# 常用快键键

ctrl+N 打开新窗口

ctrl+T 打开新标签页

ctrl+W 关闭当前标签页

ctrl+F4 关闭浏览器

ctrl+Tab 切换到下一个标签页

ctrl+shift+Tab 切换到上一个标签页

F5 刷新页面

ctrl+F5 强制刷新页面

F12 审查元素

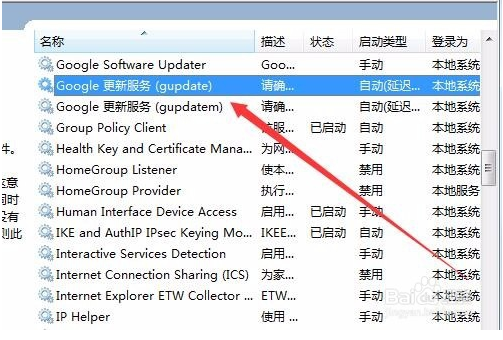
F10 设置断点后按此键可以一行一行执行js，在执行的过程中若遇到了一个js函数就可以按F11键跳到那个函数里面去

Ctrl+Shift+Delete:用于显示“清除浏览数据”界面

# 禁止自动更新

## 方法一

在打开的服务窗口中，在右侧服务中找到“Google更新服务（Gupdate）"服务 项与“Google更新服务（Gupdatem）”服务项



双击Google更新服务（Gupdate）项，在打开的窗口中点击启动类型下拉菜单，在弹出菜单中选择“禁用”菜单项，然后点击确定按钮



重新启动电脑后，chrome浏览器就不再自动升级了，删除安装根目录下的update文件夹。

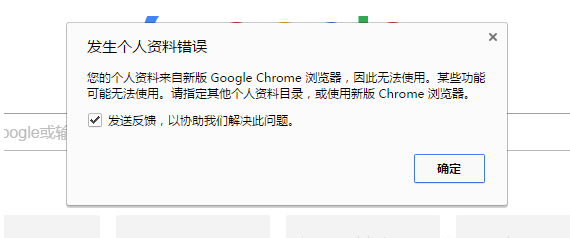
# 查看谷歌浏览器的版本

chrome://version

查看当前Chrome版本，切记不要点击帮助下面的“关于Google Chrome（G）”,否则就会自动升级。

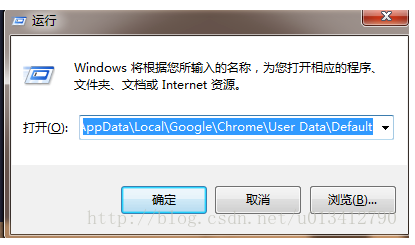
# 谷歌浏览器发生个人资料错误

谷歌浏览器升级的时候发生了这个错误



解决方法：

命令行：%USERPROFILE%\AppData\Local\Google\Chrome\User Data\Default



删掉Default文件夹下面的Web Data文件即可。

# F12开发者工具使用

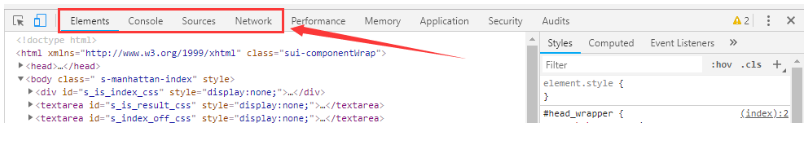
## 如何调出开发者工具

1、按F12调出

2、右键检查（或快捷键Ctrl+Shift+i）调出



## 开发者工具初步介绍



chrome开发者工具最常用的四个功能模块：元素（ELements）、控制台（Console）、源代码（Sources），网络（Network）

元素（Elements）：用于查看或修改HTML元素的属性、CSS属性、监听事件、断点等。css可以即时修改，即时显示。大大方便了开发者调试页面

控制台（Console）：控制台一般用于执行一次性代码，查看JavaScript对象，查看调试日志信息或异常信息。还可以当作Javascript API查看用。例如我想查看console都有哪些方法和属性，我可以直接在Console中输入"console"并执行~

