1. Web APIs和js关联

Js：ecmascript，dom，bom

基础：掌 握基本语法，为后面打基础。

Web APIs：主要学习dom和bom，是js所独有的，主要学习页面交互功能。

API（application programing interface，应用程序编程接口）：给程序员提供一种工具，以便能更轻松的实现想要完成的功能，不必纠结内部如何实现的。

Web API：是浏览器提供的一套操作浏览器功能和页面元素的API（BOM，DOM）

1. DOM

DOM（document object model，简称DOM），是W3C组织推荐的处理的可扩展标记语言（html）的标准编程接口。

DOM树：文档（一个页面就是一个文档，用document表示），元素（页面中的所有标签都是元素，使用elemet表示），节点（网页中所有内容都是节点，使用node表示）DOM把以上都看做是对象。

1. 获取元素
2. 根据id获取：因为文档页面是从上往下加载，所以先得有标签，script写到标签的下面；get获得element元素by通过驼峰命名法；参数id是大小写敏感的字符串；返回的是一个元素对象；console.dir打印我们返回的元素对象，更好地查兰里面的属性和方法
3. 根据标签名获取：getElementsByTagName()方法返回带有指定标签名的对象的集合，以伪数组的形式存储的，因为得到的是一个对象的集合，所以想要操作里面的元素就需要遍历，得到元素对象是动态的
4. 通过html5新增的方法获取：getElementsByClassName()方法根据类名获得某些元素的集合；document.querySelector（）元素选择器，返回指定选择器的第一个元素对象，切记里面的选择器需要加符号，.box,#nav; document.querySelectoAll（）返回指定选择器的所有元素对象
5. 特殊元素获取：获取body元素document.body;获取html元素document. documentElement；
6. 事件三要素

Js使我们有能力创建动态页面而事件是可以被js侦测到的行为。

简单理解：触发—相应机制。网页中的每个元素都可以产生某些触发js的事件，例如点击按钮产生一个事件。

事件三要素：事件源（事件被触发的对象 按钮），事件类型（如何触发；鼠标点击 onlick 什么事件），事件处理程序（通过一个函数赋值的方式完成）

执行事件的步骤：获取事件源—注册事件（绑定事件）--添加事件处理程序（采取函数赋值形式）

1. 操作元素

dom操作可以改变网页的内容结构和样式

改变元素的内容：element.innerText(从起位置到终止位置的内容，但他去除html的标签，同时空格和换行也会换掉)不识别html标签，非标准，去除空格和换行；element.innerHTML(起始位置到终止位置的全部内容，包括html标签，同时保留空格和换行)能识别html标签，w3c标准

区别：这两个属性是可读写的，可以获取元素里面的内容

操作元素的属性直接获取元素，根据条件直接修改即可；表单里面的内容是通过value来修改的；this指向的是事件函数的调用者

表单元素的属性操作：type，value，checked，disabled

遇见一个按钮可以点多次的话，可以设置flag变量，点击以后flag的值一定要变化；先把html写好，再把css写好，最后准备js

1. 样式属性操作

Element.style:行内样式操作（行内样式权重比价高，命名采用驼峰命名法）

Element.className:类名样式操作

利用for循环排列精灵图：首先精灵图图片排列是有规律的；核心思路利用for循环，修改精灵图片的背景位置是background-position

获得焦点：onfoucus；失去焦点：onblur

使用element.style获得修改元素样式，如果样式比较少或者功能简单的情况下使用，样式较多可以使用类名样式操作，class是保留字因此使用classname来操作元素类名属性；classname会直接更改元素的类名，会覆盖原先的类名；如果想要保留原来的类名，this.classname=’first change

1. 排他思想

如果有同一组元素，我们想要某一个元素实现某种样式，需要用到循环的排他思想算法，先所有元素清除样式，然后给当前元素设置样式，顺序不能颠倒。

全选取消全选：让下面的所有复选框中的checked属性（选中状态）跟随全选按钮即可

下面全部选中时，上面全选自动选中：给下面所有复选框绑定点击事件，每次点击都要循环查看下面的复选框是否有没选中的，

1. 自定义属性的操作

获取属性值：element.属性（获取的是内置属性值，元素本身自带的属性）；element.getAttrubute(主要获取的是自定义的属性，我们程序员自定的属性)

