复习：

HTML5新特性：

(1)新的语义标签和属性

(2)表单新特性

1)新的input type——10个

email、url、number、tel、search、range、color、date、month、week

2)新的表单元素——4个

datalist、progress、meter、output

3)表单元素的新属性——12个

autocomplete、autofocus、placeholder、multiple、form

required、maxlength、minlength、max、min、step、pattern

(3)视频和音频

(4)Canvas绘图

(5)SVG绘图

(6)地理定位

(7)拖放API

(8)WebWorker

(9)WebStorage

(10)WebSocket

今日目标：

(1)视频和音频 —— 简单必须掌握

(2)Canvas绘图 —— 重点&难点

1.面试题：Flash被HTML5取代在哪些方面？

音频和视频 —— <video>和<audio>

绘图 —— <canvas>

动画/游戏 —— <canvas>+定时器

统计图表 —— <canvas>、<svg>

客户端数据存储 —— WebStorage

2.HTML5新特性——视频播放

HTML5提供了一个新的标签<video>标签，用于播放视频。该标签默认是一个300\*150的inline-block。使用方法：

<video src="x.mp4"></video>

<video>

<source src="x.mp4">

<source src="x.ogg">

<source src="x.webm">

</video>

VIDEO元素/对象的属性：

(1)src：指定要播放的视频的资源路径

(2)autoplay：false，是否自动播放

(3)controls：false，是否显示播放控件，提示：不同浏览器的播放控件外观不同

(4)currentTime：0，当前播放的时间点（s）

(5)duration：60，影片总时长（s）

(6)ended：false，是否播放到结尾

(7)loop：false，是否循环播放

(8)muted：false，是否静音

(9)volume：1，音量设置（0~1），对象属性，不用于标签

(10)paused：当前是否处于暂停状态

(11)poster：''，指定视频第一帧播放前的电影海报

(12)preload：指定视频预加载方案，可取值：

auto：默认值，自动预加载视频的宽高、时长、第一帧内容、并缓冲了一定的时长

metadata：元数据，只预加载视频的宽高、时长、第一帧内容

none：不预加载任何内容

**VIDEO对象的方法：**

play()：开始播放

pause()：暂停播放

**VIDEO对象的事件：**

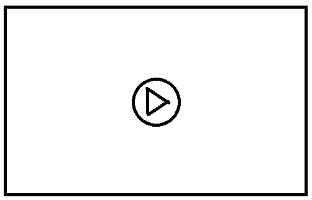
onplay：视频开始播放（可能多种原因引起）

onpause：视频开始暂停（可能多种原因引起）

onplaying:

练习：

(1)不使用video默认的播放控件，使用自定义的按钮，控制视频的播放和暂停



(2)不论何种原因，影片一暂停即显示出广告图片；一播放广告图片就隐藏；提示：不能使用poster属性（影片海报只能在第一次播放之前显示一次）

午间思考：如何将video作为DIV的背景并自动播放？

3.HTML5新特性——音频播放

HTML5提供了一个新的标签<audio>标签，用于播放音频。该标签若没有controls属性，则默认display:none；反之则是一个300\*30的inline-block。使用方法：

<audio src="x.mp3"></audio>

<audio>

<source src="x.mp3">

<source src="x.ogg">

<source src="x.wav">

</audio>

AUDIO元素/对象的属性：

(1)src：指定要播放的视频的资源路径

(2)autoplay：false，是否自动播放

(3)controls：false，是否显示播放控件，提示：不同浏览器的播放控件外观不同

(4)currentTime：0，当前播放的时间点（s）

(5)duration：60，影片总时长（s）

(6)ended：false，是否播放到结尾

(7)loop：false，是否循环播放

(8)muted：false，是否静音

(9)volume：1，音量设置（0~1），对象属性，不用于标签

(10)paused：当前是否处于暂停状态

(12)preload：指定视频预加载方案，可取值：

auto：默认值，自动预加载视频的宽高、时长、第一帧内容、并缓冲了一定的时长

metadata：元数据，只预加载视频的宽高、时长、第一帧内容

none：不预加载任何内容

**AUDIO对象的方法：**

play()：开始播放

pause()：暂停播放

**AUDIO对象的事件：**

onplay：视频开始播放（可能多种原因引起）

onpause：视频开始暂停（可能多种原因引起）

onplaying:

