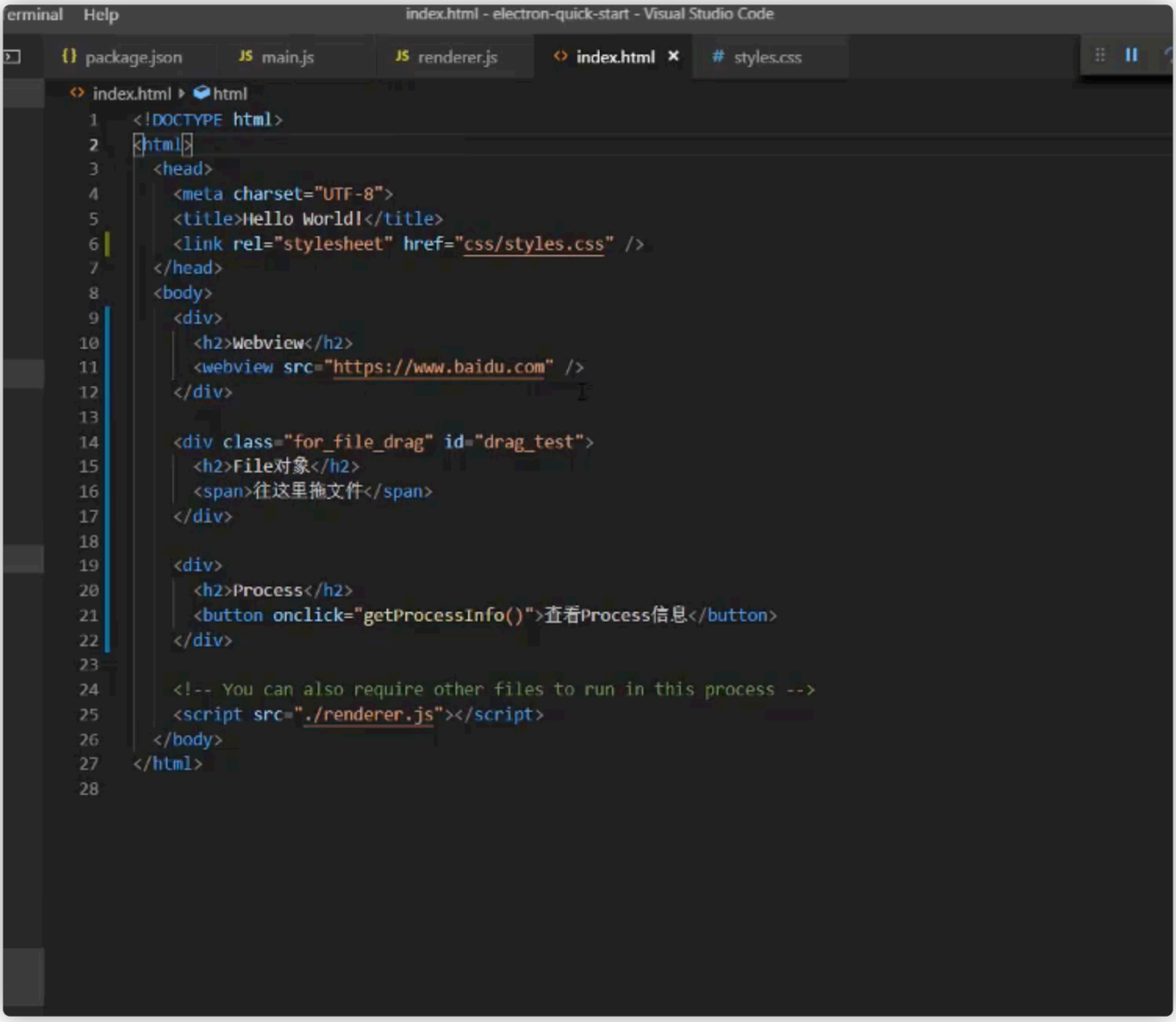
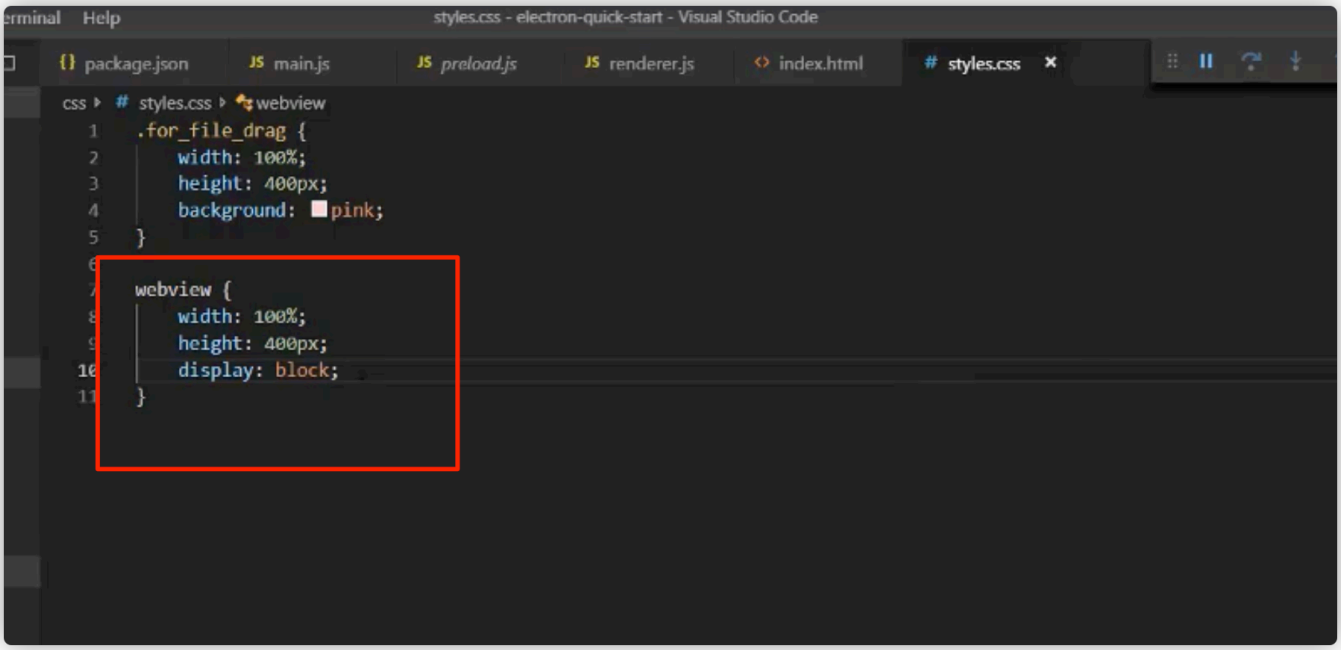
1.webview


lerplate
API 参考
简介
进程对象
支持的命令行开关
环境变量
Chrome 扩展支持
重要的API变更
自定义 DOM 元素
File 对象
<webview> 标签
window.open 函数
BrowserWindowProxy 对象