设置属性值：element.属性=‘值；element.setAttribute(属性，值)主要针对自定义属性

移除属性：removeAttribute（属性）

H5自定义属性：为了保存并使用数据有些数据可以保存到网页中而不用保存到数据库中。规定以data开头作为属性并且赋值，如果data-list-name=‘andy’获取的话两种方法：getAttribute（‘data-list-name’）；.dataset.listname采用驼峰命名法如果自定义属性里面有多个-链接的单词，我们获取的时候采取驼峰命名法

1. 节点操作

获取元素：利用dom提供的方法，逻辑性不强，比较繁琐；利用节点之间的层级关系获取元素，逻辑性强但是兼容性差。

网页中所有的内容都是节点，在dom中，节点使用node来表示。

元素节点nodeType为1；属性节点nodeType为2；文本节点nodeType为3；

在实际开发里面，节点操作主要操作是元素节点。

节点层级：常见的是父子兄层级关系node.parentNode（就近原则）找不到父节点返回为null。子节点childNodes所有的子节点包含元素节点，文本节点等等，如果只想获得里面的元素节点，则需要专门处理，所以我们一般不提倡使用childnodes，children可以得到子元素节点，不会返回其他节点。

兄弟节点：node.nextSibling返回当前元素下一个兄弟节点；node.previousSibling返回当前元素上一个兄弟节点同样也是包含所有节点。

1. 创建和添加节点

Document.creatElement(‘tagName’)动态创建元素节点，因为这些元素原先不存在，是根据我们的需求动态生成的；创建后在进行添加node.appendChild(child)将一个节点添加到指定的子节点列表末尾；node.insertBefore(childe,指定元素)必须指定添加到那个孩子前面。

删除节点：node.removechild(child)---node是父节点this.disabled=true禁用按钮；组织链接跳转需要添加javascript：void（0）或javascript：；

复制节点：node.cloneNode()返回调用该方法的节点的一个副本，如果括号参数为空或者false，则是浅拷贝只复制节点本身，不复制里面的内容；如果括号里面为true则是深拷贝，复制标签加里面的内容data[i][k]前面相当于对象，后面相当于加上属性值。

1. 三种动态创建元素区别

Document.write():直接将内容写如页面的内容流，但是文档流执行完毕，则他会导致页面全部重绘。

Element.innerHTML：是将内容写入某个dom节点，不会导致页面全部重绘

Document.createElement()：创建多个元素效率更高（不要拼接字符串，采取数组形式拼接），结构稍微复杂。

不同浏览器下，innerHTMl效率要比creatElement高

1. Dom重点核心

关于dom的操作。我们主要针对元素的操作。主要有创建，增删改查属性操作，事件操作

1. 事件高级解读

注册事件：给元素添加事件，注册事件有两种方式，传统方式和方法监听注册方式

传统注册方式：利用on开头的事件onclick；<button onclick=’alert(‘hi’)’></button>;

特点：注册事件的唯一性；同一个元素同一事件只能设置一个处理函数，最后注册的处理函数会将前面的注册的处理函数覆盖。

方法监听注册事件：w3c标准推荐方式，addEventListener（）他是一个方法，同一个元素同一个事件可以注册多个监听器（事件处理）

兼容性处理原则：首先照顾大多数浏览器，在处理特殊浏览器

删除事件（解绑事件）：传统注册方式eventTarget.onclick=null;

方法监听注册方式evenTarget.removeEventListener(type,listener[,useCapture]);

1. Dom的事件流

事件流是事件从页面接收事件的顺序，事件发生时会在元素节点之间按照特定的顺序传播，这个传播过程即dom事件流；分为三个阶段捕获阶段，当前目标阶段，冒泡阶段

Js代码只能执行捕获或者冒泡其中的一个阶段；onclick和attachEvent只能得到冒泡阶段document>html>body>father>son,实际开发中很少使用冒泡，有些时间是没有冒泡的onblur，onfocus，onmouseenter、onmouseleave。

阻止事件冒泡：stopPropagation（）

事件委托：又称事件代理，在jquery中又叫事件委派；原理是不是给每个子节点单独设置事件监听器，而是事件监听器设置在器父节点上，然后利用冒泡原理影响设置每个子节点。只操作了一次dom，提高了程序的性能。

