HTMLcode

HTML5新特性 --- （项目后台）

设置：

项目编码 file. setting .editor.file encoding global encoding[utf-8] project encoding[utf-8]

JavaScript编译标准ES-6 file. setting .editor. languages frameworks javascript

服务器Apache file. setting 

字体大小

浏览网页 ALT+F2

新的语义标签

增强型表单

新input type ：email url number search color data month week

新的表单元素：datalist progress meter output

<datalist >

<option>< /option>

</datalist>

datalist本身不可见，为input提供一个建议列表两个ID要一致

新的表单属性

progress进度条

<progress>M</progress>左右晃动进度条

<progress value=”0.1”> </progress>指定值

meter刻度尺

用于表示一个值所值的范围：不可接受红色，还可以接受黄色，非常优秀绿色

<meter min=’0’ max=’xxx’low=下限 high=上限 optimum=最佳值 value=当前值>

output输出

语义标签，没有任何外观样式，等同于span

表单元素的新属性

placeholder

autofocus

multiple运行输入框输入多个值，用,分支

form <form id = ‘1’></form> <input form= ‘1’>带着下面的input一起提交

required 不能为空

maxlength字符串最大长度

minlength

min数值最小值

max

pattern 正则表达式

音频和视频（重点）

flash被h5取代

flash绘图，视频音频播放，客户端存储

视频

<video src=”x.mp4”control视频控件 loop循环 autoplay自动播放 muted静音 poster=url视频封面 preload = ‘auto预加载一定时长视频和原数据 metadata仅加载视频原数据 none’不加载视频任何内容 ></video>：

<video>

<source src=x.mp4></source>

<source src=x.wmv></source>

<source src=x.ogg></source>

您的版本太低

</video>

同层播放（苹果手机）：点击视频IOS手机默认弹出窗口播放视频

添加属性：webkit-playsinline=”true”

playsinline=”true”微信浏览器

x5-video-player-type=”h5”安卓

全屏错位修复：x5-video-player-fullscreen=”true”

x5-video-orientation=”portraint”还需要设置视频监听window.onresize

video专用css样式：

ohject-fit:fill填充（默认） contain保存原有比例，但会留下白边cover:会改变容器cover

视频时事件：

loadstart:客户端开始请求数据

canplaythrough：下载完成可以播放

canpaly:可以播放但是会因为加载暂停

ended：播放结束

timeupdate:播放时间改变

ratechange:播放速率改变

volumechange:音量改变

video.currentTime获取视频时间

video.duration 视频整体时长

音频：<audio src=x.mp3 controls></audio>

<audio>

<source src=x.mp3></source>

<source src=x.wmv></source>

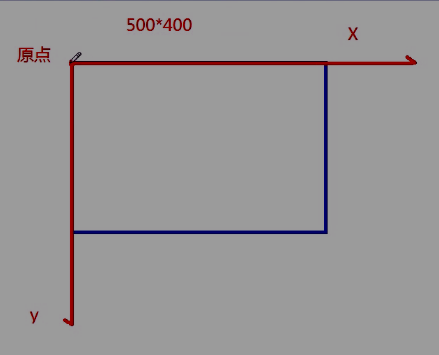
您的版本太低

</audio>

Canvas绘图 2D位图绘图技术（重点）走势图，走势图，游戏，地图应用

<canvas width= height=不能用css赋值width和height></canvas> display为inline-block

原点在左上角



canvas.getConetext(‘2d’)获取画笔

ctx.lineWidth = 1 描边宽度

ctx.fillStyle =”#999”填充样式

ctx.strokeStyle = “#000”填充矩形

ctx.fillRect(x, y, w, h)描边

ctx.clearRect(x, y, w, h)清除矩形范围内所有图形

.textBaseline = ‘alphabetic/top’文本基线

.font = ‘12px sans-serif’文本大小和字体

.fillText(str, x, y)填充一段文字

.strokeText(str, x, y)描边一段文本

.measureText(str)测量文本宽度

path:由多个坐标点组成任意形状，路径不可见，可用于“描，“填充”边”

ctx.beginPath()开始一条新路径

ctx.closePath()闭合当前路径

ctx.moveTo(x, y)移动到指定点

ctx.lineTo(x, y )从当前点到指定点画直线

ctx.arc(cx,cy,start,end) 绘制圆拱型

cx, cy 圆心

r 半径

start, end 开始角度和结束角度

ctx.stroke() 描边

ctx.fill() 填充

ctx.lineWidth= 线的粗细

ctx.strokeStyle = 颜色

图形：canvas属于客户端技术，图片在服务器，所以浏览器先下载，再绘制图片，且等待图下载完成

var p3 = new Image()

p3.src = ‘x.jpg’ 下载指定图片

p3.onload = function(){

ctx.drawImage(p3, x, y)绘制原始大小图片

ctx.drawImage(p3, x, y, w, h)拉升图片

}图片下载成功后触发事件

图形变形

ctx.rotate(弧度)旋转 角度单个对象有累加效应

ctx.translate(x，y)画布原点的平移

ctx.save(); 保存画笔状态

ctx.restore() 回复到save()的状态

SVG绘图 2D矢量绘图技术

使用标签绘图，每个图形都是元素可以直接绑定事件监听

<svg></svg>本身是一个300\*150的inline-block

<rect></rect>矩形width height x y fill fill-opacity stroke stroke-opacity

<circle></circle>原形 r cx cy fill-opacity

<ecllipse><ecllipse>椭圆 rx ry cx cy

<line></line>直线

<polyline></polyline>多边形

动态创建元素：var document.createElementNS(‘http://www.w3.org/2000/svg’, “rect”)