Main Process 模块

https://www.electronjs.org/docs/api/webview-tag

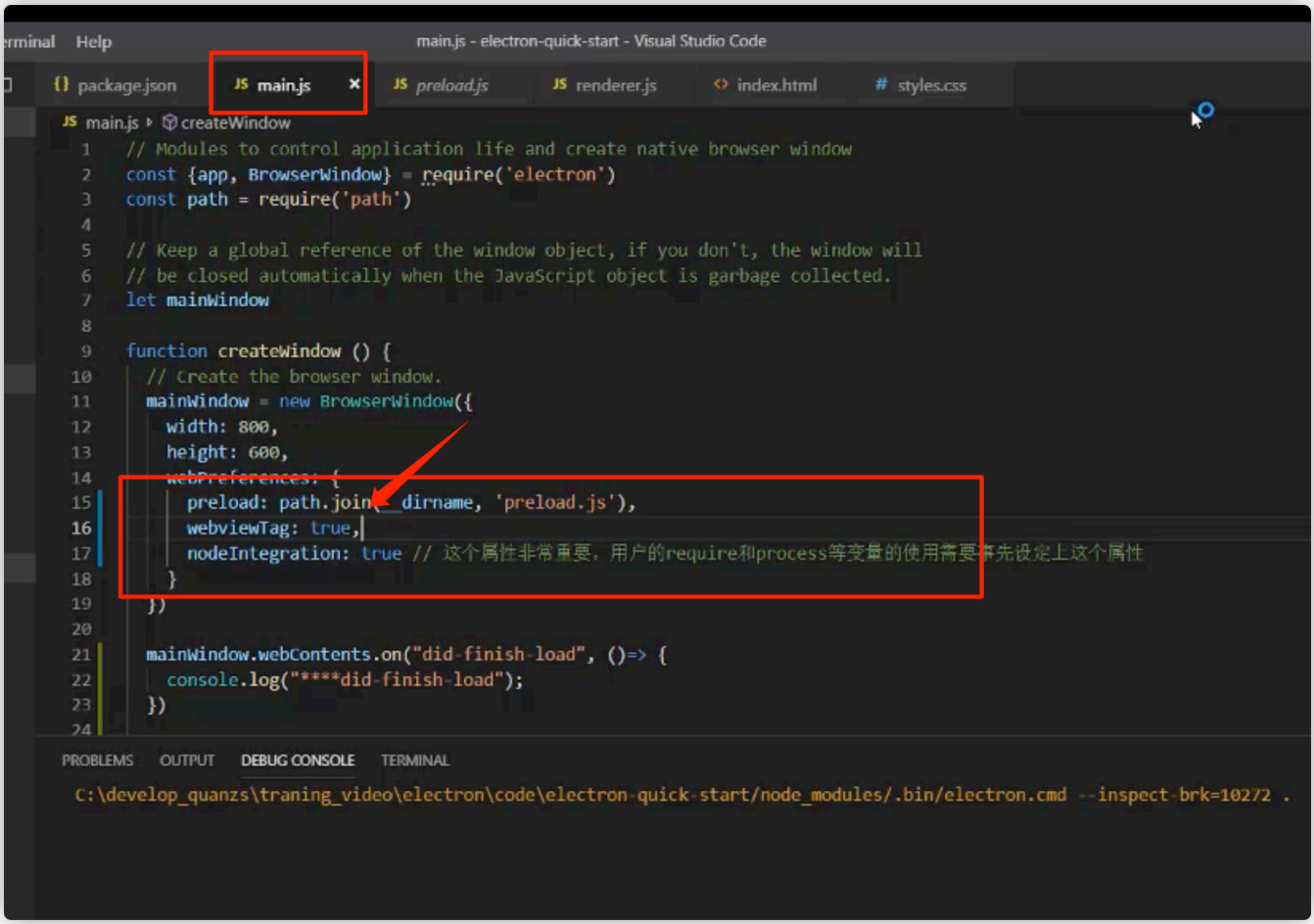




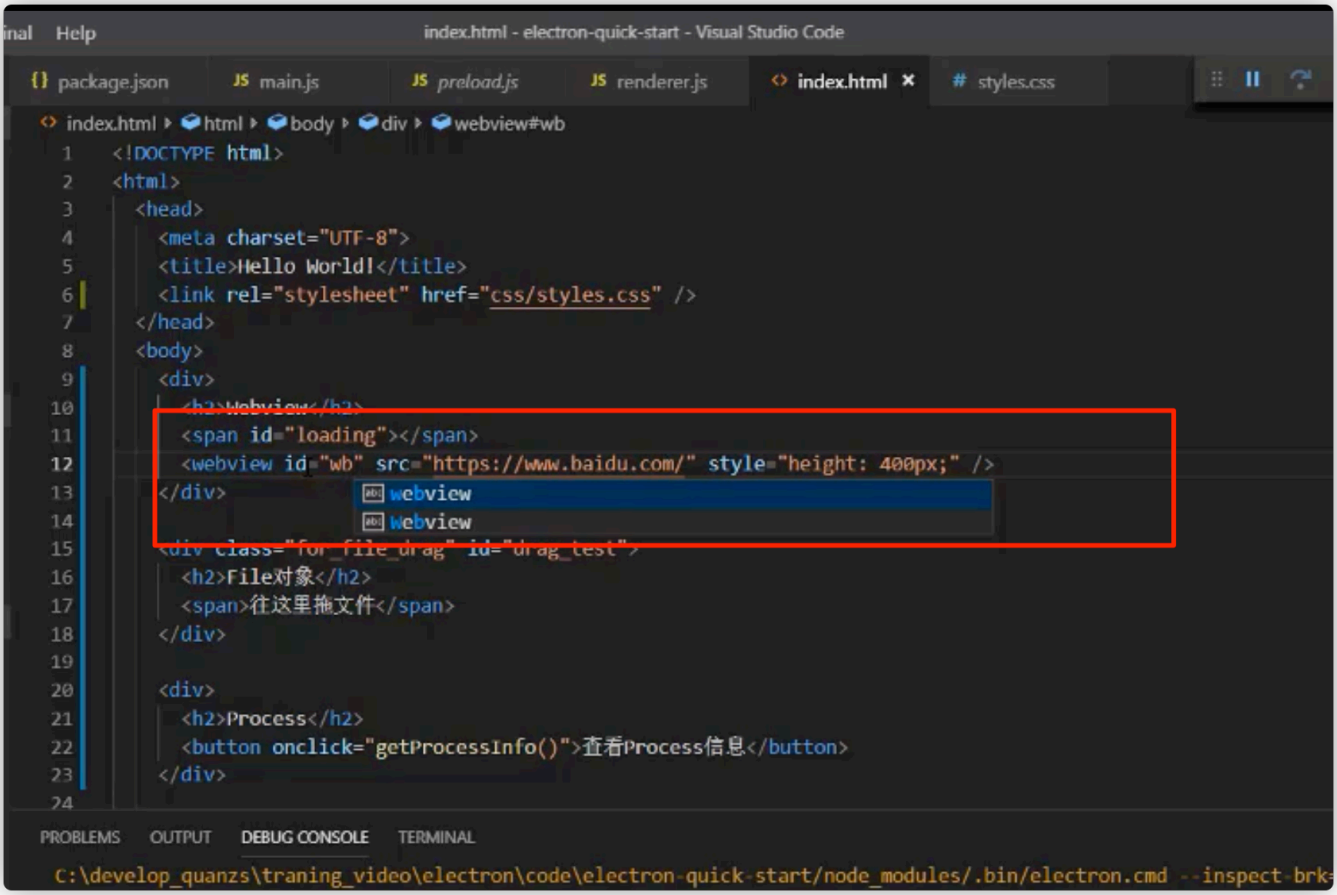
Enabling

EN

By default the `webview` tag is disabled in Electron >= 5. You need to enable the tag by setting the `webviewTag` webPreferences option when constructing your `BrowserWindow`. For more information see the BrowserWindow constructor docs.