1. 事件对象

事件对象写在侦听函数小括号里，当做形参来看，事件对象只有有了事件才会存在，系统自动创建，不需要我们传递实参。事件发生后，跟时间相关的一系列信息数据的集合都放到这对象里面，有很多属性和方法；e.target返回的是触发事件的对象，this返回的是绑定当前事件的对象。

常用的鼠标事件：禁止鼠标右键菜单contexmenu主要控制应该；禁止选中selectstart

1. 鼠标事件对象

event对象代表事件的状态，跟事件相关的一系列信息的集合，现阶段我们主要是用鼠标事件对象mousevent和键盘事件对象keyboardEvent

client返回的是鼠标在可视区的x和y坐标；page的得到的是距离文档页面的x和y坐标；screen返回的是电脑屏幕的x和y坐标

常用鼠标事件小天使移动：鼠标不断移动mousemove，在页面中注册事件，图片要移动距离，而且不占用位置，我们使用绝对定位即可，每次鼠标移动我们都会获得最新的鼠标坐标，把这个x和y坐标作为图片的top和left值就可以移动图片，千万不要忘记给left和top添加px单位。

常用键盘事件：onkeyup某个键盘按键被松开时触发（最后执行）

Onkeydown某个按键被按下是触发（先执行）

Onkeypress某个按键被按下是触发（不能识别功能键）

键盘事件对象：keyup和keydown不区分大小写。

Keydown和keypress在文本框里的特点：他们两事件触发的时候，文字还没有落入文本框中

1. BOM

BOM：浏览器对象模型（整个浏览器），他提供了独立于内容而与浏览器窗口进行交互的对象，其核心对象是window。厂商在各自浏览器上定义的，兼容性较差

由一系列相关的对象组成，并且每个对象都提供了很多方法和属性。

Window对象是浏览器的顶级对象，它具有双重角色，js访问浏览器窗口的一个接口；他是一个全局对象，定义在全局作用域中的变量、函数都会变成window对象的方法和属性。

1. Window对象

窗口加载事件：当文档内容完全加载完成会触发该事件（包括图像，脚本文件，css文件），就调用的处理函数window.onload=function(){ }(传统注册方式)或者window.addEventLisener(‘load’,function(){ })，传统注册方式只能写一次，如果有多个，会以最后一个window.onload为准。

调整窗口大小事件：触发时就调用的处理函数window.onresize=function(){ }或者window.addEventListener(‘resize’,function(){ });只要窗口大小发生像素变化就会触发这个事件，利用这个事件完成响应式布局

1. Settimeout（）定时器

Window.setTimeout(调用函数，[延迟的毫秒数])；设置一个定时器，该定时器在定时器到期后执行调用函数。Window可以省略，这个调用函数可以直接写函数或者写函数名或者采取字符串‘函数名（）’三种形式；定时器可能有很多，所以我们经常给定时器赋值一个标识符。回调函数callback，普通函数是按照代码的顺序直接调用，而这个函数需要等待时间，时间到了才去调用这个函数因此称为回调函数。上一件事干完再回头调用这个函数。

停止定时器：window.clearTimeout(timeoutID)

SetInterval()定时器：window.setInterval(回调函数，[间隔的秒数])；SetInterval()方法重复调用一个函数，每隔一段时间，就去调用一次回调函数；window可以省略，这个调用函数可以直接写函数，或者写函数名或者采取字符串‘函数名（）’三种形式；间隔的毫秒数默认是0，如果写必须是毫秒，表示每隔多少秒就会自动调用这个函数。

倒计时案例：这个倒计时是不断变化的，因此需要定时器来自动变化（setInterval），三个黑盒子存放时分秒，三个黑色盒子利用innerHTML放入计算的小时分钟秒数。

This的指向：是在函数定义的时候是确定不了的，只有函数执行的时候才能确定this指向谁，一般情况下this的最终指向的是调用他的对象。在全局作用域或者普通函数中this指向全局对象window；方法的调用中谁调用this指向谁；构造函数中this指向构造函数的实例