直线 <line x1 y1 x2 y2 stroke-width strock-linecap=round圆角></line>

折线<polyline points=’x ,y x,y x,y ’></polyline>

文本<text x y font-size></text>

图像<image xlink:href=”” x y width height></image>

渐变对象<defs><lineargradient x1 y1 x2 y2>

<stop offset= stop-color=>

<stop offset= stop-color=>

<stop offset= stop-color=>

</defs>

<rect width=400 height= fill=’url(#xx)’></rect>

滤镜：高斯模糊<defs>

<filter ><feGaussianBlur stdDeviation=’2’></ feGaussianBlur >

</ filter >

</defs>

<rect width=400 height= fill=’url(#xx)’></rect>

WebGL绘图 3D绘图技术

地理定位

geolocation:地理定位，使用js获得当前浏览器所在地理坐标（经纬度，海拔速度）等数据，用于是想LBS应用，地图，饿了么等等

获取地理定位类型

浏览器自带js对象geolocation实现不了，联网比较差

window.navigator.geolocatioin的类 调用getCurrentPosition:fn获得定位

用百度地图，腾讯地图的js对象

百度地图开发者<http://lbsyun.baidu.com>

注册账户，创建网站为网站申请访问秘钥

百度地图拾取坐标系统

手机浏览器从GPS芯片定位，或者手机通信基站进行定位，定位精度在KM

pc浏览器 ip地理解析反向查找

拖动API 拖动上传图片。。。

Drag&Drop:拖动，释放

拖动的原对象会动的 3个事件

dragstart 拖动开始

drag 拖动中

dragend 拖动结束

拖动的目标对象不动的4个事件

dragenter 拖动进入

dragover 拖动悬停在上方 ，默认悬停后立即离开，最好阻止默认行为

dragleave 拖动离开

drop 拖动释放

web worker

程序 program可以被CPU执行的代码，储存在外存中

进程 process指陈序被调用内存，分配执行空间，随时供cpu调用

线程 thread线程是进程内执行代码的基本单位

并发和并行：线程在工作时并发状态执行

宏观上同时执行 微观上依次执行

浏览器：

多线程发起http请求下载资源

一个线程进行ui绘制和js代码执行

指定代码<button></button><script></script><button></button>

创建多一个线程工作<script> let x = new Worker(‘js程序’); </script>

注意：worker执行时不允许存在一个或者多个DOM或者BOM元素，因为浏览器只允许ui线程操作DOM、BOM

解决：UI线程发送数据给worker

worker收数据

onmessage = function(e){e.data}

ui发

var w6 = new Worker(.js);

w6.postMessage(stringmsg);

worker发送数据给UI线程

ui收数据

var w5 = new Worker(‘.js’);

w5.onmessage = function(e){e.data}

worker发数据

postMessage();

web storage

在浏览器汇总存储当前用户专用数据访问历史，内容定制，样式定制

在客户端存储数据可以使用技术

1. cookie浏览器兼容性好，操作复杂；不能超过4KB
   1. document.cookie = ‘name=value’ ;保存数据； document.cookie获取数据
2. flash存储，依赖于flash播放器
3. H5 webstorage: 不能超过8MB 操作简单
4. indexDB 存储大量数据，不是标准技术

session会话：一个操作过程称为会话

浏览器从打开某个网站的第一个页面开始会话开始，中间为会话进行中，关闭网页会话结束

两个对象

window.sessionStorage 类数组对象 在同一个会话中所有页面之间共享数据，将数据保存在sessionStorage中，只限本次会话使用

存储数据

sessionStorage[key] = val

sessionStorage.setltem(key,val)

获取数据

let val = sessionStorage[key]

let val = sessionStorage.getltem(key)

sessionStorage.removeltem(key)

sessionStorage.length

sessionStorage.key(i)获取名字

window.localStorage 跨会话存储

把session改成local就行了

localStorage每次修改会触发window.onstorage事件

web socket

http协议。属于请求响应，只有客户端发起请求消息，服务器才会返回响应消息，没有请求就没有响应，一个请求只能得到一个响应，在有些场景不适合：股票的实时走势图

解决方案：长轮询——心跳请求——定时器+ajax

websocket协议：广播-接收模块，客户端连接到服务器不再断开，永久链接，随时向对方发送消息

ws:服务器端：php/java/node.js

ws:客户端php/java/html5提供对象

使用html5创建ws协议客户端

let socket = new WebStocket(“ws://服务器IP：9001”);创建服务器对象

socket.send(stringMessage); 发送消息

socket.onmessage = function(){} 接收消息

socket.close()关闭服务器

two.js

1. 创建2D绘图对象 var two = new Two({

css：

type:Two.Types.xxx

}).appendTo(box);

1. 绘制 two.makeXxxx()
2. 旋转：x.ratation =
3. 组合：x = two.makeGroup(x, y, z….)
4. 平移原点：translation.set(x, y )
5. 把绘制的图像更新在DOM数 two.update();

所有选择以自己为中心，而且不会累加

two.js动画原理：调用two.play()方法，其美妙调用60次two.updat()

two.on(‘unpdate’, function(){})

echarts.js

1. 获取元素，初始echarts实例
   1. var dom
   2. var myChart = echarts.init(dom);
2. 指定图标数据源和配置项
   1. var option = {

title:{text:标题}，

legend:{data:[数据],}，数据来源

xAxis:{data:[x名称数据]}，

yAxis:{}可自动生成，

series：[{name:数据， type:图表名， data：[数值]}]规定各个数据

}

1. 指定echarts实例配置项目
   1. myChart.setOption(option);

还是去看手册吧。。。

HTML5游戏引擎 layabox.js…….baidu