1. 网络api地址网站 https://www.bootcdn.cn/
2. Android与ios兼容问题：

ios 设备 new Date() 只能用 18/01/01 这种格式

* + 1. android下没该格式要求
    2. 兼容问题列表:https://www.cnblogs.com/stoneniqiu/p/6077112.html （未测试过）

1. Ios下滚动条卡顿：

-webkit-overflow-scrolling:touch; 开启硬件加速 解决ios滑动卡顿现象 未测试过

1. Js问题:
   1. 获取地址栏后的参数：

function getQueryString(name) {

var reg = new RegExp('(^|&)' + name + '=([^&]\*)(&|$)', 'i');

var r = window.location.search.substr(1).match(reg);

if(r != null) {

return unescape(r[2]);

}

return null;

}

window.location.search 该方法用于截取？后的所有参数

通过上面方法可获去参数的值

1. 时间戳改为正常时间：

time(n) {

var time = new Date(n).getTime();

var year = new Date(n).getFullYear();

var mon = new Date(n).getMonth() + 1;

mon = mon >= 10 ? mon : '0' + mon;

var day = new Date(n).getDate();

day = day >= 10 ? day : '0' + day;

var hour = new Date(n).getHours() < 10 ? "0" + new Date(n).getHours() : new Date(n).getHours();

var min = new Date(n).getMinutes() < 10 ? "0" + new Date(n).getMinutes() : new Date(n).getMinutes();

var seconds = new Date(n).getSeconds() < 10 ? "0" + new Date(n).getSeconds() : new Date(n).getSeconds()

var str = year + '-' + mon + '-' + day;

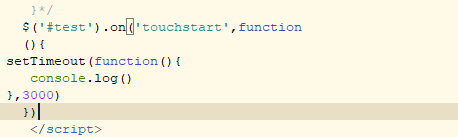
return str;

},

1. 跳转问题：

window.location.origin 只显示地址不显示后面的参数

window.location.href 显示地址和参数

1. 长按效果实现：
2. 
3. 原生js判断是否有class：

zixuanList[i].classList.contains('zixuan-select')

1. 判断android或ios

var u = navigator.userAgent,

isAndroid = u.indexOf('Android') > -1 || u.indexOf('Adr') > -1,

isIOS = !!u.match(/\(i[^;]+;( U;)? CPU.+Mac OS X/);

console.log(isAndroid,isIOS)

script标签参数

基本参数：src==路径[charset](http://www.w3school.com.cn/tags/att_script_charset.asp)==‘utf-8’

特殊参数: [defer](http://www.w3school.com.cn/tags/att_script_defer.asp)==’defer’ 页面加载完在加载js

[async](http://www.w3school.com.cn/tags/att_script_async.asp" \o "HTML5 <script> async 属性) 异步 js文件之间没有先后顺序是使用

10 循环

for of 只能输出数组的项 不能输出对象的项

for in 输出数组或对象的下标

for each 只能输出对象的项；

Object.keys(\_this.form).forEach(function(key2){})

中止foreach return false; breack报错

11 判断数据类型

1. 普通版 typeof 缺点：判断不出数组与对象 均为object
2. 高级版instanceof 缺点 死板不灵活必须自己指定类型 通过返回值判断

例子：

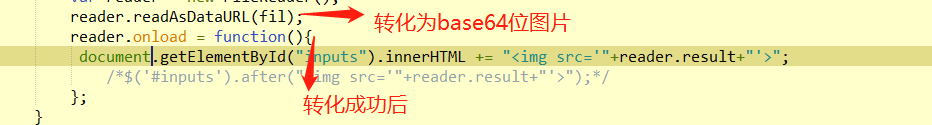
3、进阶版Object.prototype.toString.call(aa) 能判断数组与对象 数组时==【object array】 对象时==【object object】

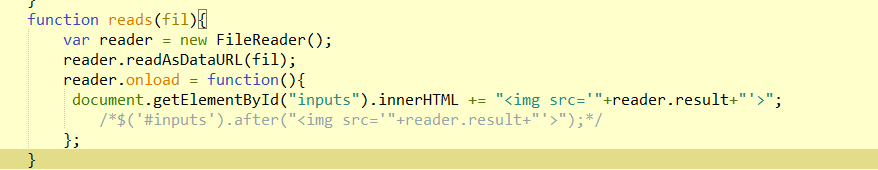
获取类型值 Object.prototype.toString.call(aa).split(' ')[1].replace(/\]/g,'')

结果未（Array,String）

12 读取input选中的文件

通过new FileReader(); 该方法能读取到图片 并将其转化为base64位的图片





13 console属性

地址<http://www.alloyteam.com/2013/11/console-log/>



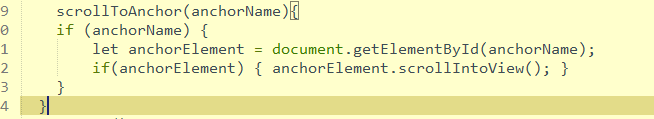
14、js锚点链接

方法一 通过id与href实现

<div id=’box’></div>

<a href=’/#box’></a>

方法二 通过scrollIntoView()实现



15、拼接字符串传自身；

‘<a onclick=”aa(this)”>asd</a>’

16promise使用

var promise = new Promise(function (resolve, reject) {

if (a>5) {resolve();}})

Promise.then(()=>{xxxxx})