1. Js执行机制

Js语言的一大特点就是单线程，同一时间只能做一件事，所有的任务都需要排队，前一个任务完成才会执行后一个任务，会导致页面渲染不连贯所以允许js脚本创建多个线程，于是出现了同步（前一个任务执行结束才能执行下一个任务，程序的执行与任务的序列是一致的）和异步（在做一件事的同时还会处理其他事情），区别是流水线上各个流程不同。

同步任务：同步任务都放在了主线程上执行，形成一个执行栈。

异步任务：通过回调函数实现的。三种类型：普通事件click、resize；资源加载load、error；定时器setinterval、settimeout；异步任务相关；回调函数添加到任务队列中（任务队列也称为消息队列）。

先执行主线程中的同步任务，遇到回调函数就把异步任务放入任务队列之中，一旦执行栈中的所有同步任务执行完毕，系统就会按次序读取任务队列中的异步任务，于是被读取的异步任务结束等待状态，进入执行栈，开始执行。（当有多个异步任务时会先把异步任务交给异步进程处理，点击后进入异步任务，不点击不进入）eventloop时间循环（由于主线程不断地重复获得任务、执行任务、在获得任务、在执行）

1. Location对象

用于获取或设置窗体的url，并且可以用于解析url，因为这个属性返回的是一个对象，所以称为location对象

url（uniform resource locator）统一资源定位符:是互联网上标准资源的地址，互联网上每个文件都有一个唯一的url，它包含的信息指出文件的位置以及浏览器该怎么处理他。

通信协议，主机，端口号，路径，参数，片段

1. Location对象方法

Location.assign():跟href一样，也可以跳转页面（也称为重定向页面）

Location.replace():替换当前页面，因为不记录历史，所以不能后退页面

Location.reload():重新加载页面，相当于刷新按钮或者F5如果参数为true强制刷新ctrl+F5

1. Navigator和history对象

包含有关浏览器的信息，他有很多属性，最常用的是userAgent，该属性可以返回由客户机发送服务器的user-agent头部的值。

Back（）：可以后退功能；forward（）：前进功能；go（参数）：前进后退功能 参数如果是1前进1个页面，如果是-1就后退一个页面。

1. 元素偏移量

Offset就是偏移量，使用offset系列相关属性可以动态的得到该元素的位置，大小。

获得元素距离带有定位父级元素的位置；获得元素自身的大小（宽度高度）；返回的数值都不带单位。

Element.offsetTop(Left):返回元素相对带有定位父元素上（左）方的偏移; Element.offsetWidth(Height):返回自身包括padding、边框、内容区的高度，返回数值不带单位。

Offset:可以得到任意样式表中的样式值，获得的数值是没有单位的，offsetwidth等属性是只读属性，只能获取不能赋值，所有我们想要获取元素大小位置，用它更合适；

Style：只能得到行内样式值，获得的是带有单位的字符串，style.Width是可读属性，可以获取也可以赋值，所以要想给元素更改值，则需用它。

1. Offset与style的区别

Offset：可以得到任意样式表中的样式值；获得的数值是没有单位的（蓝色）；包含padding+border+width；属性是只读属性，只能获取不能赋值；想要获取元素大小的位置，用offset更合适；

Style：只能得到行内样式表中的样式值；获得的单位是带有单位的字符串（黑色）；获得不包含padding和border的值；是可读属性，也可以获取也可以赋值；想要给元素更改值，则需要style改变。

1. 元素可视区client

Client翻译过来就是客户端，使用client系列的相关属性来获取元素可视区的相关信息。通过client系列的相关属性可以动态的的到该元素的边框大小、元素大小等。Client.Top

返回元素上边框的大小；clientLeft返回元素左边框的大小；clientWidth返回的是自身包括padding、内容区的宽度，不含边框，返回数值不带单位；clientHeight返回的是自身包括padding、内容区的高度，不含边框，返回数值不带单位；

1. 淘宝flexibleJS源码分析

立即执行函数：（function（）{ }）（）或者（function（）{ }（））不需要调用直接执行；第二个小括号可以看做是调用函数 ，可以传递多个实参，中间加逗号即可。立即函数最大的作用就是独立创建了一个作用域，里面的所有变量都是局部变量，不会有命名冲突的情况。