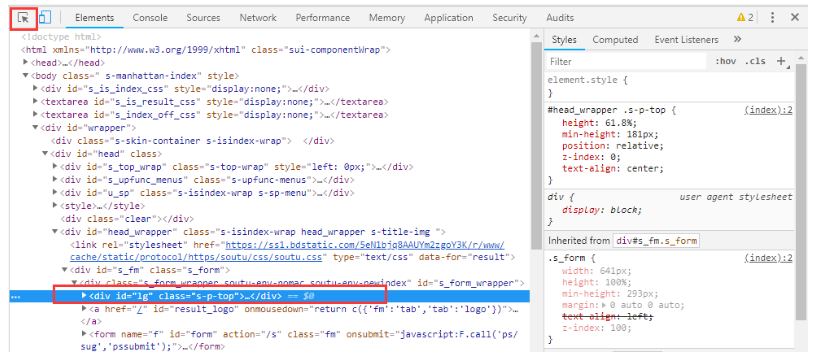
源代码（Sources）：该页面用于查看页面的HTML文件源代码、JavaScript源代码、CSS源代码，此外最重要的是可以调试JavaScript源代码，可以给JS代码添加断点等。

网络（Network）：网络页面主要用于查看header等与网络连接相关的信息。

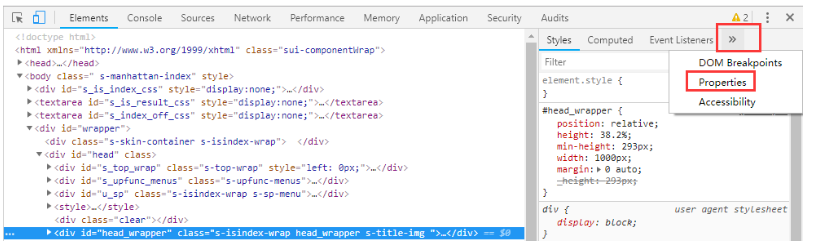
### 元素

查看元素的代码：点击左上角的箭头图标（或按快捷键Ctrl+Shift+C）进入选择元素模式，然后从页面中选择需要查看的元素，然后可以在开发者工具元素（Elements）一栏中定位到该元素源代码的具体位置。

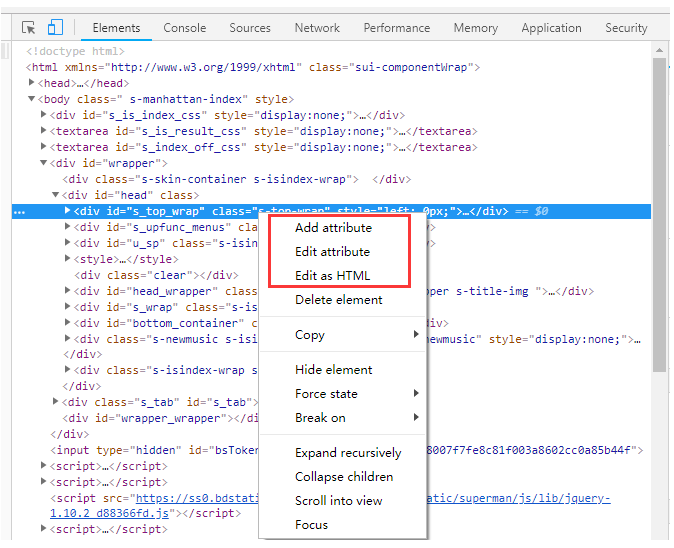
查看元素的属性：定位到元素的源代码之后，可以从源代码中读出改元素的属性。如下图中的class、src、width等属性的值。



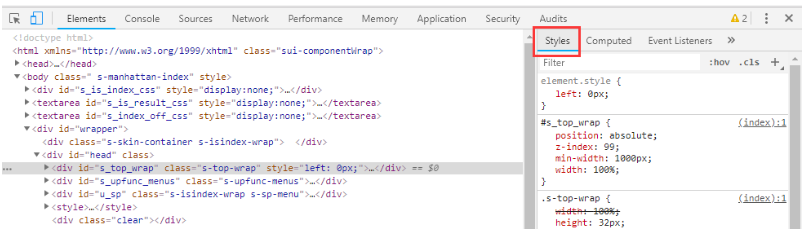
当然从源代码中读到的只是一部分显式声明的属性，要查看该元素的所有属性，可以在右边的侧栏中查看：