练习：为网页添加自动播放的背景音乐，用户可选暂停或继续



可以使用定时器修改audio.volume属性实现音量淡入和淡出。

提示：

(1)<body bgsound="x.mp3">属性是老IE的专有属性，也可以用于播放背景音乐，但音量、静音、暂停、停止都无法精准的控制。其它浏览器不支持此属性。

(2)当前iOS中的Safari浏览器默认不支持audio标签！只能使用video标签代替。

4.Web前端中可用的绘图技术

在网页中绘图可以使用的功能：

(1)实时走势图

(2)统计图表

(3)随机内容的图片

(4)在线画图板

(5)HTML5游戏——2D/3D

可用的绘图技术：

(1)Canvas技术 —— 专用于绘制2D图形/图像

(2)SVG技术 —— 专用于绘制矢量图

(3)WebGL技术 —— 目前不是HTML5标准技术，功能最强大，3D图形/图像

5.Canvas绘图技术 —— 最重要&难点

|  |
| --- |
| 难点： (1)小学/中学数学知识 (2)单词记忆 |

HTML5引入了<canvas>标签用于绘图，默认是一个300\*150的inline-block。使用width/height属性指定尺寸，**但不能使用CSS样式指定宽和高**！

<canvas width="600" height="500" id="c">

您的浏览器不支持Canvas标签！

</canvas>

往“画布”上绘图需要使用其对应的“画笔”对象：

var ctx = c.getContext('2d'); //绘图上下文——“画笔”

