ELEMENTS

* 检查和修改你的网页（[Inspect and Tweak Your Pages](https://developers.google.com/web/tools/chrome-devtools/iterate/inspect-styles/)）
  + 实时编辑DOM节点（Live-edit a DOM node）
    - 编辑DOM （Edit the DOM）
      * 检查元素（Edit the DOM）
        1. 点击检查元素按钮点击**检查元素**按钮检查图标进入检查元素模式，然后单击某个元素。
        2. 使用[inspect](https://developers.google.com/web/tools/chrome-devtools/debug/command-line/command-line-reference#inspect)方法在控制台中，如inspect(document.body)。
      * 导航DOM（Navigate the DOM）
        1. 控制面板主内容

折叠节点代表该节点包含折叠节点

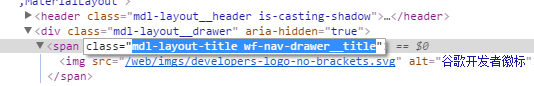
展开的节点扩展折叠节点的展示方式

* + - * 1. 控制面板底部

延长面包屑

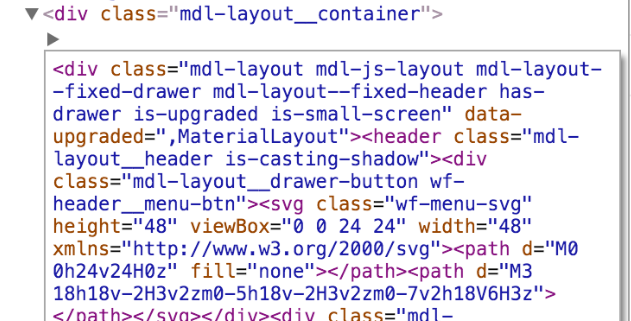
当前选择的节点以蓝色高亮显示。到左侧的节点是当前节点的父节点。再左边是父亲的父亲。等等，一路上涨的DOM树。

* + - * 编辑DOM节点和属性（Edit DOM nodes and attributes）
        1. 编辑DOM节点和属性



* + - * 1. 编辑DOM节点及其子女为HTML

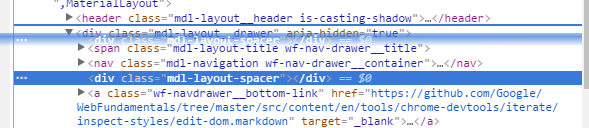
选中节点之后打开右键菜单再选择“Edit as HTML”



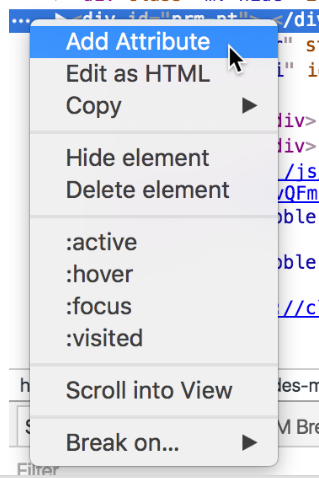
Ctrl + Enter : 保存

* + - * 移动DOM节点（Move DOM node）

选中节点，按住并拖动节点来移动它。



* + - * 删除DOM节点（Delete DOM node）
        1. 选中节点后打开右键操作菜单，然后选择“Delete Element”。
        2. 选择节点，然后按Delete键。
      * 右键操作菜单（Show more actions menu）



* + - * 滚动进入视野（Scroll into view）

有时在Element面板中选中的元素在页面的视窗上并没有展示，这就是需要Scroll into view了；

选中右键操作菜单的“Scroll into View“，页面视窗就会滚动到对应的节点。

* + - * 设置断点DOM（Set DOM breakpoints）
        1. 在特定的代码行设置一个断点（Set a breakpoint on a particular line of code）

设置断点的基础知识

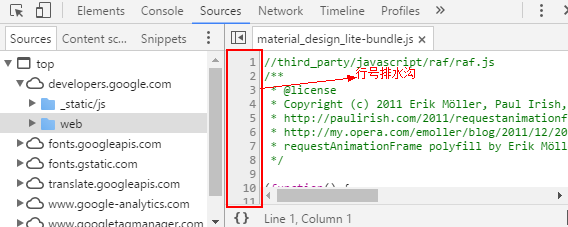
先打开”source ”面板，在左侧窗格中选择相应的脚本文件。如果您无法看到文件浏览器，按切换文件导航按钮



