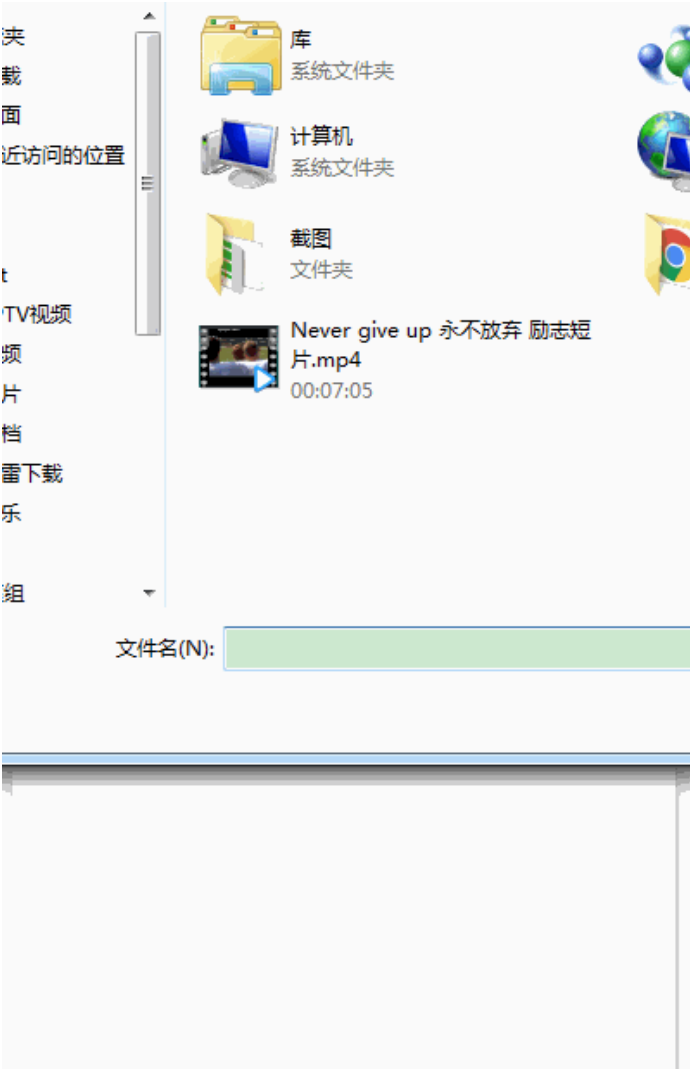


微信小程序把玩《十二》： Video API, Storage API, location API

一： Video API



电脑端不能测试拍摄功能只能测试选择视频功能，好像只支持 mp4 格式，值得注意的是成功之后返回的临时文件路径是个列表 **tempFilePaths** 而不是 **tempFilePath** 文档写的有点问题。

主要属性：

wx.chooseVideo(object)

参数	类型	必填	说明
sourceType	StringArray	否	"album"从相册选视频，"camera"使用相机拍摄，默认为：['album', 'camera']
maxDuration	Number	否	拍摄视频最长拍摄时间，单位秒。最长支持60秒
camera	StringArray	否	前置或者后置摄像头，默认为前后都有，即：['front', 'back']
success	Function	否	接口调用成功，返回视频文件的临时文件路径，res={tempFilePath: 视频文件的临时路径}
fail	Function	否	接口调用失败的回调函数
complete	Function	否	接口调用结束的回调函数（调用成功、失败都会执行）

成功之后返回参数

参数	说明
 tempFilePath tempFilePaths	选定视频的临时文件路径
duration	选定视频的时间长度
size	选定视频的数据量大小
height	返回选定视频的长
width	返回选定视频的宽

wxml

```
<button type="primary" bindtap="listenerBtnOpenVideo">打开视频</button>
```

<!--默认视频组件是隐藏的-->

```
<video src="{{videoSource}}" hidden="{{videoHidden}}" style="width: 100%; height: 100%" />
```

js

```
Page({  
  data:{  
  
    // text:"这是一个页面"
```

```
    videoSource: '',
    videoHidden: true
  },

  listenerBtnOpenVideo: function() {
    var that = this;
    wx.chooseVideo({

      //相机和相册

      sourceType: ['album', 'camera'],

      //录制视频最大时长

      maxDuration: 60,

      //摄像头

      camera: ['front', 'back'],

      //这里返回的是 tempFilePaths 并不是 tempFilePath

      success: function(res) {
        console.log(res.tempFilePaths[0])
        that.setData({
          videoSource: res.tempFilePaths[0],
          videoHidden: false
        })
      },
      fail: function(e) {
        console.log(e)
      }
    })
  },

  onLoad:function(options){
```

```
        // 页面初始化 options 为页面跳转所带来的参数
    },
    onReady:function() {

        // 页面渲染完成

    },
    onShow:function() {

        // 页面显示

    },
    onHide:function() {

        // 页面隐藏

    },
    onUnload:function() {

        // 页面关闭

    }
})
```