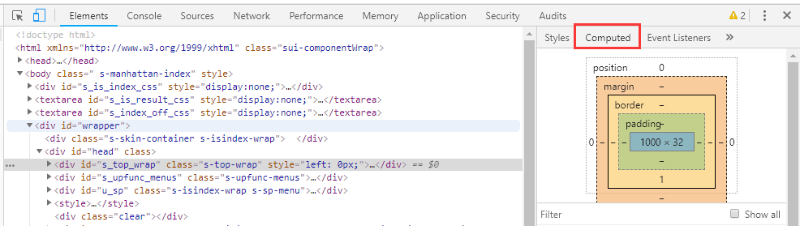
修改元素的代码与属性：点击元素，然查看右键菜单，可以看到chrome提供的可对元素进行的操作：包括编辑元素代码（Edit as HTML）、修改属性（Add attribute、Edit attribute）等。选择Edit as HTML选项时，元素进入编辑模式，可以对元素的代码进行任意的修改。当然，这个修改也仅对当前的页面渲染生效，不会修改服务器的源代码，故而这个功能也是作为调试页面效果而使用。



查看元素的CSS属性：在元素的右边栏中的styles页面可以查看该元素的CSS属性，这个页面展示该元素原始定义的CSS属性以及从父级元素继承的CSS属性。从这个页面还可以查到该元素的某个CSS特性来自于那个CSS文件，使编码调试时修改代码变得非常方便。

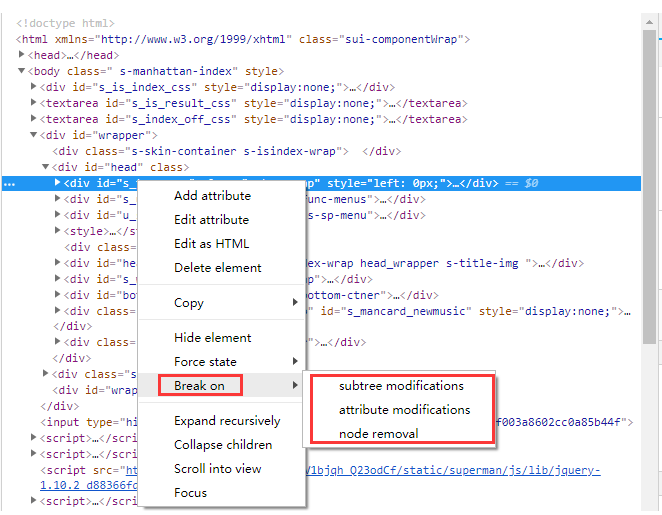


在Styles页旁边，有一个Computed页面，这个页面展示该元素经过计算之后的所有CSS属性，即最后浏览器渲染页面时使用的属性。属性的计算由浏览器自动进行，是浏览器渲染页面的一个必不可少的过程。



修改元素的CSS属性：在元素的Styles页面，可以对元素的CSS属性进行修改，甚至删除原有、添加新属性。不过这些修改，仅对当前浏览器的页面展示生效，不会修改CSS源代码。所以在这里进行CSS属性的修改一般用来调整和完善元素的渲染效果。

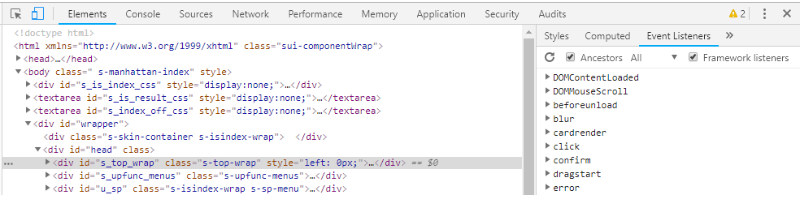
给元素添加断点：在元素的右键菜单中选择断点选项（Break on…），选中之后，当元素被修改（通常是被JS代码修改）时，页面加载会暂停，然后可以查看该元素的属性。