提示：如果您使用缩小的代码工作，按底部的大括号，使其可读。



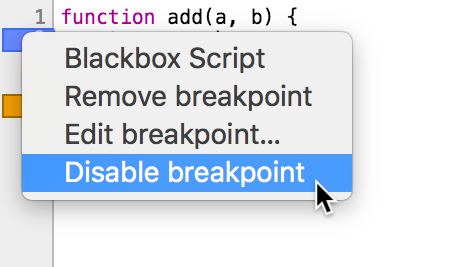
随着您的源代码的左侧可以看到行号。这个区域被称为行号排水沟。点击行号排水沟内该行的代码添加断点。

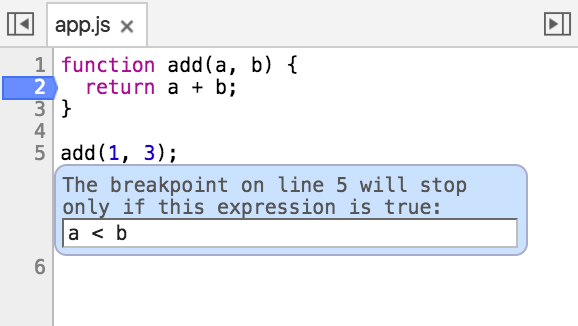


设置行号的断点条件或给正在断点的变量赋值

断点条件

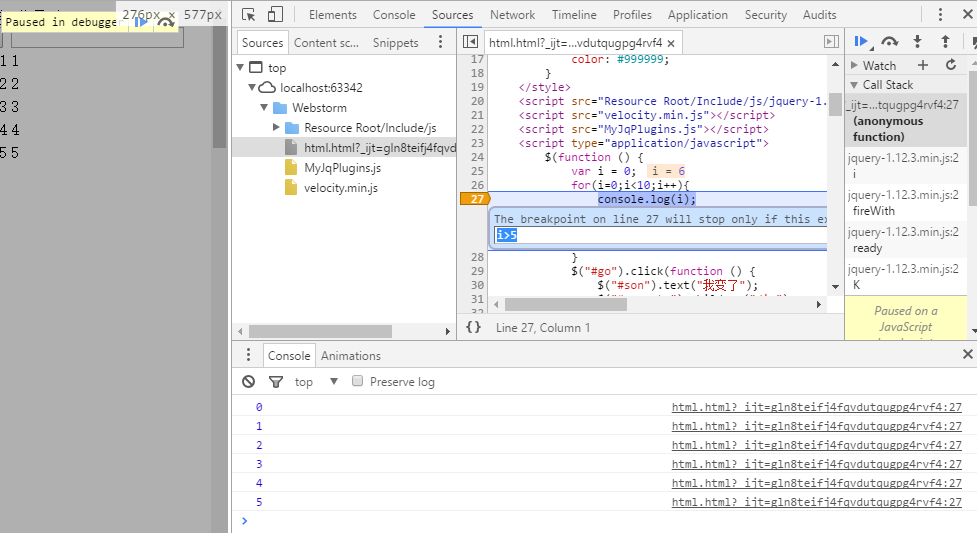
当您指定的条件是真实的条件断点时才会触发。如果您已经添加了一个断点，你想设置断点条件，右键单击，然后按下Edit Breakpoint 。