二: Storage API



其实这个存储在新建 Demo 的时候就已经用到了就是存储就是那个 logs 日志，数据存储主要分为同步和异步

异步存储方法：

存数据

wx.setStorage(object) 相同 key 会覆盖，可写回调方法

参数	类型	必填	说明
key	String	是	本地缓存中的指定的key
data	Object/String	是	需要存储的内容

获取方法：
wx.getStorage(object)

参数	类型	必填	说明
key	String	是	本地缓存中的指定的key
success	Function	是	接口调用的回调函数,res={data:"key对应的内容"}

清除方法：
wx.clearStorage()里面可以写回调函数 成功，失败，完成

同步存储方法：

存数据 相同 key 会覆盖
wx.setStorageSync(key,data)

读数据
wx.getStorageSync(key) 存储是指定的 key

清除数据

wx.clearStorageSync() 不可写回调方法

wxml

<!--动态获取数据-->

<text>{{storageContent}}</text>

<!--存-->

<button type="primary" bindtap="listenerStorageSave">storage 存储信息会在 text 上显示</button>

<!--取-->

<button type="primary" bindtap="listenerStorageGet">获取 storage 存储的信息</button>

<!--清-->

<button type="warn" bindtap="listenerStorageClear">清楚异步存储数据</button>

<text>{{storageSyncContent}}</text>

<button type="primary" bindtap="listenerStorageSyncSave">storageSync 存储信息会在 text 上显示</button>

<button type="primary" bindtap="listenerStorageSyncGet">获取 storageSync 存储信息</button>

<button type="warn" bindtap="listenerStorageSyncClear">清除同步存储数据</button>

js

```
Page({
  data:{

    // text:"这是一个页面"

    storageContent: '',
    storageSyncContent: ''
  },
  onLoad:function(options){

    // 页面初始化 options 为页面跳转所带来的参数

  },
  /**
   * 异步存储
   */
  listenerStorageSave: function() {

    //以键值对的形式存储 传进去的是个对象

    wx.setStorage({
      key: 'key',

      data: '我是 storeage 异步存储的信息',

      success: function(res) {
        console.log(res)
      }
    })
  },
  /**
   * 异步取信息
   */
})
```

```
listenerStorageGet: function() {
  var that = this;
  wx.getStorage({

    //获取数据的 key

    key: 'key',
    success: function(res) {
      console.log(res)
      that.setData({
        //
        storageContent: res.data
      })
    },
    /**

    * 失败会调用

    */
    fail: function(res) {
      console.log(res)
    }
  })
},

/**

* 清除数据

*/
listenerStorageClear: function() {
  var that = this;
  wx.clearStorage({
    success: function(res) {
      that.setData({
        storageContent: ''
      })
    }
  })
},
```

```
    })  
  }  
})  
,
```

```
/**
```

```
 * 数据同步存储
```

```
 */
```

```
listenerStorageSyncSave: function() {  
  wx.setStorageSync('key', '我是同步存储的数据')  
},
```

```
/**
```

```
 * 数据同步获取
```

```
 */
```

```
listenerStorageSyncGet: function() {  
  // var that = this;  
  var value = wx.getStorageSync('key')  
  this.setData({  
   StorageSyncContent: value  
  })  
},
```

```
/**
```

