. avi B. mpeg4 C. ogg D. webm

7、HTML5支持的视频格式有（A、B、D）

A. Ogg B. mp4 C. flv D. Web

8、以下选项是HTML5特有的存储类型有（A、C、D）

A. localStorage B. Cookie C. ApplicationCache D. sessionStorage

9、下列哪些是HTML5新加入的标签？（C、D、E）

A. p B. h1 C. hgroup

D. figure E. header

E. title

10、下面哪些项属于HTML5中新增的内容？（A、B、C、D、E）

A. article B. footer C. header D. nav E. section

11、以下说法正确的是：(B,C,D)

A、HTML5 标准还在制定中

B、HTML5兼容以前HTML4下浏览器

C、<canvas>标签替代Flash

D、简化的语法

1. 填空题（每空1分，共10分）
2. \_\_\_\_\_\_\_\_\_\_用于持久化的本地存储，数据永远不会过期，关闭浏览器也不会丢失。\_\_\_\_\_\_\_\_\_在同一个会话的页面才能访问并且当会话结束后数据也会随之销毁。 (localStorage、sessionStorage )
3. \_\_\_\_\_\_\_\_\_标签定义可选数据的列表。与input元素配合使用，就可以制作出输入值的下拉列表。(<datalist>)
4. \_\_\_\_\_\_\_\_\_标签定义命令按钮，比如单选按钮、复选框或按钮。(<command>)
5. HTML5之前的html版本是 HTML4.01 。
6. HTML5的正确doctype是 <!DOCTYPE html> 。
7. 用于播放html5视频文件是　<video>　　，音频文件是　<audio>　　。
8. 定义滑块控件　range　　　的输入类型。
9. 定义周和年控件（无时区） week 的输入类型。
10. HTML5 元素用于显示已知范围内的标量测量 <meter> 。
11. <tr>….</tr>是用来定义\_\_\_\_\_\_\_\_\_\_；<td>…</td>是用来定义\_\_\_\_\_\_\_\_\_；<th>…</th>是用来定义\_\_\_\_\_\_\_(表格的一行，表格的一列，表格的标题)

设定图片边框的属性是\_\_\_\_\_\_\_\_\_\_\_ (border)

1. HTML5的canvas元素使用\_\_\_\_\_\_\_\_\_\_\_\_\_\_在网页上绘制图像。(JavaScript)
2. 在HTML5中，onblur和onfocus是\_\_\_\_\_\_\_\_\_\_。（事件属性）
3. HTML5中来获得用户的位置方法是（getCurrentPosition()）
4. HTML5所支持的媒体标签分别是：\_\_\_\_\_\_\_\_\_\_ 、\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_ 、\_\_\_\_\_\_\_\_\_\_ 、\_\_\_\_\_\_\_\_\_。( audio、video、embed、source、track)
5. canvas的方法有哪些\_\_[createLinearGradient()](http://www.w3school.com.cn/tags/canvas_createlineargradient.asp" \o "HTML5 canvas createLinearGradient() 方法)\_\_\_\_\_\_\_\_ 、\_\_\_[createPattern()](http://www.w3school.com.cn/tags/canvas_createpattern.asp" \o "HTML5 canvas createPattern() 方法)\_\_\_\_\_\_。
6. 判断题
7. 在html5中可以使用canvas绘制矢量图。（ F ）
8. 高版本的浏览器都实现了索引数据库功能，可以永久存储数据。（ T ）
9. <canvas>标签定义图形，比如图表和其他图像。这个HTML元素是为了客户端矢量图形而设计的。（ T ）
10. <video>标签定义声音，比如音乐或其他音频流（ F ）
11. 文档脱机时触发事件是onoffline。　　（　Ｔ　）
12. 苹果野生动物园浏览器最新版本中不支持html5。　　（　Ｆ　）
13. 在会话终止后，会话存储数据将被浏览器删除。　　（　Ｔ　）
14. Itemprop不是用于分组项目。　（　Ｆ　）
15. 当元素被拖动到有效的放置目标上时触发ondragover事件。　（　Ｔ　）
16. HTML文档的所有标签都是成对出现，如<h1>....<h1>。F
17. 使用专业化的网页制作工具能够提高HTML文档的制作效率。T
18. HTML5全局属性"contenteditable"用于规定元素内容是否是可编辑的。（ T ）
19. 用于播放HTML5视频文件的正确HTML5元素是<vedio>。（T）
20. 用于播放HTML5音频文件的正确HTML5元素是<audio>。（T）
21. <video> 与 </video> 之间插入的内容是供不支持 video 元素的浏览器显示的。（ T ）
22. HTML5不是基于SGML的，因此不需要对DTD进行引用；HTML4基于SGML的，因此需要对DTD进行引用。（ T ）
23. readAsText方法有3个参数，其中第二个参数是文本的编码方式，默认值为 UTF-8。（ T ）