Dpr物理像素比 var dpr=window.devicePixelRatio||1,前面的或者1

1. 元素滚动scroll系列

Scroll翻译过来就是滚动的，我们是用scroll系列的相关属性可以动态的的到该元素的大小、滚动距离等。

Scrolltop（left）是返回被卷去的上（左）侧距离，返回的数值不带单位；scrollWidth（height）返回自身实际的高（宽）度，不含边框，返回数值不带单位。

如果浏览器的高（宽）度 不足以显示整个页面时，会自动出现滚动条。当滚动条向下滚动时，页面上面被隐藏掉的高度，我们就称页面被卷去的头部，滚动条在滚动时会触发onscroll事件。

页面被卷去的头（左侧）部：可以通过window.pageY（X）Offset获得

元素被卷去的头部是element.scrollTop。

Banner.offsetTop就是被卷去头部的大小，一定要写到滚动的外面。

1. 三大系列总结

Offset返回的包括边框；client返回的是不包含边框是内容区宽度；scroll返回的是自身实际的宽度

Mouseover：经过自身盒子触发，经过子盒子还会触发

Mousenter：只会经过自身盒子才会触发，盒子不冒泡，搭配mouseleave同样不会冒泡

1. 动画函数封装

动画实现原理：通过定时器setInterval（）不断移动盒子位置。

实现步骤：获得盒子的当前位置；让盒子当前位置加上1个移动距离；利用定时器不断重复这个操作；注意此元素需要添加定位，才能使用element.style.left

动画函数的封装：函数需要传递两个参数，动画对象和移动到的距离；

动画里别忘了加定位，否则动不起来；

如果多个元素都使用这个动画函数，每次都要var声明定时器，我们可以给不同的元素使用不同的定时器（自己专门用自己的定时器）核心原理就是利用就是是一门动态语言，可以很方便的给当前对象添加属性。

1. 缓动动画

让元素的运动速度有所变化，最常见的是让速度慢慢停下来。

原理：让盒子每次移动的距离慢慢变小，速度就会慢慢落下来；核心算法：（目标值-现在的位置）/10作为每次移动的距离（每次进入定时器都需要计算，所以写在定时器里面）；停止的条件让当前的盒子位置等于目标位置就停止定时器。注意如果出现小数会出现除不尽的情况，就会停的不准确 利用Math.ceil进行右取整

1. 动画函数添加回调函数

回调函数原理就是函数可以作为一个参数，将这个函数作为参数传递到另一个函数里面，当那个函数执行完之后，在执行传进去的这个函数这个过程就叫做回调；

回调函数写的位置：定时器结束的位置。

动画函数封装：因为会经常使用这个动画函数，可以单独封装到一个js文件里面，使用的时候可以直接引用这个js文件即可（单独新建一个js文件）

1. 轮播图

有定位才能使用动画函数，

功能需求：鼠标经过轮播图模块，左右按钮显示，离开隐藏左右按钮

点击右侧按钮一次，图片往右播放一张 20

图片的播放，下面小圆圈模块跟随一起变化

点击小圆圈，可以播放相应的图片

鼠标不经过轮播图，轮播图也会自动播放图片

鼠标经过，轮播图模块自动停止

点击小圆圈滚动图片：用到animate动画函数，将js文件引入，使用动画函数前提该元素必须有定位，注意是ul移动，而不是li；滚动元素的核心算法点击某个小圆圈，小圆圈的索引号乘以图片的宽度作为ul移动距离。

图片无缝滚动原理：把ul的第一个li复制一份，放到ul的最后面，当图片滚动到克隆的最后一张图片是，让ul快速的不做动画的调到最左侧：left为0，同时num赋值为0，可以重新开始滚动图片了。

克隆第一张图片：克隆ul第一个li cloneNode（）加true深克隆复制里面的子节点false浅克隆，克隆完成添加到ul最后面appendChild。

1. 滚动窗口至文档中的特定位置

Window.scroll(x,y)

页面滚动多少距离可以通过window.pageYOffset得到

1. 筋斗云案例

利用动画函数做动画效果，原先筋斗云的起始位置是0，鼠标经过某个小li把当前小li的offsetleft位置作为目标值即可；鼠标离开某个小li，就把目标值设为0 ；如果点击了某个小li就把当前的位置存储起来，作为筋斗云的起始位置