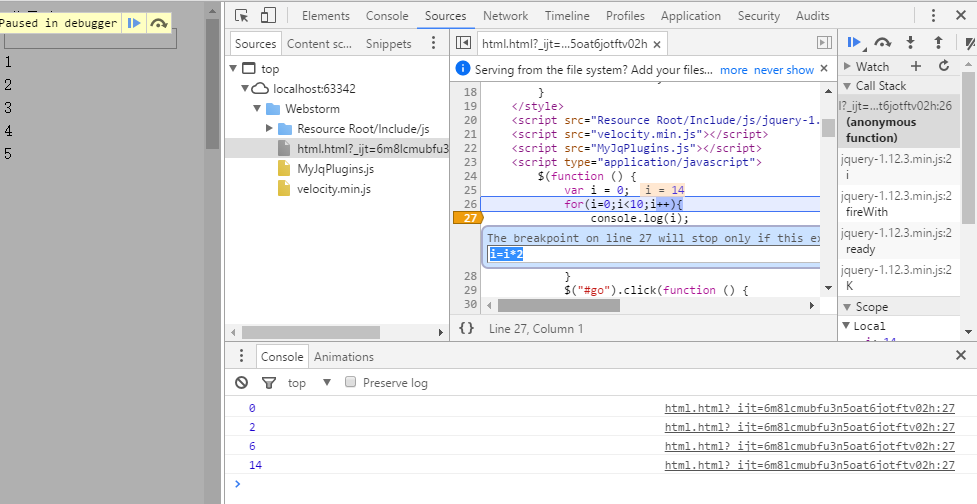


设置好了然后按Enter进行保存。

效果：

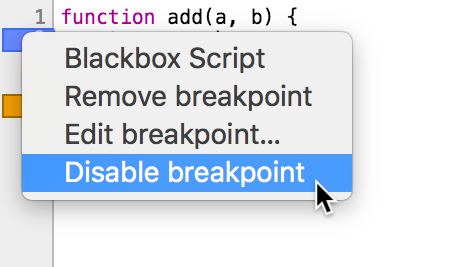


给正在断点的重新变量赋值 //设置语句i=i\*2,结果输出0、2、6、14

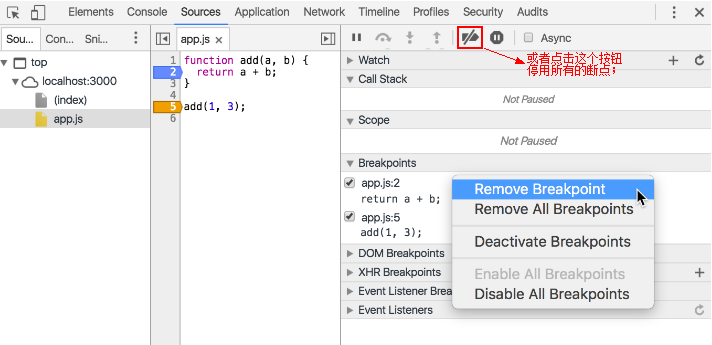


删除或禁用行号断点

第一种方法



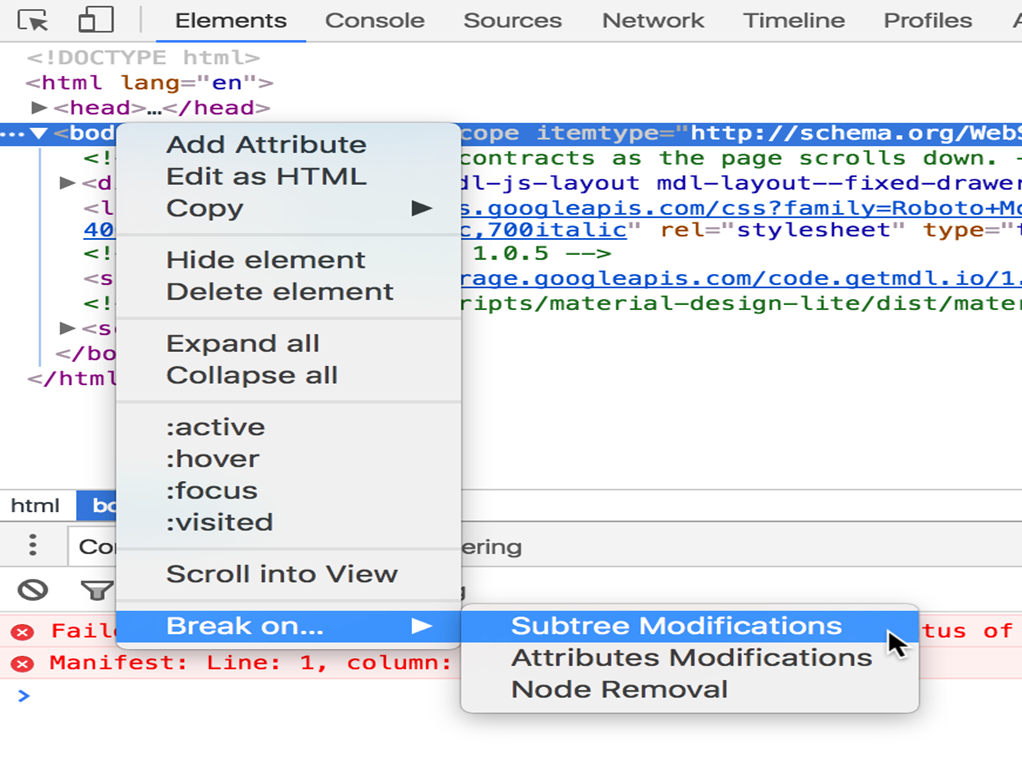
第二种方法



* + - * 1. 设置DOM改变时的断点（Set a breakpoint on DOM change）

意义：当DOM节点被添加，删除或以意想不到的方式修改时，而你不知道是什么脚本导致的DOM

