## 02.js笔记—多媒体和图形编程

**（一）图形编程概述**

* 图形-使用js直接在客户端生成图片的好处很多：
  + 用于在客户端生成图片的代码大小要比图片小的多，节省带宽；
  + 根据数据生成图形是一件非常消耗CPU的事情，交给客户端处理，可以减轻服务器压力，节省硬件开支；
* 图形-JS可以控制图片的显示地址、宽度、高度等
  + JavaScript可以动态切换图片，为了实现较高的响应度，我们需要提前把要展示的图片缓存起来；
  + JavaScript提供了一个API可以满足该要求：
    - (new Image()).src = '需要缓存的图片路径';
  + 该图片没有追加到文档元素中，它是不可见的；

**（二）多媒体-音频视频控制**

* 音频视频元素：
  + audio：音频标签；video视频标签；
  + 设置controls属性，则显示音频面板，你可以播放、暂停或进行音量控制等；
  + autoplay是自动播放属性；
  + loop是循环播放属性；
  + preload指定用户开始播放媒体前，是否或有多少媒体内容需要加载；
    - 设置为none表示不需要预加载；
    - 设置为metadata则表示需要加载视频时长、帧大小等元数据，不加载视频内容；
    - 设置为auto则表示浏览器自己预加载他认为合适的媒体内容量；
* 音频对象构造函数：
  + Audio()构造函数，可以创建一个音频（视频则没有此构造函数）；
  + Audio构造函数创建的是屏幕外音频，没有任何视觉效果；代码如下：

new Audio('音频路径').play();

* 音频视频元素播放控制：
  + play():开始；
  + pause():暂停；
  + muted：设置为true表示静音；设置为false恢复到原来的音量；
  + volume：音量属性，0~1之间，0是静音，1是最大声音；
  + playbackRate：用于指定播放速度，0~1慢放，1正常播放，大于1快进；defaultPlaybackRate属性则用于指定默认播放速度；
* 音频视频状态查询及控制：
  + paused:暂停属性，停止返回true，播放时返回false；
  + ended：播放完并且停下来返回true；设置了loop循环播放属性的不会返回true；
  + duration：媒体的时长，单位是秒；// 开发建议用定时器获取
  + initialTime：媒体开始时间，一般视频是0；对于流媒体，表示已经缓存的数据的最早时间以及能够回退到的最早时间；
  + currentTime：定点播放，从指定的时间开始播放；
  + played：返回已经播放的时间；
  + buffered：返回已经缓存的时间；
  + seekable：当前播放器需要调到的时间；
  + 这三个属性都是TimeRanges对象，其中：
    - length:属性，表示当前(有多少个)时间段;
    - start:方法，返回当前时间段的起始时间点;
    - end:方法，返回当前时间段的结束时间点;

**（三）音频/视频底层状态查询和事件类型，分别参照状态控制和网页事件进行学习。**