元素断点添加之后，可以在右侧栏的DOM Breakpoints页面中看到，这个页面可以看到当前网页的所有元素断点。



查看元素的监听事件：元素的右边栏的Event Listener页面，可以查看到该元素的所有监听事件。在开发中，尤其是维护其他人的代码时，会出现不了解元素对应的监听事件，这个时候，可以在这个页面中找到。这个页面不仅能看到对应的事件函数，还可以定位该函数所在的JS文件以及在该文件中的具体位置（行数），大大提高开发维护的效率。

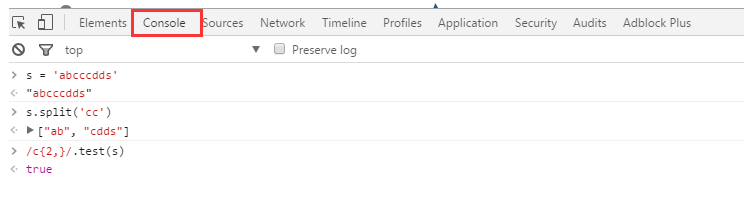


### 控制台

查看JS对象的及其属性



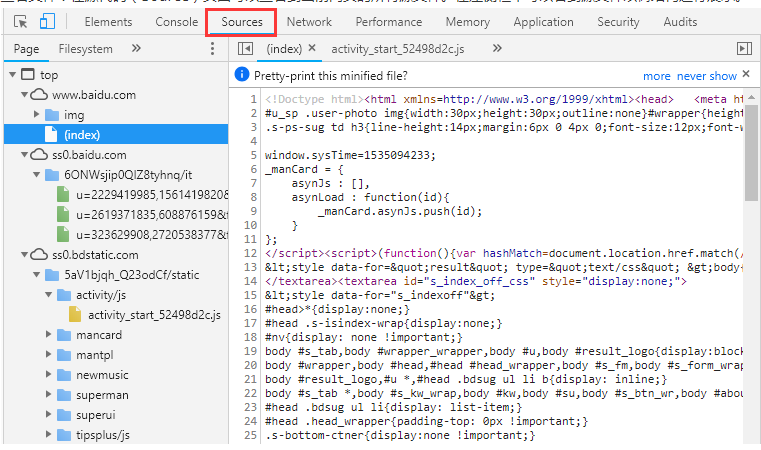
执行JS语句



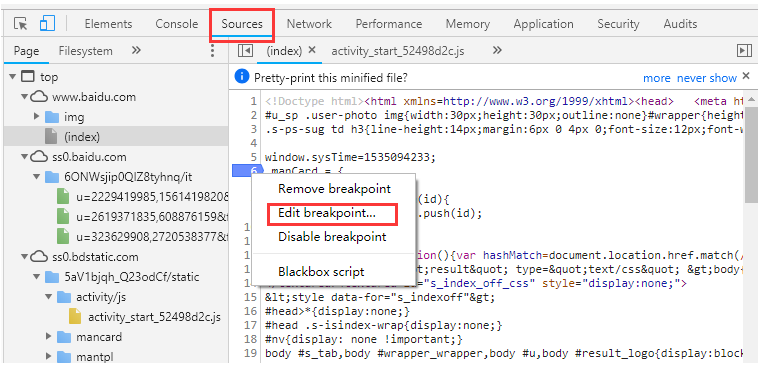
查看控制台日志：当网页的JS代码中使用了console.log()函数时，该函数输出的日志信息会在控制台中显示。日志信息一般在开发调试时启用，而当正式上线后，一般会将该函数去掉。