节点的变化；这时设置一个DOM断点是非常有用的。



**提示**：这不是唯一的选项。可以同时设置多个选项。

子树的修改（Subtree Modifications）

当前所选节点的子节点被删除，添加或内容被改变时会导致断点触发。

子节点的属性变化或当前选中节点改变任何条件时都不会触发断点。

属性修改（Attributes Modifications）

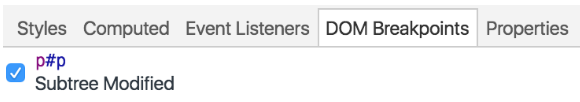
当前选中节点的属性发生变化时会触发断点。

节点删除（·Node Removal）

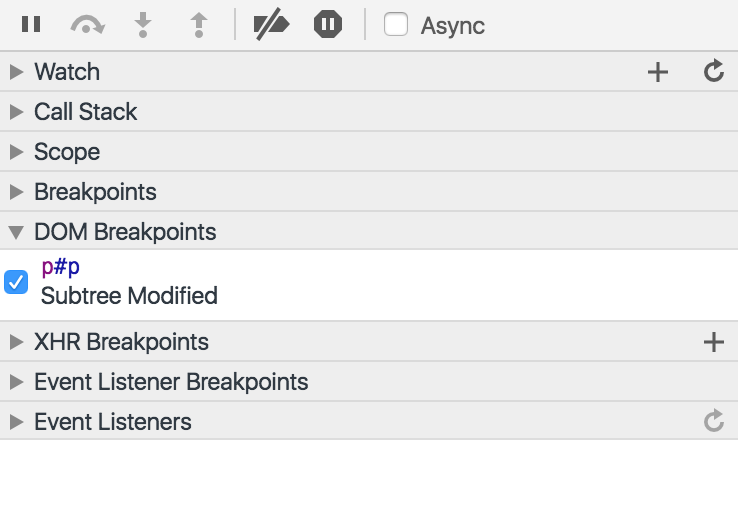
当前选中节点被删除时会发生触发

