**3、zepto重点**

3、zepto 重点

(1)编译zepto模块

1.Zepto基础库不支持很多css选择器

比如很常用的$(":selected"),$("p:eq(1)"),$("li:first")这类，不过Zepto的库提供很多拓展的模块，你只需要在他的官网上按需要编译你需要的模块然后保存为zepto.js即可，或者直接使用在线编译，其中如果开启了selector模块，你就能支持大部分的css选择器了。

2.如果你开启了detect模块，那么你就可以用Zepto判断用户设备、操作系统和浏览器的功能(测试了几个还算可以用，不知是否准确)

3.如果开启了form模块，就可以对你的表单进行数据序列化，类似jquery的jquery form插件

4.如果开启了touch模块，你就可以使用tap(触屏点击) 和 swipe(触屏滑动),类似Jquery mobile 插件

============================================================

(2)设备检查

代码有两个主要的分支，针对IE和Webkit内核的浏览器。在IE浏览器中，MSPointerDown事件不会自动阻止鼠标事件，因此需要通过event.pointerType来判断指针类型。event.pointerType是一个枚举变量，一共有三个值：

MSPOINTER\_TYPE\_TOUCH = 2

MSPOINTER\_TYPE\_PEN = 3

MSPOINTER\_TYPE\_MOUSE = 4

==================================================================================

(3)touch事件

兼容性问题：

1. Window.navigator.msPointerEnabled语句只会判断浏览器是否支持MSPointer相关的事件，而不会判断用户的设备是否支持触摸操作 。目前只有在IE10上该对象不会返回undefined，其它版本的浏览器均视该对象不存在。如果你想判断用户的设备是否支持触摸操作，应 该使用Window.navigator.msMaxTouchPoints，如果该对象存在并且返回的结果大于1，则表示设备支持触摸操作并且是支持多点触摸的。

2. 在IE中，MSPointer相关的事件只会在浏览器支持的时候被触发，如果页面元素同时还带有鼠标事件，则鼠标事件也会被同时触发。

3. Webkit内核的浏览器支持touchstart事件，MSPointer相关的事件在这些浏览器上被视为无效。通过e.preventDefault()语句可以阻止 鼠标默认行为，从而不让mouse hover事件触发。

===================================================================

(4)zepto动画

zepto的动画完全采用css3中的动画（/过度）来完成的未用JS做任何兼容实现；

**4、添加iScroll的内容(很重要)：实现区域滚动**

4、添加iScroll的内容(很重要)：实现区域滚动

iScroll的诞生，主要是因为早期的webkit内核浏览器没有一种原生支持固定高度的容器;

使用下面的CSS类可以简单的改变滚动条样式。

.iScrollHorizontalScrollbar，这个样式应用到横向滚动条的容器。这个元素实际上承载了滚动条指示器。

.iScrollVerticalScrollbar，和上面的样式类似，只不过适用于纵向滚动条容器。

.iScrollIndicator，真正的滚动条指示器。

.iScrollBothScrollbars，这个样式将在双向滚动条显示的情况下被加载到容器元素上。通常情况下其中一个（横向或者纵向）是可见的 ;

IScroll是一个类，每个需要使用滚动功能的区域均要进行初始化。每个页面上的iScroll实例数目在设备的CPU和内存能承受的范围内是 没有限制的。

销毁：在不需要使用iScoll的时候调用iScroll实例的公共方法destroy()可以释放一些内存。

**5、swiper内容：实现图片轮换，分组滑动展示等等**

Swiper常用于移动端网站的内容触摸滑动

Swiper是纯javascript打造的滑动特效插件，面向手机、平板电脑等移动终端。

鼠标在swiper 停止滑动，鼠标移出swiper 继续滑动

<script language="javascript">

var mySwiper = new Swiper( '.swiper-container' ,

{autoplay : 1000,}

)

$('.swiper-container').hover({function(){

mySwiper.lockSwipes();

},function(){

mySwiper.unlockSwipes();

};

})

额，不太确定，但是你说的应该是这样吧，但是如果你想做，鼠标经过自动播放就暂停的话，用的其实不是这两个。

<script language="javascript">

var mySwiper = new Swiper( '.swiper-container' ,

{autoplay : 1000,}

)

$('.swiper-container').hover({function(){

swiper.stopAutoplay();

},function(){

swiper.startAutoplay();

};

})

=============================================

6、Android 4.2- 版本的CSS，JS兼容问题 点透问题

Android2.3的overflow问题

在android2.3及以下系统版本的浏览器不支持overflow：scroll / auto，即在页面元素里面的内容如果超过了父元素或祖先元素的高度是无法滚动的，可以通过css3的transform来实现滚动，代表作：iscroll。