五、简答题

1、HTML5的文件离线储存怎么使用，工作原理是什么？

答案：在线情况下，浏览器发现HTML头部有manifest属性，它会请求manifest文件，如果是第一次访问，那么浏览器就会根据manifest文件的内容下载相应的资源，并进行离线存储。如果已经访问过并且资源已经离线存储了，那么浏览器就会使用离线的资源加载页面。然后浏览器会对比新的manifest文件与旧的manifest文件，如果文件没有发生改变，就不会做任何操作，如果文件改变了，那么就会重新下载文件中的资源，并且进行离线存储。

2、HTML5有哪些新增的表单元素？

答：HTML5新增了很多表单元素让开发者构建更优秀的web应用程序。

Datalist、datetime、output、keygen 、date 、month 、week、time、color

number 、range 、email 、url。

3、HTML5 标准提供了哪些新的 API？

答：HTML5 提供的应用程序 API 主要有：

Media API

Text Track API

Application Cache API

User Interaction

Data Transfer API

Command API

Constraint Validation API

History API

4、HTML5中如何嵌入音频？

答：HTML5支持MP3、Wav和Ogg格式的音频，下面是在网页中嵌入音频的简单实例：

<audio controls>

          <source src="jamshed.mp3" type="audio/mpeg">

            Your browser doesn't suppport audio embedding feature.

<audio>

5、HTML5 Canvas元素有什么用？

答：Canvas 元素用于在网页上绘制图形，该元素标签强大之处在于可以直接在HTML上进行图形操作，

<canvas id="canvas1" width="300" height="100"></canvas>

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

cookie在浏览器和服务器间来回传递。 sessionStorage和localStorage不会

sessionStorage和localStorage的存储空间更大；

sessionStorage和localStorage有更多丰富易用的接口；

sessionStorage和localStorage各自独立的存储空间；

7、html：HTML（HyperTextMark-upLanguage）即超文本[标记语言](https://wenwen.sogou.com/s/?w=%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80&ch=ww.xqy.chain" \t "_blank)或超文本链接标示语言，是目前网络上应用最为广泛的语言，也是构成网页文档的主要语言。

8、简述HTML5中Canvas和SVG区别（5分）

答案： SVG 是一种使用 XML 描述 2D 图形的语言。SVG 基于 XML，这意味着 SVG DOM 中的每个元素都是可用的。您可以为某个元素附加 JavaScript 事件处理器。在 SVG 中，每个被绘制的图形均被视为对象。如果 SVG 对象的属性发生变化，那么浏览器能够自动重现图形。Canvas 通过 JavaScript 来绘制 2D 图形。Canvas 是逐像素进行渲染的。在 canvas 中，一旦图形被绘制完成，它就不会继续得到浏览器的关注。如果其位置发生变化，那么整个场景也需要重新绘制，包括任何或许已被图形覆盖的对象。

9、HTML5表单新增的input输入类型有哪些？

答案：Autocomplete属性；autofocus属性；form属性；表单重写属性；height与width 属性；list属性；min、max和step属性；multiple属性；pattren属性； placeholder属性； required属性

10、请列举HTML5 中的一些有趣的新特性。(分别写出3条即可)（5分）

答案：

用于绘画的 canvas 元素

用于媒介回放的 video 和 audio 元素

对本地离线存储的更好的支持

新的特殊内容元素，比如 article、footer、header、nav、section

新的表单控件，比如 calendar、date、time、email、url、search

11、HTML5 废弃了哪些 HTML4 标签。(分别写出3个即可)（5分）

答案：

frame

frameset

noframe

applet

big

center

basefront

12、HTML5 有哪些新增的表单元素。(分别写出3个即可)（5分）

答案：

datalist

datetime

output

keygen

date

month

week

time

color

number

range

email

url

13、HTML5 存储类型有哪些，它们之间有什么区别？（5分）

答案：

HTML5 提供了下面两种本地存储方案：

localStorage ：用于持久化的本地存储，数据永远不会过期，关闭浏览器也不会丢失。

sessionStorage： 同一个会话中的页面才能访问并且当会话结束后数据也随之销毁。因此sessionStorage不是一种持久化的本地存储，仅仅是会话级别的存储。

