其实webview加载资源的速度并不慢，但是如果资源多了，当然就很慢。图片、 css 、 js 、 html 这些资源每个大概需要 10-200ms ，一般都是 30ms 就 ok 了。不过webview是必须等到全部资源都完成加载，才会进行渲染的，所以加载的速度很重要！从Google上我们了解到，webview 加载页面的顺序是：先加载 html ，然后从里面解析出 css 、 js 文件和页面上的图片资源进行加载。如果 webkit 的缓存里面有，就不加载。加载完这些资源之后，就进行 css 的渲染和 js 的执行。 Css 的渲染一般不需要很长时间，几十毫秒就 ok 。关键是 js 的执行，如果用了 jQuery ，则执行起来需要 5-6 秒。而在这段时间，如果不在 webview 里设置背景，网页部分是白色的，很难看。这是一个很糟糕的用户体验。所以如果用网页布局程序，最好别用那些庞大的 js 框架。最好使用原生的js写业务脚本，以提升加载速度，改善用户体验。

  Android 的 webview 是基于 webkit 内核的，webview 中集成了js与java互调的接口函数，通过addJavascriptInterface方法，可以将Java的类注册进webkit，给网页上的js进行调用，而且还可以通过loadUrl方法是给webkit传递一个uri，供浏览器来进行解析，实现Java和js交互。

  要想运行网页上的js脚本，webview必须设置支持Javascript

**Java代码  [收藏代码](javascript:void())**

1. mWebview.getSettings().setJavaScriptEnabled(**true**);

  然后是设置webview要加载的网页：

    web的网页：webView.loadUrl("http://www.google.com");

    本地的网页：webView.loadUrl("file:///android\_asset/XX.html");  //本地的存放在：assets文件夹中

  webview做完基本的初始化后我们还要要给它，加进一个回调的代理类JavaScriptInterface，并给它一个调用的名称：ncp

**Java代码  [收藏代码](javascript:void())**

1. mWebView.addJavascriptInterface(**new** JavaScriptInterface(), "ncp");

  JavaScriptInterface可以是一个普通的Java类，类实现的方法，均可被js回调：

**Java代码  [收藏代码](javascript:void())**

1. **final** **class** JavaScriptInterface {
3. **public** **int** callOnJs() {
5. **return** 1000;
6. }
8. **public** **void** callOnJs2(String mode) {
9. //TODO
10. }
12. }

  Java要调用js的方法，只需知道js的方法名称即可：

**Java代码  [收藏代码](javascript:void())**

1. mWebView.loadUrl("javascript:onSaveCallback()");

  js 这边就更简单：

**Js代码  [收藏代码](javascript:void())**

1. <script type="text/javascript">
2. window.onload = **function**(){
3. document.getElementById('btn\_1').addEventListener('click', onBtnClick, **false**);
4. **var** \_int\_value = window.ncp.callOnJs();
5. alert("get int from java:" + \_int\_value );
6. }
7. **function** onBtnClick() {
8. window.ncp.callOnJs2("click");
9. }
10. </script>

Java和js交互有以下一些特点：

 1．Java 调用 js 里面的函数，速度并不令人满意，大概一次一两百毫秒吧，如果要做交互性很强的事情，这种速度会让人疯掉的。而反过来就不一样了， js 去调 java 的方法，速度很快，基本上 40-50 毫秒一次。所以尽量用 js 调用 java 方法，而不是 java 去调用 js 函数。

 2．Java 调用 js 的函数，没有返回值，而 Js 调用 java 方法，可以有返回值。返回值可以是基本类型、字符串，也可以是对象。如果是字符串，有个很讨厌的问题，第 3 点我会讲的。如果是对象，这个对象会被转换为 js 的对象，直接可以访问里面的方法。但是我不推荐 java 返回给 js 的是对象，除非是必须。因为 js 收到 java 返回的对象，会产生一些交换对象，而如果这些对象的数量增加到了 500 或 600 以上，程序就会出问题。所以尽量返回基本数据类型或者字符串。

 3．Js 调用 Java 的方法，返回值如果是字符串，你会发现这个字符串是 native 的，不能对它进行一些修改操作，比如想对它 substr ，取不到。怎么解决呢？转成 locale 的。使用 toLocaleString() 函数就可以了。不过这个函数的速度并不快，转化的字符串如果很多，将会很耗费时间。