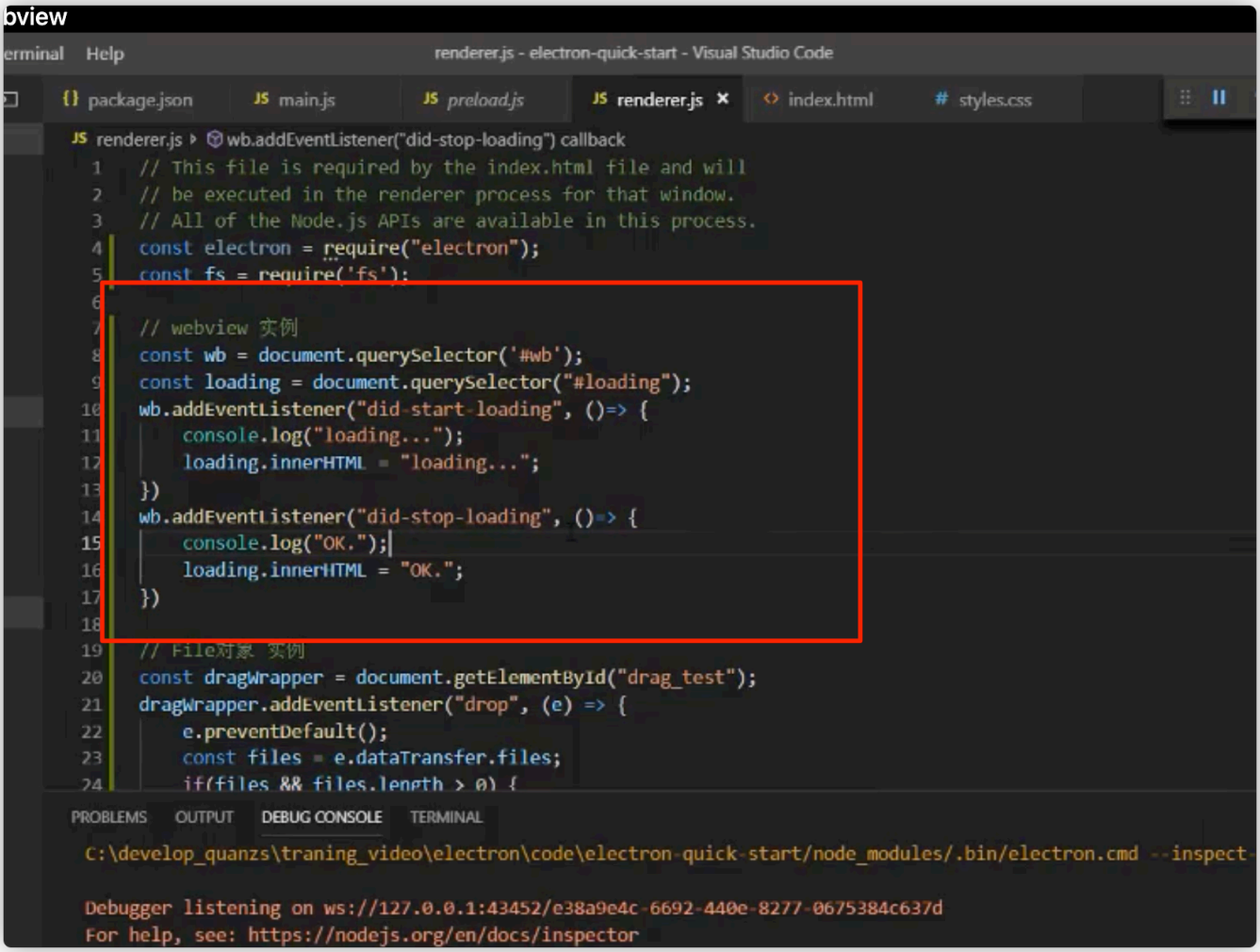



Hello World!
File Edit View Window Help
Webview
新闻 hao123 地图 视频 贴吧 学术 登录 设置


index.html - electron-quick-start - Visual Studio Code
package.json
main.js
preload.js
renderer.js
index.html
styles.css
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Hello World!</title>
<link rel="stylesheet" href="css/styles.css" />
</head>
<body>