管理DOM断点

Element面板

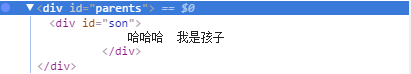


Sources面板



例子

先在id为parents的div上同时设置三个DOM断点：

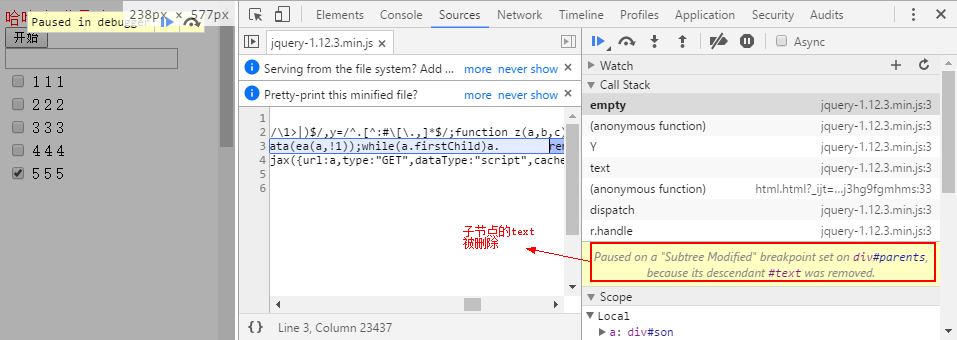


然后设置触发事件： 注意：如果更改DOM操作直接在$(function(){})里面就不起作用

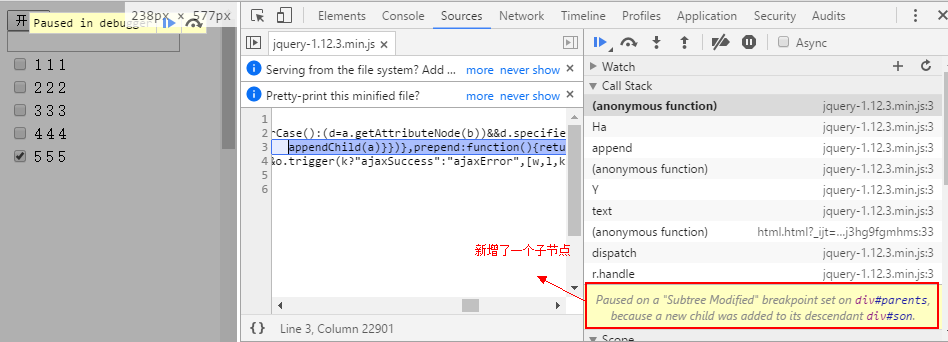


结果：

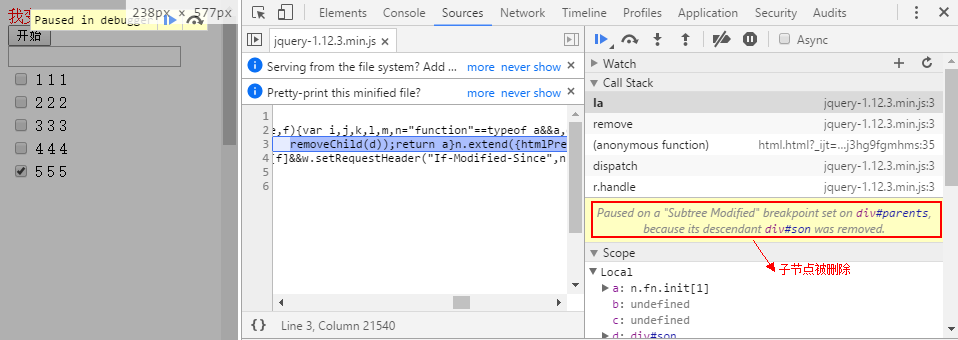




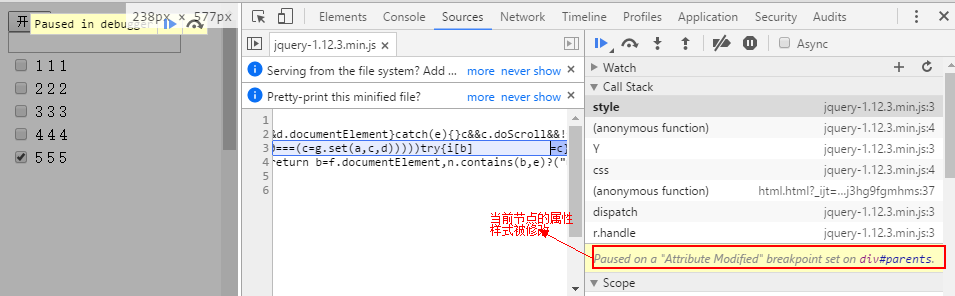