Android2.3的聚焦问题

Android2.3会出现点击聚焦的情况，出现黄色的边框，通过对相应的元素设置css –webkit-tap-highlight-color: rgba(0,0,0,0)解决，另外一些高版本的系统也可能会出现有黄色边框的情况，这时候试着用outline：none来解决。

Android2.3 position:fixed及input中文输入bug

在android2.3系统中，如果页面中有position为fixed的元素，input外层的元素定位为relative或者自身的定位为relative，就会产生一种奇怪的现象，在input中输入数字或者字母都很正常，当切换输入法为中文后，发现无法输入，当把position设置为absolute后一切正常，所有当遇到这种情况时就只能改变布局了。

Android2.3 border-radius问题

不支持百分比，可用一个比较大的值 如： 9999px。

Android2.3 动画问题

@-webkit-keyframes wave1{0%{ opacity:0;}8.33%{ opacity:1;}99%{opacity:1;}100%{}}

100%要留一帧空出来。在100%之前写一个最后值 99%那里写。2.3系统的动画才会动。

Android2.3渐变问题

渐变这样写2.3才兼容：

background:-webkit-gradient(linear,00,0100%, color-stop(1%,rgba(111,191,7,0)), color-stop(74%,rgba(111,191,7,0)), color-stop(75%,rgba(111,191,7,0.9)), color-stop(100%,rgba(111,191,7,0.9)));

Linear是指渐变方式，0 0指从哪个方向开始。0 0 到 0 100% 说明是从上到下渐变。 后面的color-stop 可以指定百分比和颜色值。

Android4.0及低版本系统touch事件bug

在android4.0系统中，存在部分子版本的系统有touch事件的bug，表现为touchstart是正常的，touchmove只有在touchstart触发的时候触发一次，touchend不会被触发，android2.3会触发startmove一次，当touch结束后会执行剩余的move和end。在zepto库中此问题也没有得到解决,在android4.2系统中也存在这样的问题，后来发现是浏览器将事件cancel调了，目的应该是防止在滚动页面的时候调用，所以在touchmove事件中添加e.preventDefault()，发现touchend被触发了，所以解决方案根据实际使用场景来进行处理。

Android4.X placeholder问题

placeholder 文字字体会偏上。 但是placeholder只能设置他的color。如：

input::-webkit-input-placeholder{

color:red;}

行高不能设置。这时候把line-height设成normal 就能上下居中了。

各个系统对position:fixed的支持情况

手机Safari – iOS5及以后版本都支持. iOS4及以下版本不支持，会当做static处理.

Android

Android 2.1 及以下版本不支持.

Android 2.2 滚动过程中不保持fixed的位置，滚动完成后回到fixed的位置.

Android 2.3 支持fixed, 但是要求页面禁止缩放.

Android 3和4支持fixed.

无刷新改变当前页面url的问题整理

Android 2.3系统存在的问题：

无刷新改变当前页面url。不支持onpopstate方法，可用onhashchange方法代替。

虽然支持history的pushState方法和window.onpopstate方法。但是按回退键，即发生history.back()的时候，不会触发window.onpopstate方法。只会触发hashchange事件。

浏览器对popstate事件的整体支持情况：

history的pushState可操作浏览器历史，并且改变当前页面的URL。添加对应的url到历史记录里。但是用pushState方法会产生历史记录，可用replaceState方法代替，不会产生额外的历史记录。

hashchange 和popstate事件 触发的时候，hash已经改变了，无法阻止默认行为。即e.preventDefault()不起作用。

pushState()方法永远不会触发hashchange事件

popstate方法的触发条件：

\* 页面前进、后退时触发。

\* Hashchange时触发

5. 如何跳转

\* 如果history堆栈里有pushState过，那么就回跳到pushState所定的hash页面。如window.history.pushState(‘object’, document.title, “#explorer”);的#explorer页面

\* 没有pushState过，就按照history的记录进行回退和前进。)

==============================================================

\*\*\*tap事件冲突问题\*\*\*

\*\*\*placeholder 字数长的处理\*\*\*

====================================================================

多行文本溢出的处理

……

(一般的文字截断（适用于内联与块）：

div css xhtml xml example Source Code example Source Code

.text-overflow {

display:block;/\*内联对象需加\*/

width:31em;

word-break:keep-all;/\* 不换行 \*/

white-space:nowrap;/\* 不换行 \*/

overflow:hidden;/\* 内容超出宽度时隐藏超出部分的内容 \*/

text-overflow:ellipsis;/\* 当对象内文本溢出时显示省略标记(...) ；需与overflow:hidden;一起使用。\*/

})