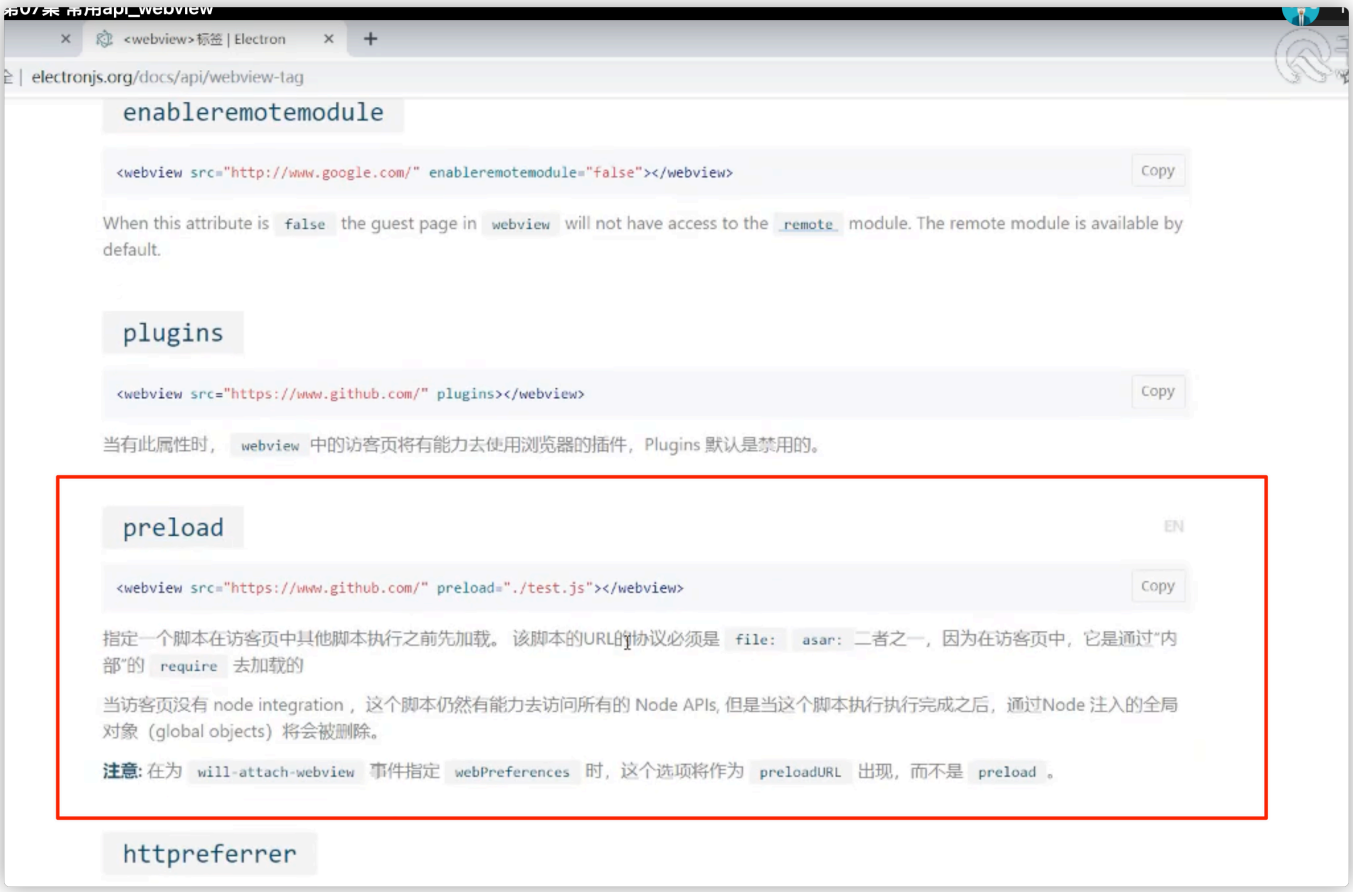
<webview id="wb" src="https://www.baidu.com/" style="height: 400px;" />

webview
webview
<h2>File对象</h2>
往这里拖文件
<h2>Process</h2>
<button onclick="getProcessInfo()">查看Process信息</button>
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
C:\develop_quanzs\traning_video\electron\code\electron-quick-start/node_modules/.bin/electron.cmd --inspect-brk