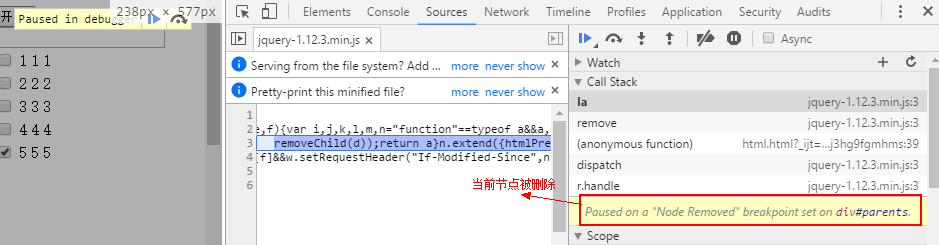




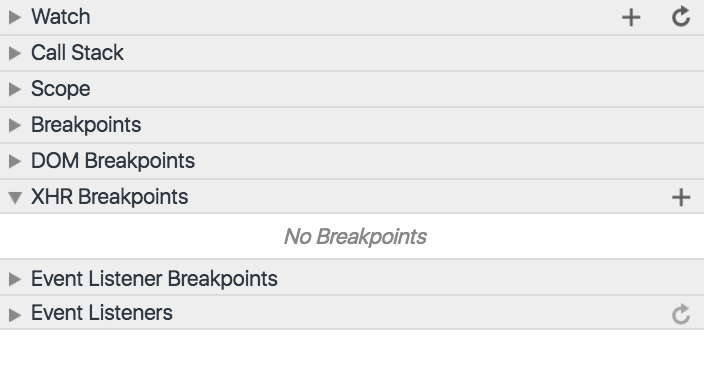






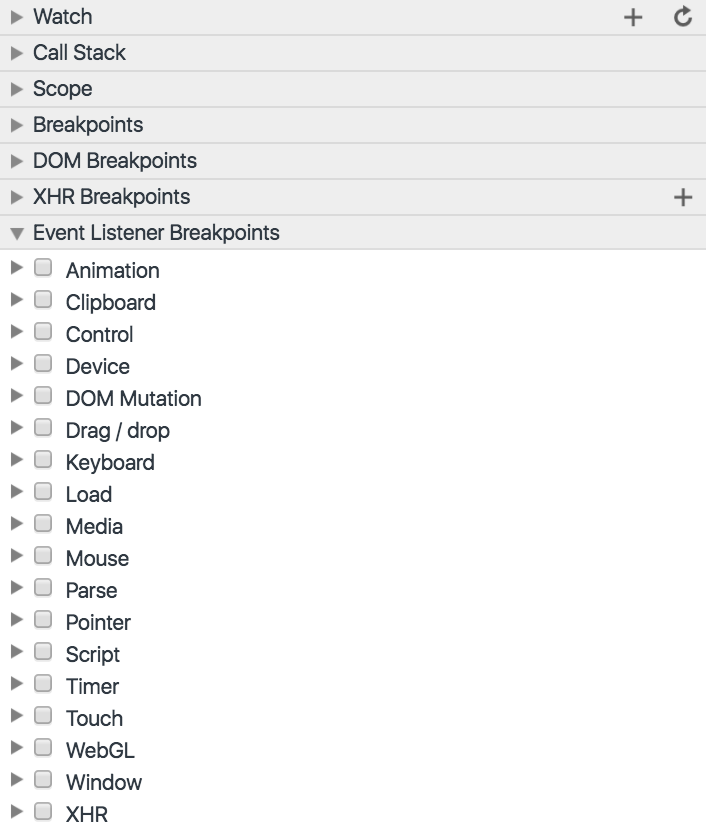


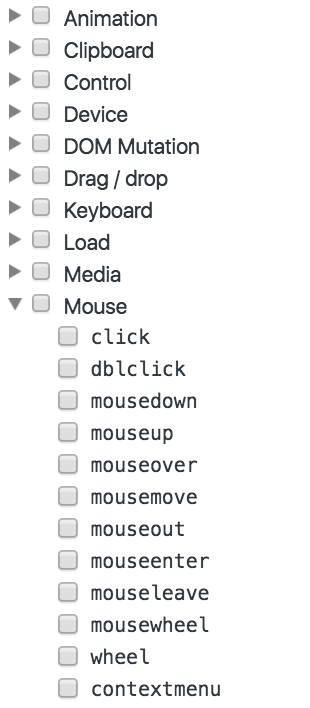
* + - * 1. XHR断点（Break on XHR）

XHR Breakpoints,可以方便的断到XMLHttpRequest Ajax请求；推荐原生的javascript写法

* + - * 1. 事件触发断点（Break when an event is fired）

监听页面所有节点的事件(Sources面板)