14、html5有哪些新特性

新特性，新增元素：

1）内容元素：article、footer、header、nav、section

2）表单控件：calendar、date、time、email、url、search

3）控件元素：webworker，websockt，Geolocation

15、html5移除了那些元素

移除元素：

1）显现层元素：basefont，big，center，font，s，strike，tt，u

2）性能较差元素：frame，frameset，noframes

16、如何处理HTML5新标签的浏览器兼容问题

处理兼容问题有两种方式：

1）IE6/IE7/IE8支持通过document方法产生的标签，利用这一特性让这些浏览器支持HTML5新标签。

2）使用是html5shim框架

另外，DOCTYPE声明的方式是区分HTML和HTML5标志的一个重要因素，此外，还可以根据新增的结构，功能元素来加以区分

17、HTML5的离线储存怎么使用，工作原理能不能解释一下？

答案解析：

localStorage 长期存储数据，浏览器关闭后数据不丢失；

sessionStorage 数据在浏览器关闭后自动删除

18、如何实现浏览器内多个标签页之间的通信？

答案解析：

调用localstorge、cookies等本地存储方式

19、webSocket如何兼容低浏览器？

答案解析：

Adobe Flash Socket 、 ActiveX HTMLFile (IE) 、 基于 multipart 编码发送 XHR 、 基于长轮询的 XHR

20、Doctype作用? 严格模式与混杂模式如何区分？它们有何意义?

答案解析：

1）<!Doctype>声明位于文档中的最前面，处于<html>标签之前。告知浏览器的解析器，用什么文档类型规范来解析这个文档。

2）严格模式的排版和JS运作模式是以该浏览器支持的最高标准运行。

3）在混杂模式中，页面以宽松的向后兼容的方式显示。模拟老式浏览器的行为以防止站点无法工作。

4）DOCTYPE不存在或格式不正确会导致文档以混杂模式呈现。

21、iframe有那些缺点？

答案解析：

1）在网页中使用框架结构最大的弊病是搜索引擎的“蜘蛛”程序无法解读这种页面；

2）框架结构有时会让人感到迷惑，页面很混乱；

22、如何实现浏览器内多个标签页之间的通信？

答案解析：

调用localstorge、cookies等本地存储方式

23、webSocket如何兼容低浏览器？

答案解析：

Adobe Flash Socket 、 ActiveX HTMLFile (IE) 、 基于 multipart 编码发送 XHR 、 基于长轮询的 XHR

六、程序实现题

1、canvas绘图：使用HTML5中扩展的canvas在网页上绘制一条直线

（10分）

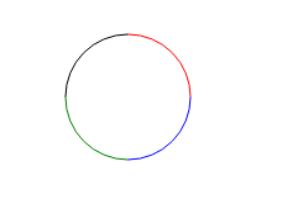
|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title>HTML5 Canvas绘制线条入门示例</title>  </head>  <body>  <!-- 添加canvas标签，并加上红色边框以便于在页面上查看 -->  <canvas id="myCanvas" width="400px" height="300px" style="border: 1px solid red;">  您的浏览器不支持canvas标签。  </canvas>  <script type="text/javascript">  //获取Canvas对象(画布)  var canvas = document.getElementById("myCanvas");  //简单地检测当前浏览器是否支持Canvas对象，以免在一些不支持html5的浏览器中提示语法错误  if(canvas.getContext){  //获取对应的CanvasRenderingContext2D对象(画笔)  var ctx = canvas.getContext("2d");    //注意，Canvas的坐标系是：Canvas画布的左上角为原点(0,0)，向右为横坐标，向下为纵坐标，单位是像素(px)。    //开始一个新的绘制路径  ctx.beginPath();  //定义直线的起点坐标为(10,10)  ctx.moveTo(10, 10);  //定义直线的终点坐标为(50,10)  ctx.lineTo(50, 10);  //沿着坐标点顺序的路径绘制直线  ctx.stroke();  //关闭当前的绘制路径  ctx.closePath();  }  </script>  </body>  </html> |