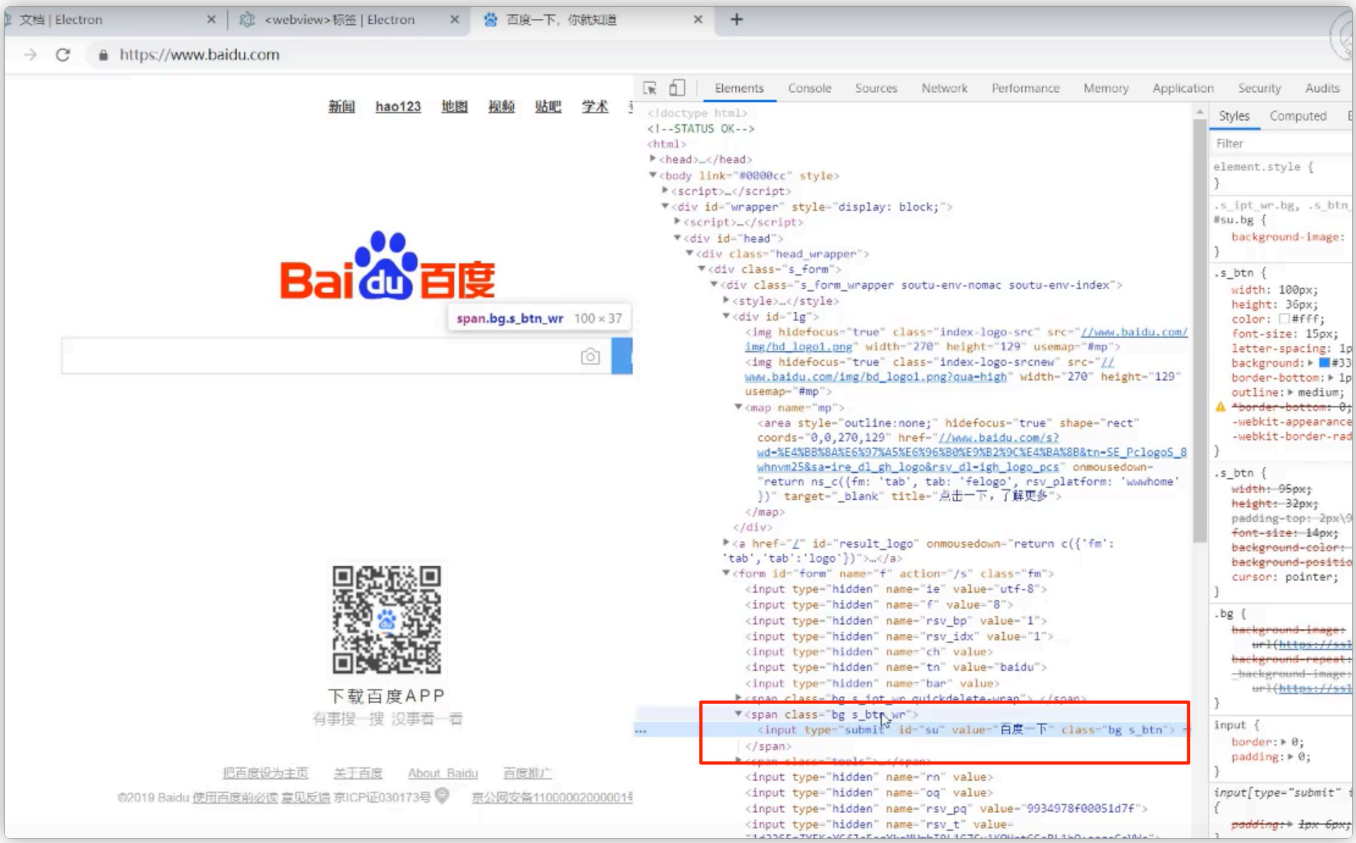


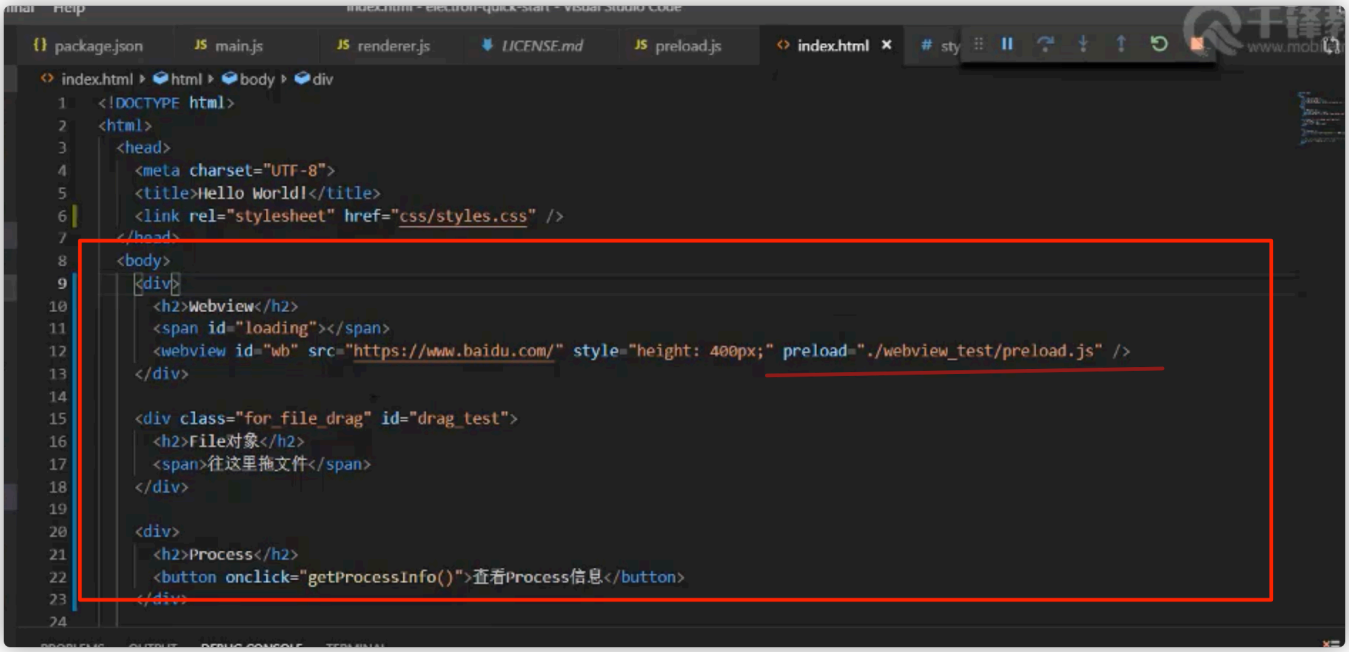

```
<script>
  onload = () => {
    const webview = document.querySelector('webview')
    const indicator = document.querySelector('.indicator')

    const loadstart = () => {
      indicator.innerText = 'loading...'
    }

    const loadstop = () => {
      indicator.innerText = ''
    }

    webview.addEventListener('did-start-loading', loadstart)
    webview.addEventListener('did-stop-loading', loadstop)
  }
</script>
```