### 源代码

查看文件：在源代码（Source）页面可以查看到当前网页的所有源文件。在左侧栏中可以看到源文件以树结构进行展示。

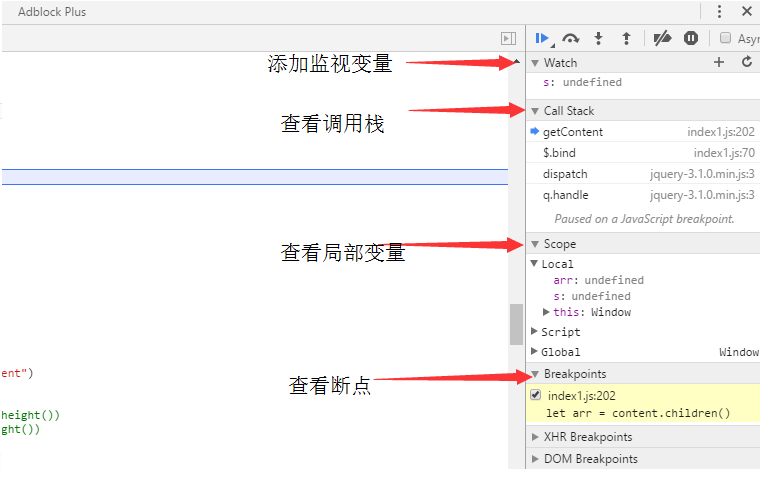


添加断点：在源代码左边有行号，点击对应行的行号，就好给改行添加上一个断点（再次点击可删除断点）。右键点击断点，在弹出的菜单中选择Edit breakpoint可以给该断点添加中断条件。

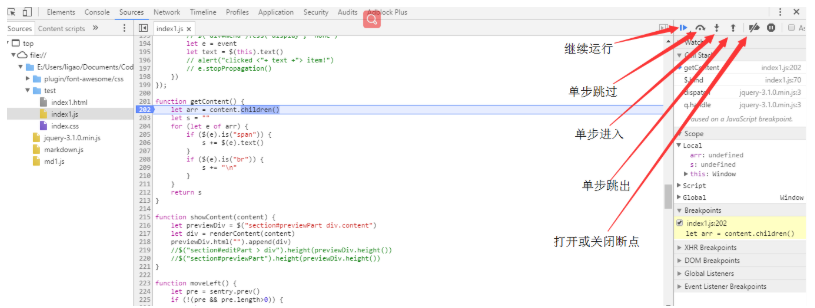


中断调试：添加断点后，当JS代码运行到断点时会中断（对于添加了中断条件的断点在符合条件时中断），此时可以将光标放在变量上查看变量的

也可以在右边的侧栏上查看

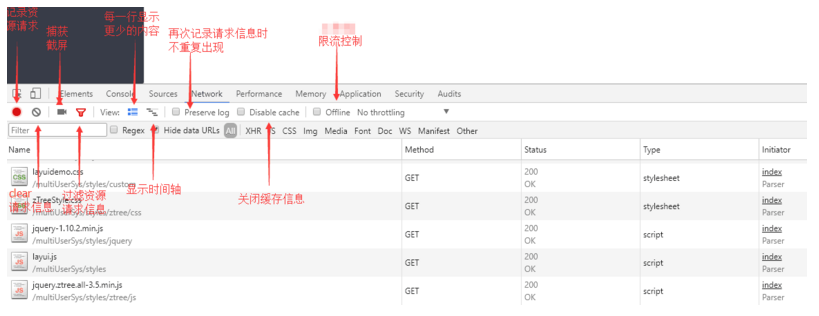


在右侧变量上方，有继续运行、单步跳过等按钮，可以在当前断点后，逐行运行代码，或者直接让其继续运行。



### 网络

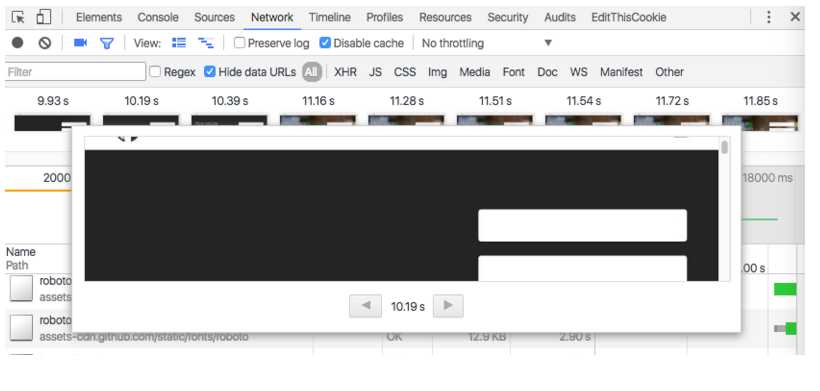
#### 基本概述



https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824162719196-1156474310.png：记录按钮 处于打开状态时会在此面板进行网络连接的信息记录，关闭后则不会记录。

https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824162746778-827981360.png：清除按钮 清除当前的网络连接记录信息。（点击一下就能清空）