接下来所有的绘图任务都由画笔实现。

Content：内容

Context：上下文

**绘图上下文对象的常用属性**——console.log(ctx)：

fillStyle：'#000'，填充样式

strokeStyle：'#000'，描边/轮廓样式

lineWidth：1，描边/轮廓的宽度

font：'10px sans-serif'，绘制文本所用的字号/字体

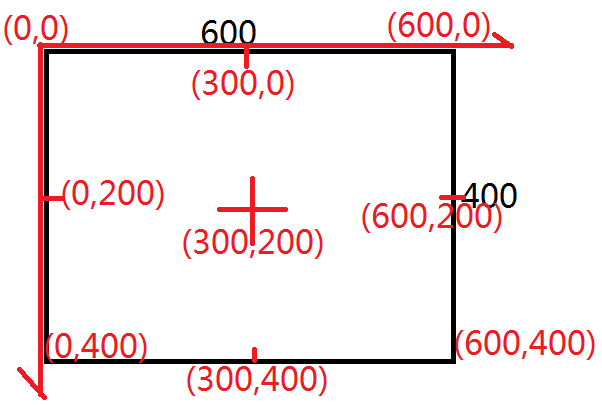
textBaseline：'alphabetic'，文本对齐的基线

showdowOffsetX：0，阴影水平偏移量

showdowOffsetY：0，阴影竖直偏移量

showdowColor：'rgba(0,0,0,0)'，阴影颜色

showdowBlur：0，阴影模糊半径



6.使用Canvas绘制矩形

提示：矩形的定位点在自己左上角

ctx.fillStyle = '#000' 填充颜色

ctx.strokeStyle = '#000' 描边颜色

ctx.fillRect(x,y,w,h) 填充一个矩形

ctx.strokeRect(x,y,w,h) 描边一个矩形

ctx.clearRect(x,y,w,h) 清除一个矩形范围内的所有内容

练习：在600\*400的画布上绘图 —— 16:10

(1)左上角填充一个矩形100\*80，默认颜色

(2)右上角描边一个矩形100\*80，默认颜色

(3)左下角填充一个矩形100\*80，红色

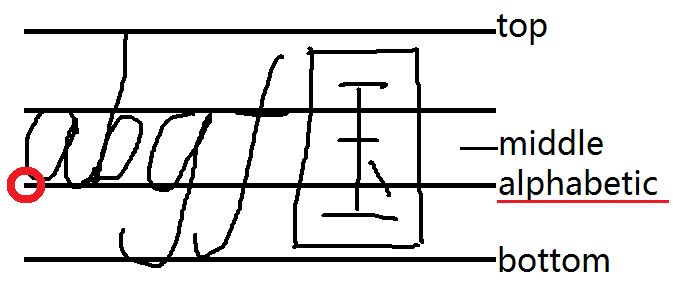
(4)右下角描边一个矩形100\*80，青色

(5)正中央填充+描边一个矩形100\*80，注意是什么颜色

(6)重新创建一个画布，使用定时器，绘制一个可以不断向右移动的矩形

(7)绘制一个斜向30度角移动的矩形

课下挑战性任务：绘制一个绕圆形路径移动的矩形



7.使用Canvas绘制文本

提示：文字的定位点默认在文本基线的起点(左侧)

ctx.textBaseline = 'alphabetic' 文本基线，可取为top/bottom/middle/alphabetic

ctx.fillText(txt, x, y) 填充文本

ctx.strokeText(txt, x, y) 描边文本

ctx.measureText(txt).width 测量，根据当前指定的字号和字体计算指定文本的宽度

练习：绘制文本

(1)在左下角绘制一行文本

(2)在右下角描边一行文本

(3)在画布中央绘制一个可以向右移动的文本，向户外LED招牌一样

课下挑战性任务：绘制可以画布上左右移动的文字

8.使用Canvas绘制路径

下次课内容

9.使用Canvas绘制图像

下次课内容

10.为图形文字添加阴影

ctx.shadowColor = '#666'; //阴影颜色

ctx.shadowOffsetX = 8; //阴影偏移量

ctx.shadowOffsetY = 8;

ctx.shadowBlur = 10; //阴影模糊半径

11.在绘图时使用渐变色

//创建渐变对象

var g = ctx.createLinearGradient(50,100, 550,100);

g.addColorStop(0, '#f00'); //添加颜色点

g.addColorStop(1, '#0f0'); //添加颜色点

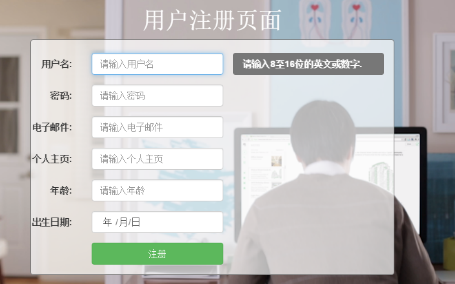
//使用渐变对象

ctx.fillStyle = g;

ctx.strokeStyle = g;

课后练习：

(1)使用视频做DIV元素的背景



提示：Video自动播放、循环播放、静音，绝对定位到目标元素下面，z-index为负值即可

(2)使用AJAX从服务器端的PHP页面获取如下的JSON数据：

[

{"label": "部门1", "value":300},

{"label": "部门2", "value":500},

{"label": "部门3", "value":150},

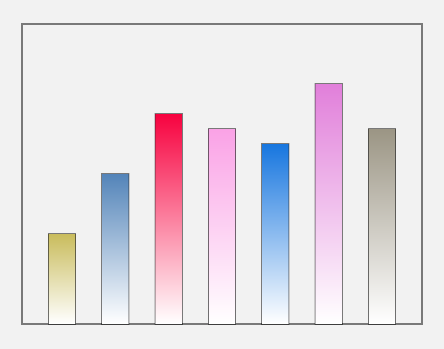
{"label": "部门4", "value":400},

{"label": "部门5", "value":550},

{"label": "部门6", "value":250}

]

根据这些数据，绘制出如下图所示的统计图：



提示：为简化起见，可以把value值看做每个柱的高度。

挑战性需求：柱子初次显示时，高度有动画效果。