preload



使用百度网址，使用preload在加载的时候一些事件或者脚本



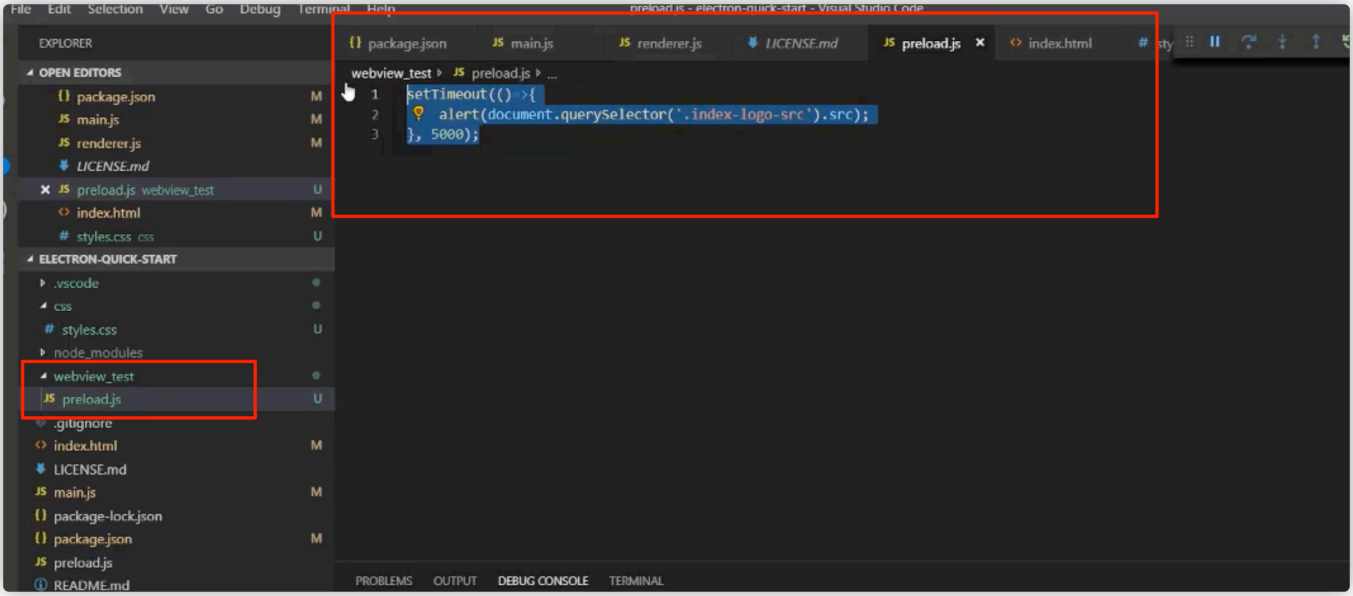

package.json
main.js
renderer.js
LICENSE.md
preload.js
index.html
index.html ▸ html ▸ body ▸ div
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Hello World!</title>
<link rel="stylesheet" href="css/styles.css" />
</head>
<body>

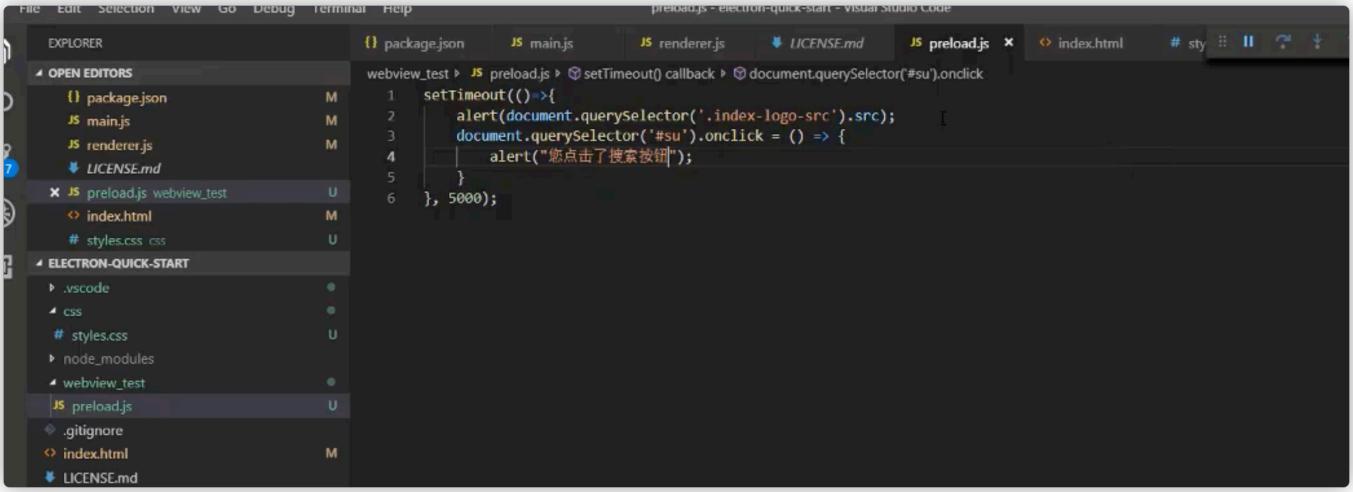
<h2>Webview</h2>

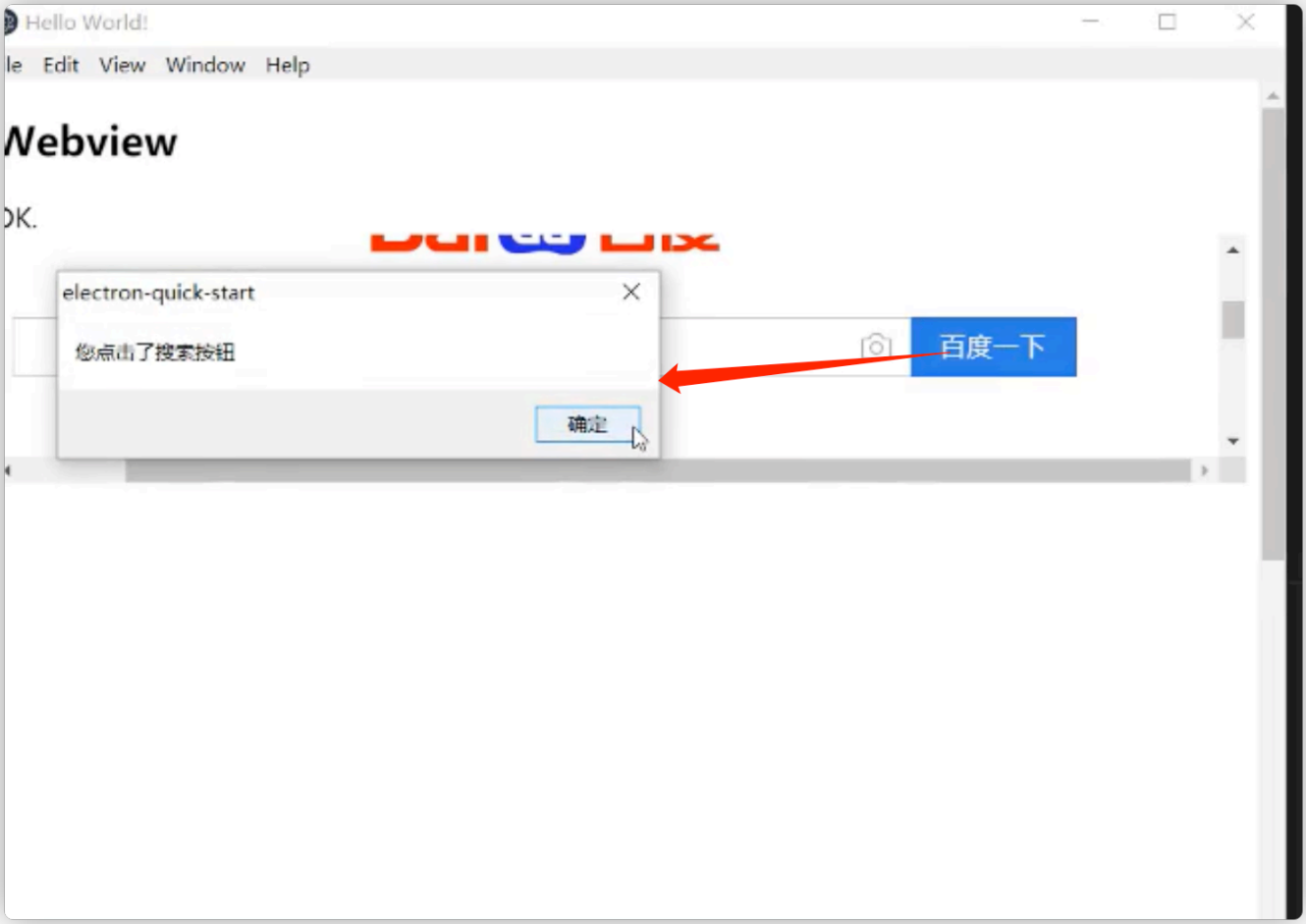
<webview id="wb" src="https://www.baidu.com/" style="height: 400px;" preload="./webview_test/preload.js" />