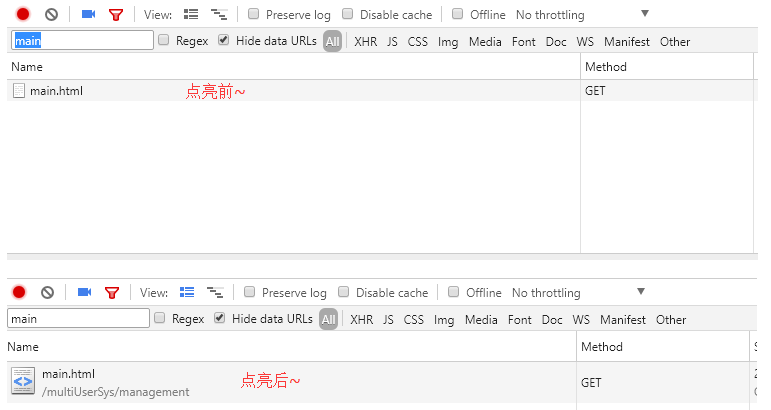
https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824162810119-979271078.png：捕获截屏 记录页面加载过程中一些时间点的页面渲染情况，截图根据可视窗口截取，如下图所示。



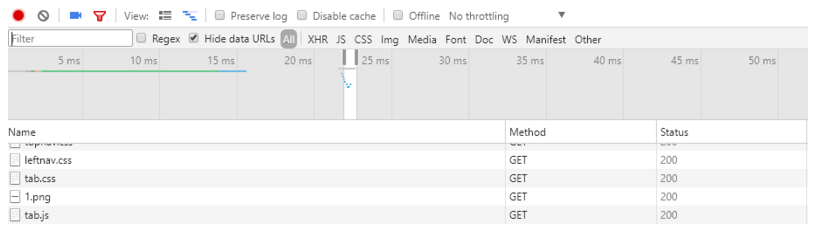
https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824162925039-1864034518.png：过滤器 能够自定义筛选条件，找到自己想要资源信息，如下图所示。



https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824163013483-1490909242.png：显示详细信息



https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824163121256-394789081.png：显示时间流



能够根据时间，查看对应时间下浏览器请求的资源信息

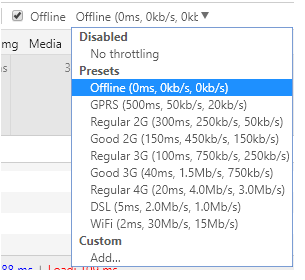
https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824163206011-209234690.png：是否保留日志

当选择保留日志，重新加载url当前界面时，之前请求显示的资源信息，会保留下来，不会清空的。

https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824163226893-1504363441.png：是否进行缓存

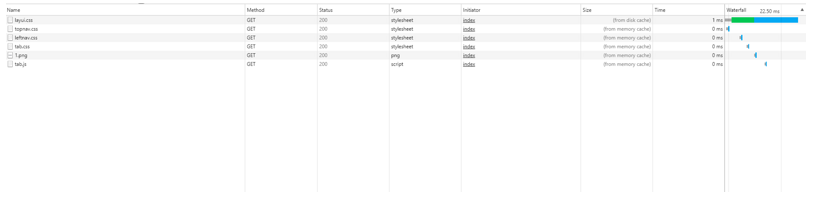
当打开开发者工具时生效，打开这个开关，则页面资源不会存入缓存，可以从Status栏的状态码看文件请求状态。

https://images2018.cnblogs.com/blog/1377691/201808/1377691-20180824163250374-973968006.png：设置模拟限速，如下图所示。



设置限速可以模拟处于各种网络环境下的不同用户访问本页面的情况。

#### 内容介绍



下列介绍中，前者为名词解释，后者为举例

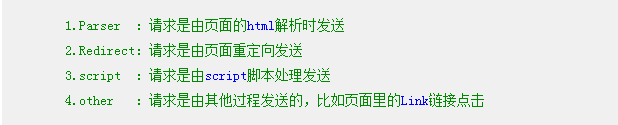
Name/Pat：资源名称以及URL路径 （main.css）

Method：Http请求方法 (GET或者POST)

status/Text：Http状态码/文字解释 （200，ok）

Type ：请求资源的MIME类型，MIME是Multipurpose Internet Mail Extensions (html,css,js等)

