

# 验证相关的新属性

**required** 必填项,内容不能为空 `<input required>`

**maxlength** 指定字符串的最大长度  
`<input maxlength="8">`

**minlength** 指定字符串的最小长度  
`<input minlength="3">`

**max** 指定数字的最大值 `<input max="60">`

**min** 指定数字的最小值 `<input min="18">`

**pattern** 指定输入必需符的正则表达式  
`<input pattern="1[35789]\d{9}">`

# H5新特性：播放视频

```
<video src="res/birds.mp4"></video>
```

#解码器:

```
<video>
```

```
<source src="res/birds.mp4">
```

```
<source src="res/birds.ogg">
```

```
<source src="res/birds.webm">
```

您的浏览器不支持video播放，请更新

```
</video>
```

本身是一个  
300\*150  
的inline-  
block元素

# 成员属性

`autoplay:false`

是否自动播放

`controls:false`

是否显示播放控件

`loop:false`

是否循环播放

`muted:false`

是否是静音

`poster:""`

在播放第一帧之前显示的海报

`preload:`

{  
  `auto`: 预加载视频的元数据以及缓冲一定时长视频  
  `metadata`: 仅预加载视频的元数据(尺寸, 时长, 第一帧内容)  
  `none`: 不预加载任何数据

看不完的书就推到明天吧  
反正早晚会的忘的

# JS对象属性

currentTime

当前播放的时长

duration

总时长

paused

true 当前视频是否处理暂停状态

volume

1 当前音量

playbackRate

1 大于1快放，小于1 慢放

看不完的书就推到明天吧  
反正早晚会忘的

## 成员方法

play()

pause()

播放视频  
暂停视频

## 事件

onplay

onpause

当视频播放时触发事件  
当视频暂停时触发事件

看不完的书就推到明天吧  
反正早晚会忘的

# H5新特性：播放音频

```
<audio src="res/bg.mp3"></audio>
```

#解码器:

```
<audio>
```

```
<source src="res/bg.mp3" />
```

```
<source src="res/bg.ogg" />
```

```
<source src="res/bg.wav" />
```

您的浏览器不支持 audio 标签

```
</audio>
```

本身是一个  
300\*30的  
inline-block  
元素，没有  
controls 属性

看不完的书就推到明天吧  
反正早晚会忘的

# 成员属性

autoplay:false

是否自动播放

controls:false

是否显示播放控件

loop:false

是否循环播放

muted:false

是否是静音

poster:""

在播放第一帧之前显示的海报

preload:

- auto: 预加载视频的元数据以及缓冲一定时长视频
- metadata: 仅预加载视频的元数据(尺寸, 时长, 第一帧内容)
- none: 不预加载任何数据

看不完的书就推到明天吧  
反正早晚会的忘的



# JS对象属性

currentTime

当前播放的时长

duration

总时长

paused

true 当前视频是否处理暂停状态

volume

1 当前音量

playbackRate

1 大于1快放，小于1 慢放

看不完的书就推到明天吧  
反正早晚会忘的



## 成员方法

play()

pause()

播放视频

暂停视频

## 事件

onplay

onpause

当视频播放时触发事件

当视频暂停时触发事件

看不完的书就推到明天吧  
反正早晚会忘的

# H5新特性: Canvas绘图技术

## Canvas:画布

`<canvas width="400" height="400" id="c2">`  
您的  
`</canvas>`

## 画笔对象

`var ctx = c2.getContext("2d");`

canvas标签在浏览器中默认 300\*150的 inline-block.画布的宽度和高度只能通过html/js属性来赋值,不能用CSS样式.

看不完的书就推到明天吧  
反正早晚会的忘的

# 使用 Canvas绘制矩形

矩形的定位点在自己的左上角

`ctx.lineWidth = 1;` 描边宽度(空心)

`ctx.fillStyle = "#fff";` 填充样式/颜色(实心)

`ctx.strokeStyle="#fff"` 描边样式/颜色(空心)

`ctx.fillRect(x,y,w,h);` 填充一个矩形(实心)

`ctx.strokeRect(x,y,w,h);`描边一个矩形(空心)

`ctx.clearRect(x,y,w,h);` 清除一个矩形范围内所有

绘图

# Canvas绘图中使用渐变对象

线性渐变:linearGradient

径向渐变:radiaGradient

```
var g = ctx.createLinearGradient(x1,y1,x2,y2);
```

```
g.addColorStop(offset,color);
```

...

```
ctx.strokeStyle=g;
```

```
ctx.fillStyle=g;
```

看不完的书就推到明天吧  
反正早晚会忘的