<h2>File对象</h2>
往这里拖文件

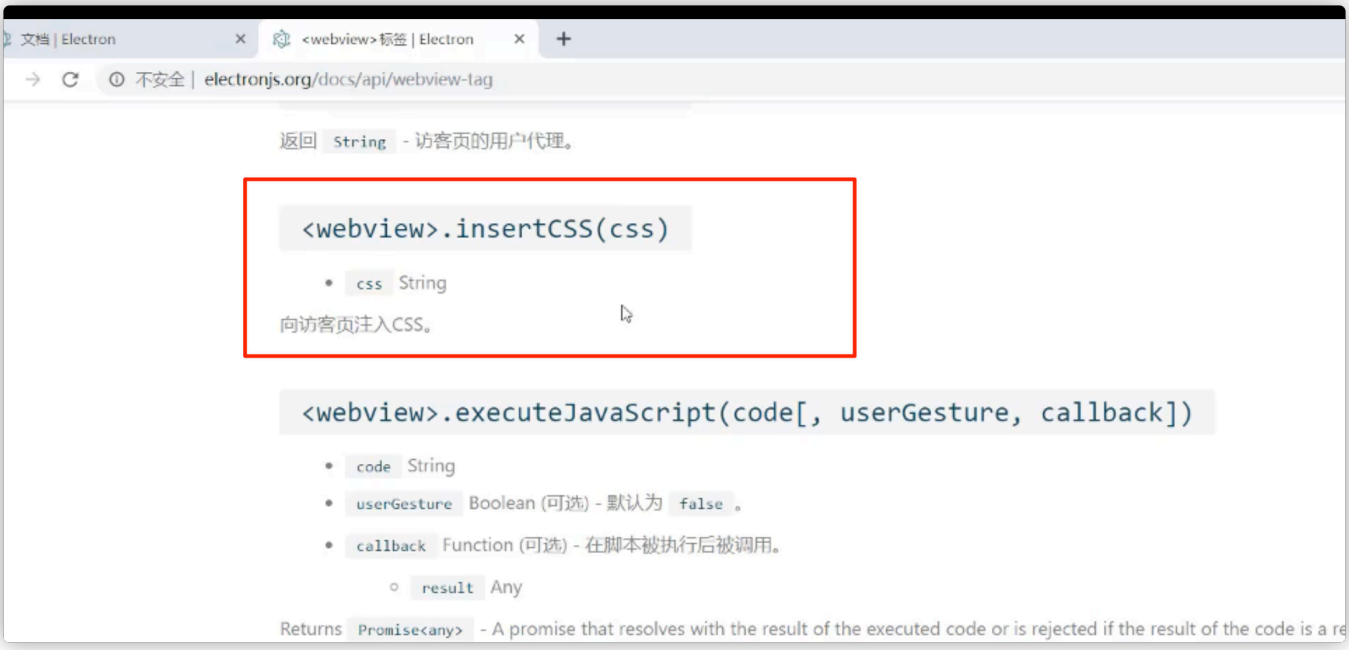
<h2>Process</h2>
<button onclick="getProcessInfo()">查看Process信息</button>