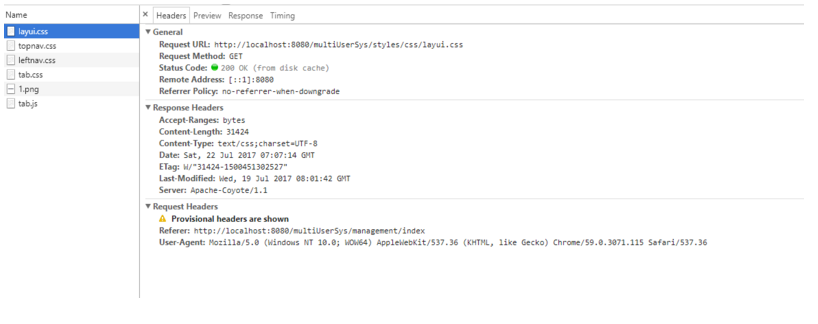
Initiator：解释请求是怎么发起的，有四种可能的值



size/content：size是响应头部和响应体结合的大小，content是请求解码后的大小

#### 请求文件具体说明

点击某个具体请求后的界面，如下图所示：



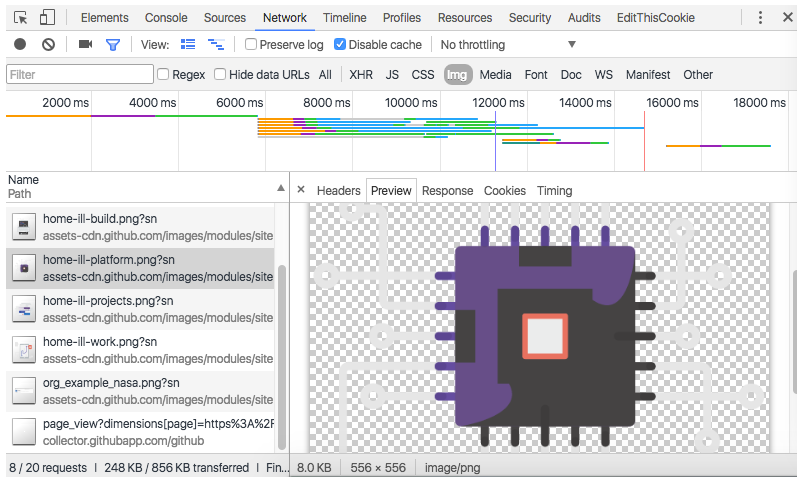
一共分为四个模块：

Headers



Header面板列出资源的请求url、HTTP方法、响应状态码、请求头和响应头及它们各自的值、请求参数等等

Preview：预览面板，用于资源的预览。



Response：响应信息面板包含资源还未进行格式处理的内容

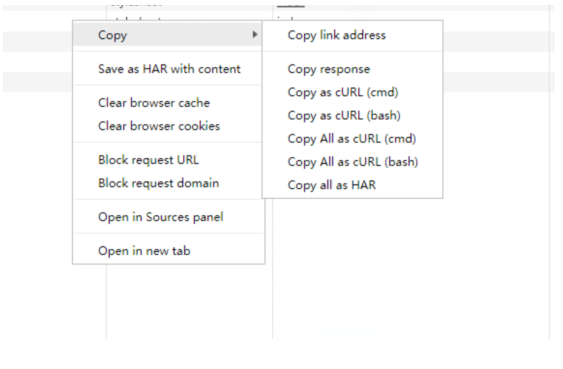


Timing：资源请求的详细信息花费时间



### 细节补充

对某请求右键，出现页面如下图所示



Copy Request Headers：复制HTTP请求头到系统剪贴板

Copy Response Headers：复制HTTP响应头到系统剪贴板

Copy Response：复制HTTP响应内容到系统剪贴板

Copy as cURL：将网络请求作为一个curl的命令字符复制到系统剪贴板(curl是一种开源的命令行工具和库，用于配合url语法进行数据传输)