2、使用HTML5中的canvas实现太极图

（10分）

|  |
| --- |
| <!DOCTYPE HTML>  <html>  <body>  <canvas id="myCanvas" width="600" height="400" style="background-color: #EED2EE; ">  </canvas>  <script type="text/javascript">  var c=document.getElementById("myCanvas");  var cxt=c.getContext("2d");  //绘制左侧的白色半圆直径都是150  cxt.fillStyle="#FFFFFF";  cxt.beginPath();  cxt.arc(300,200,150,1.5\*Math.PI,Math.PI/2,false);  cxt.closePath();  cxt.fill();  //绘制右侧的黑色半圆  cxt.fillStyle="#000000";  cxt.beginPath();  cxt.arc(300,200,150,Math.PI/2,1.5\*Math.PI,false);  cxt.closePath();  cxt.fill();  //绘制下面的黑色圆  cxt.fillStyle="#000000";  cxt.beginPath();  cxt.arc(300,275,75,0,2\*Math.PI,false);  cxt.closePath();  cxt.fill();  //绘制上面的白色圆  cxt.fillStyle="#FFFFFF";  cxt.beginPath();  cxt.arc(300,125,75,0,2\*Math.PI,false);  cxt.closePath();  cxt.fill();  //绘制两个小圆  cxt.fillStyle="FFFFFF";  cxt.beginPath();  cxt.arc(300,275,10,0,2\*Math.PI,false);  cxt.closePath();  cxt.fill();  //绘制黑色小圆  cxt.fillStyle="#000";  cxt.beginPath();  cxt.arc(300,125,10,0,2\*Math.PI,false);  cxt.closePath();  cxt.fill();  </script>  </body>  </html> |

**3、**利用HTML5中的canvas制作以下效果（10分）



代码实现：

<!DOCTYPE html>

<htmls>

<head>

<meta charset="UTF-8">

<title>canvas</title>

</head>

<style>

</style>

<body>

<div id="canvas-warp">

<canvas id="canvas" style=" display: block; margin: 50px auto;">

1

</canvas>

</div>

<script>

window.onload = function() {

var canvas = document.getElementById("canvas");

canvas.width = 800;

canvas.height = 600;

var context = canvas.getContext("2d");

context.beginPath();

context.arc(100, 70, 50, 0.5 \* Math.PI, 1 \* Math.PI);

context.strokeStyle = "green";

context.fillStyle = "white";

context.fill();

context.stroke();

context.beginPath();

context.arc(100, 70, 50, 1 \* Math.PI, 1.5 \* Math.PI);

context.strokeStyle = "dimgray";

context.fillStyle = "white";

context.fill();

context.stroke();

context.beginPath();

context.arc(100, 70, 50, 1.5 \* Math.PI, 2 \* Math.PI);

context.strokeStyle = "red";

context.fillStyle = "white";

context.fill();

context.stroke();

context.beginPath();

context.arc(100, 70, 50, 2 \* Math.PI, 0.5 \* Math.PI);

context.strokeStyle = "blue";

context.fillStyle = "white";

context.fill();

context.stroke();

}

</script>

</body>

</html>

**4、**在Canvas画布上使用您指定的颜色来绘制渐变背景。（10分）

|  |
| --- |
| <!DOCTYPE HTML>  <html>  <body>  <canvas id="myCanvas" width="200" height="100" style="border:1px solid #c3c3c3;">  Your browser does not support the canvas element.  </canvas>  <script type="text/javascript">  var c=document.getElementById("myCanvas");  var cxt=c.getContext("2d");  var grd=cxt.createLinearGradient(0,0,175,50);  grd.addColorStop(0,"#FF0000");  grd.addColorStop(1,"#00FF00");  c xt.fillStyle=grd;  cxt.fillRect(0,0,175,50);  </script>  </body>  </html> |

**5、**写一个简单的拖放实例

（10分）

|  |
| --- |
| <!DOCTYPE HTML>  <html>  <head>  <script type="text/javascript">  function allowDrop(ev)  {  ev.preventDefault();  }  function drag(ev)  {  ev.dataTransfer.setData("Text",ev.target.id);  }  function drop(ev)  {  ev.preventDefault();  var data=ev.dataTransfer.getData("Text");  ev.target.appendChild(document.getElementById(data));  }  </script>  </head>  <body>  <div id="div1" ondrop="drop(event)"  ondragover="allowDrop(event)"></div>  <img id="drag1" src="img\_logo.gif" draggable="true"  ondragstart="drag(event)" width="336" height="69" />  </body>  </html> |

**6、**编写显示本地时间格式日历的HTML代码（10分）

7、使用画布绘制图片

（10分）

|  |
| --- |
| <script type="application/javascript">  /\*绘制画布\*/  function drawCanvas(image1) {  var canvas1 = document.getElementById('canvas1');  var context = canvas1.getContext('2d');    //绘制图片到画布中，参数：图片对象，x坐标，y坐标，宽度，高度  context.drawImage(image1,20,30,image1.width,image1.height);  }    function init() {  //创建图片对象  var image1 = new Image();  image1.src = 'icss.jpg';    //当图片被加载完毕之后调用绘制画布程序  image1.onload = function() {  drawCanvas(image1)  };  }    window.onload = function() {  init();  };  </script>  <body>  <canvas id="canvas1" width="200" height="200" style="border:1px #CCCCCC solid">  您的浏览器不支持画布操作  </canvas>  </body> |