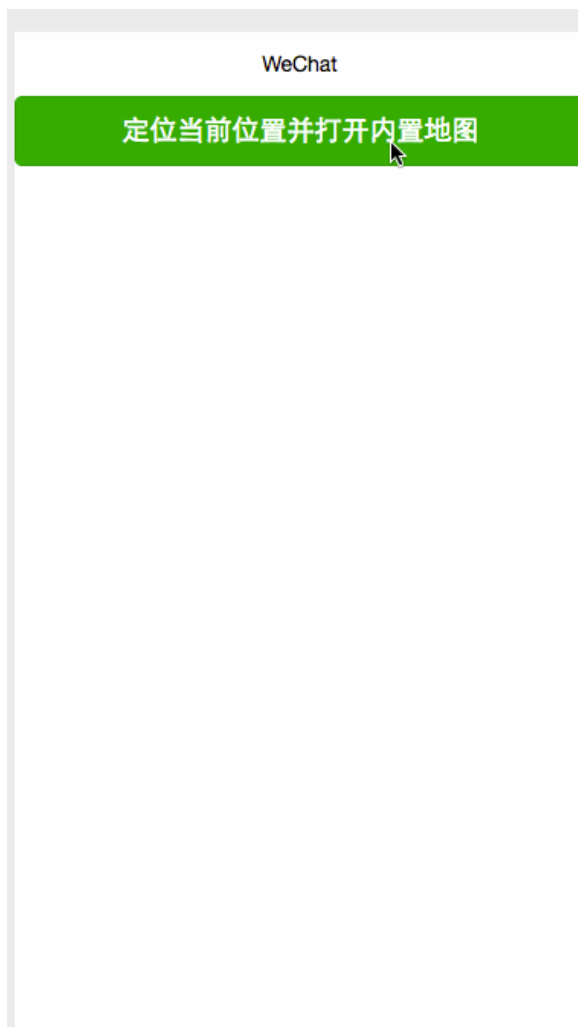
```
 * 清除同步存储数据
```

```
 */
```

```
listenerStorageSyncClear: function() {  
  wx.clearStorageSync()
```

```
},  
  
onReady:function() {  
    // 页面渲染完成  
},  
onShow:function() {  
    // 页面显示  
},  
onHide:function() {  
    // 页面隐藏  
},  
onUnload:function() {  
    // 页面关闭  
}  
})
```

三: location API



location API 也就分这里分两种 `wx.getLocation(object)`获取当前位置和 `wx.openLocation(object)`通过经纬度打开内置地图。其中定位获取位置信息返回参数是有问题的 `speed`，`accuracy` 这两个是没有的。还有一个就是打开内置地图之后再返回会报一个错误(Page route 错误—WASERVICE.js:2 navigateBack 一个不存在的 webviewId0)如果有知道的请告知，我找到解决方式也会补充下！

主要属性：

`wx.getLocation(object)`获取当前位置

参数	类型	必填	说明
type	String	否	默认为"wgs84"返回gps坐标, "gcj02"返回可用于wx.openLocation的坐标
success	Function	是	成功获取地理位置的回调
fail	Function	否	接口调用失败的回调函数
complete	Function	否	接口调用结束的回调函数（调用成功、失败都会执行）

- 成功之后返回参数

参数	说明
latitude	纬度, 浮点数, 范围为-90~90, 负数表示南纬
longitude	经度, 浮点数, 范围为-180~180, 负数表示西经
speed	速度, 浮点数, 单位m/s 没返回
accuracy	位置的精确度 没返回
success	Function
fail	Function
complete	Function

wx.openLocation(object)打开微信内置地图

参数	类型	必填	说明
latitude	Float	是	纬度， 范围为-90~90， 负数表示南纬
longitude	Float	是	经度， 范围为-180~180， 负数表示西经
scale	INT	否	缩放比例， 范围1~28， 默认为28
infoUrl	String	否	在查看位置界面底部显示的超链接,可点击跳转
name	String	否	位置名
address	String	否	地址的详细说明
success	Function	否	接口调用成功的回调函数
fail	Function	否	接口调用失败的回调函数

这里直接进入微信内置应用，当使用导航返回键时是内部写的外界无法干预所以 **WAService.js:2 navigateBack** 一个不存在的 **webViewId0** 这个错估计也带等小程序修复吧！！

wxml

```
<button id="0" type="primary" bindtap="listenerBtnGetLocation">定位当前位置并打开内置地图</button>
```

js

```
Page ( {
```

```
data:{
  text:"Page location"
},
onLoad:function(options){

  // 页面初始化 options 为页面跳转所带来的参数
},

/**
 * 监听定位到当前位置
 */
listenerBtnGetLocation: function() {
  wx.getLocation({

    //定位类型 wgs84, gcj02
    type: 'gcj02',
    success: function(res) {
      console.log(res)
      wx.openLocation({

        //当前经纬度
        latitude: res.latutude,
        longitude: res.longitude,

        //缩放级别默认 28
        scale: 28,

        //位置名
        name: '测试地址',

        //详细地址
```



```
address: '火星路 24 号',

//成功打印信息

success: function(res) {
  console.log(res)
},

//失败打印信息

fail: function(err) {
  console.log(err)
},

//完成打印信息

complete: function(info) {
  console.log(info)
},
})

},
fail: function(err) {
  console.log(err)
},
complete: function(info) {
  console.log(info)
},
})
},

onReady:function() {

  // 页面渲染完成

},
```

```
onShow:function(){  
    // 页面显示  
},  
onHide:function(){  
    // 页面隐藏  
},  
onUnload:function(){  
    // 页面关闭  
}  
})
```