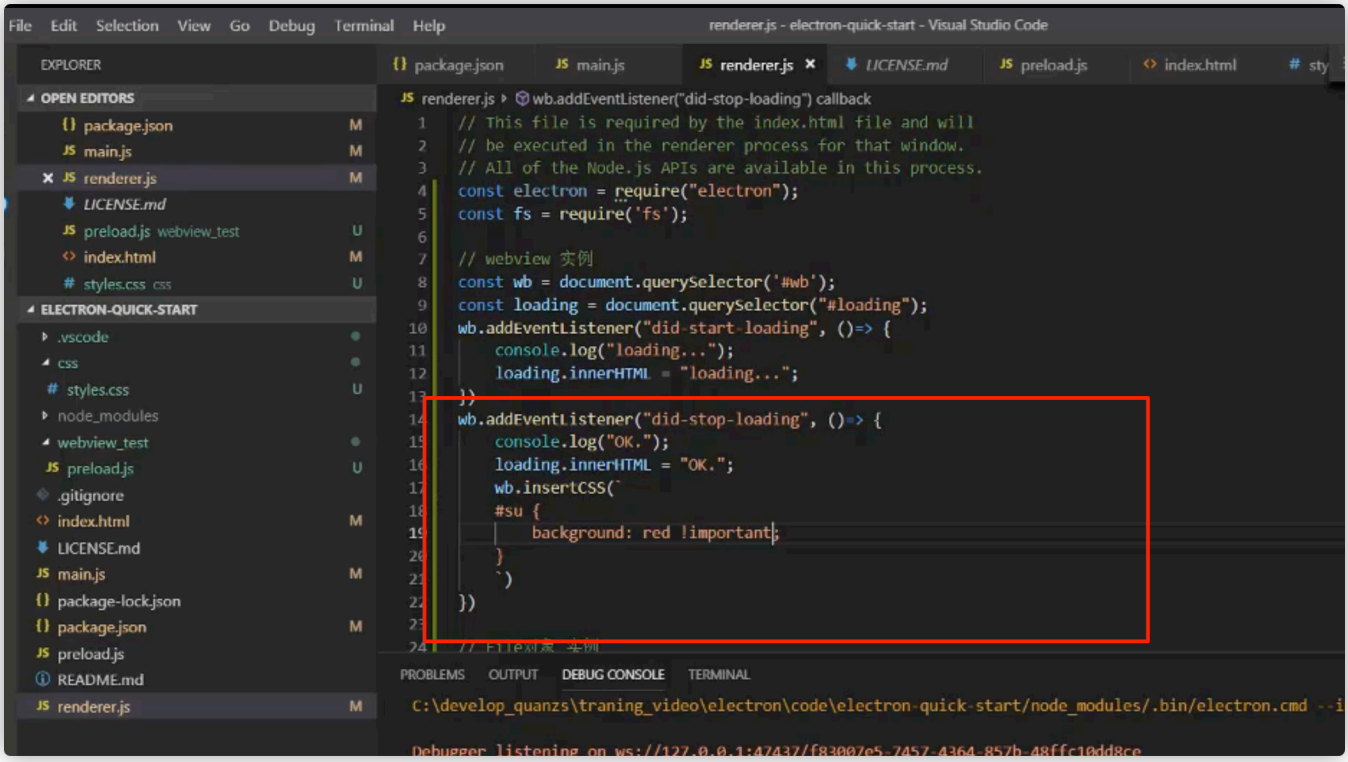

EXPLORER
OPEN EDITORS
package.json
main.js
renderer.js
LICENSE.md
preload.js webview_test
index.html
styles.css
ELECTRON-QUICK-START
.vscode
css
styles.css
node_modules
webview_test
preload.js
.gitignore
index.html
LICENSE.md
main.js
package-lock.json
package.json
preload.js
README.md
webview_test ▸ preload.js ▸ ...
setTimeout(()=>{
alert(document.querySelector('.index-logo-src').src);
}, 5000);
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL


EXPLORER
OPEN EDITORS
package.json
main.js
renderer.js
LICENSE.md
preload.js webview_test
index.html
styles.css
ELECTRON-QUICK-START
.vscode
css
styles.css
node_modules
webview_test
preload.js
.gitignore
index.html
LICENSE.md
webview_test ▸ preload.js ▸ setTimeout() callback ▸ document.querySelector('#su').onclick
setTimeout(()=>{
alert(document.querySelector('.index-logo-src').src);
document.querySelector('#su').onclick = () => {
alert("您点击了搜索按钮");
}
}, 5000);


Hello World!
le Edit View Window Help
Webview
electron-quick-start
您点击了搜索按钮
确定
百度一下

insertcss


文档 | Electron
<webview>标签 | Electron
不安全 | electronjs.org/docs/api/webview-tag
返回 String - 访客页的用户代理。
<webview>.insertCSS(css)
css String
向访客页注入CSS。
<webview>.executeJavaScript(code[, userGesture, callback])
code String
userGesture Boolean (可选) - 默认为 false 。
callback Function (可选) - 在脚本被执行后被调用。
result Any
Returns Promise<any> - A promise that resolves with the result of the executed code or is rejected if the result of the code is a re



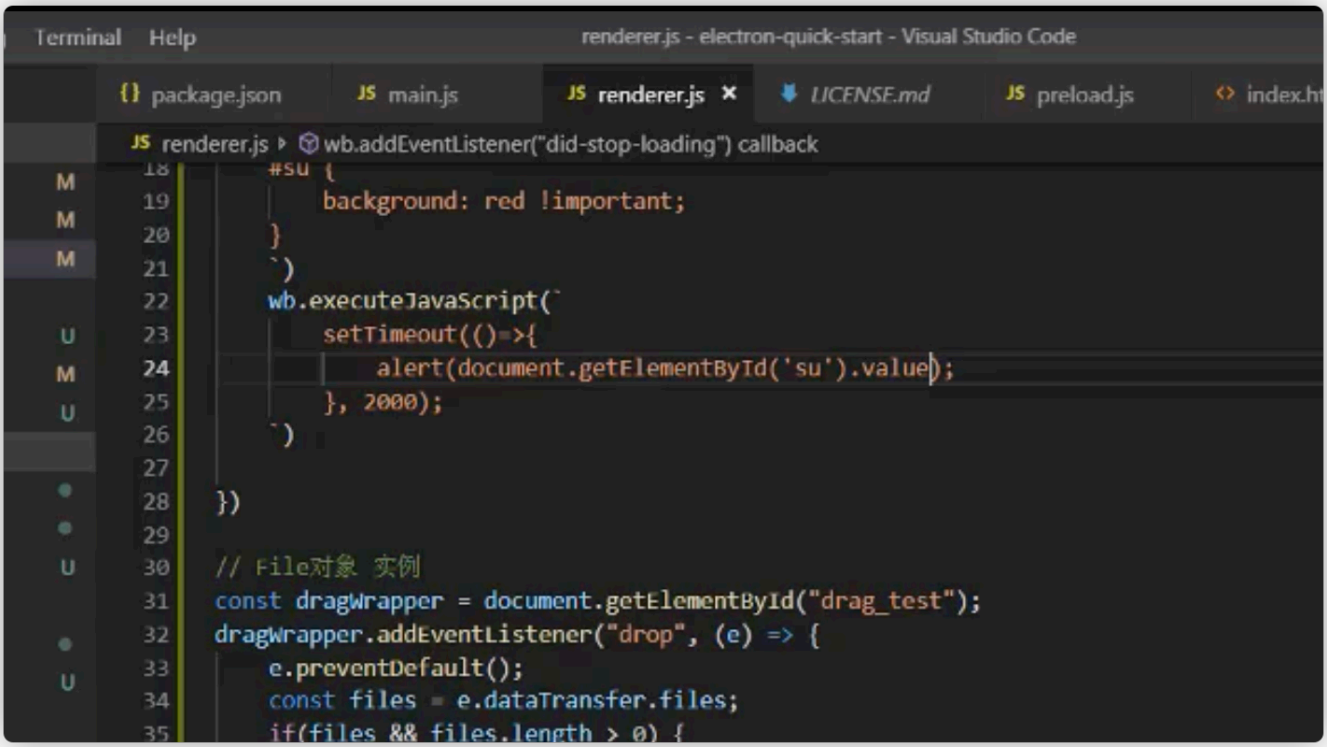
也可以注入js

`<webview>.executeJavaScript(code[, userGesture])`

- `code` String
- `userGesture` Boolean (可选) - 默认为 `false`。

Returns `Promise<any>` - A promise that resolves with the result of the executed code or is rejected if the result of the code is a rejected promise.

在页面中执行 `code`。 如果设置了 `userGesture`，它将在页面中创建用户手势上下文。 像 `requestFullScreen` 这样的需要用户操作的HTML API可以利用这个选项来实现自动化。



打开devtools

`<webview>.openDevTools()`

Opens a DevTools window for guest page.