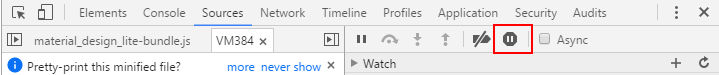




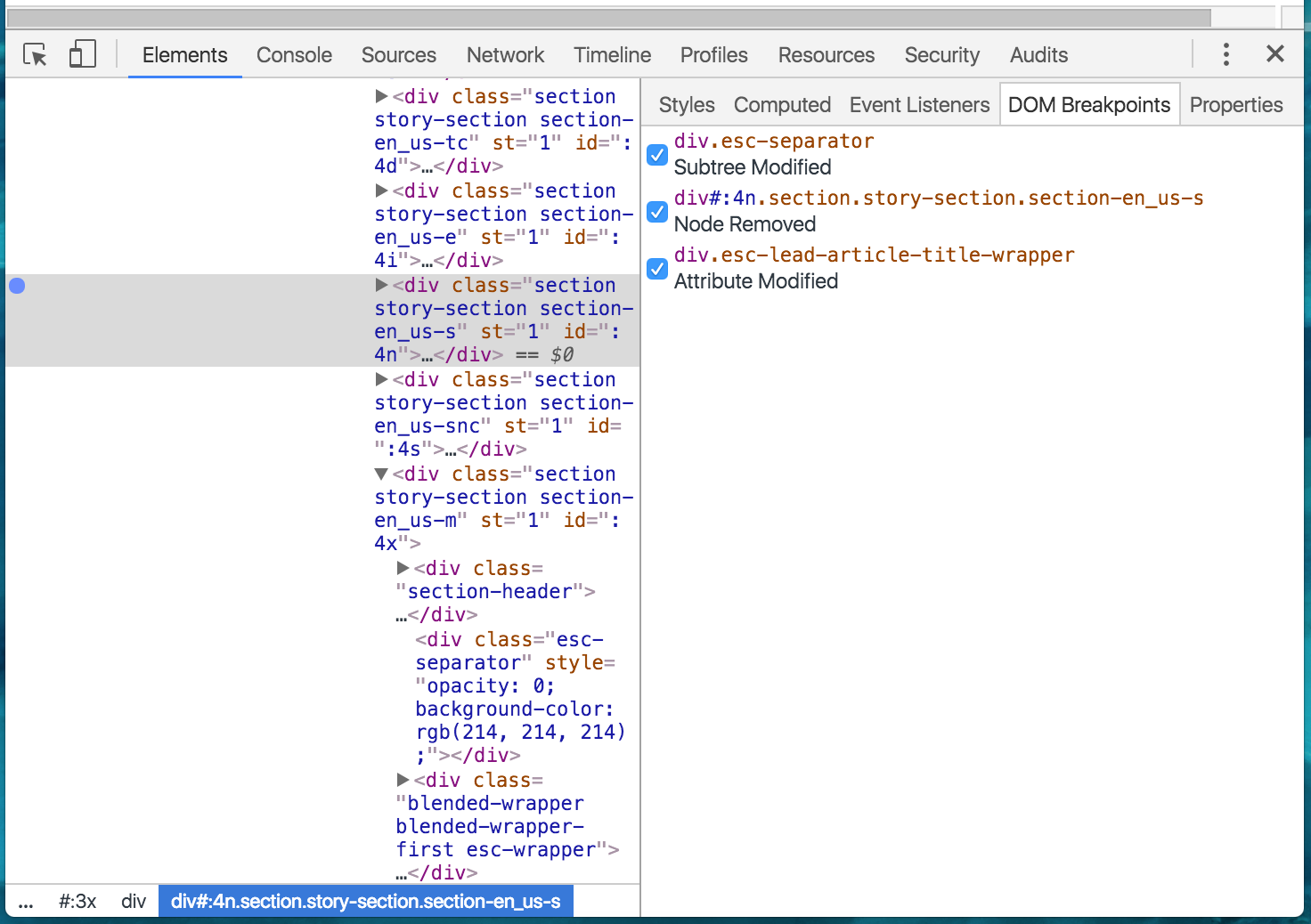
* + - * 1. 抛出未捕获的异常（Break on uncaught exceptions）

如果你的代码抛出异常，而你不知道他们从哪里来，按下Sources面板的异常暂停按钮

DevTools自动断开在其中抛出异常的行。



* + - * 与DOM断点互动（Interact with DOM breakpoints） //触发断点时抛出的异常上文已经有截图

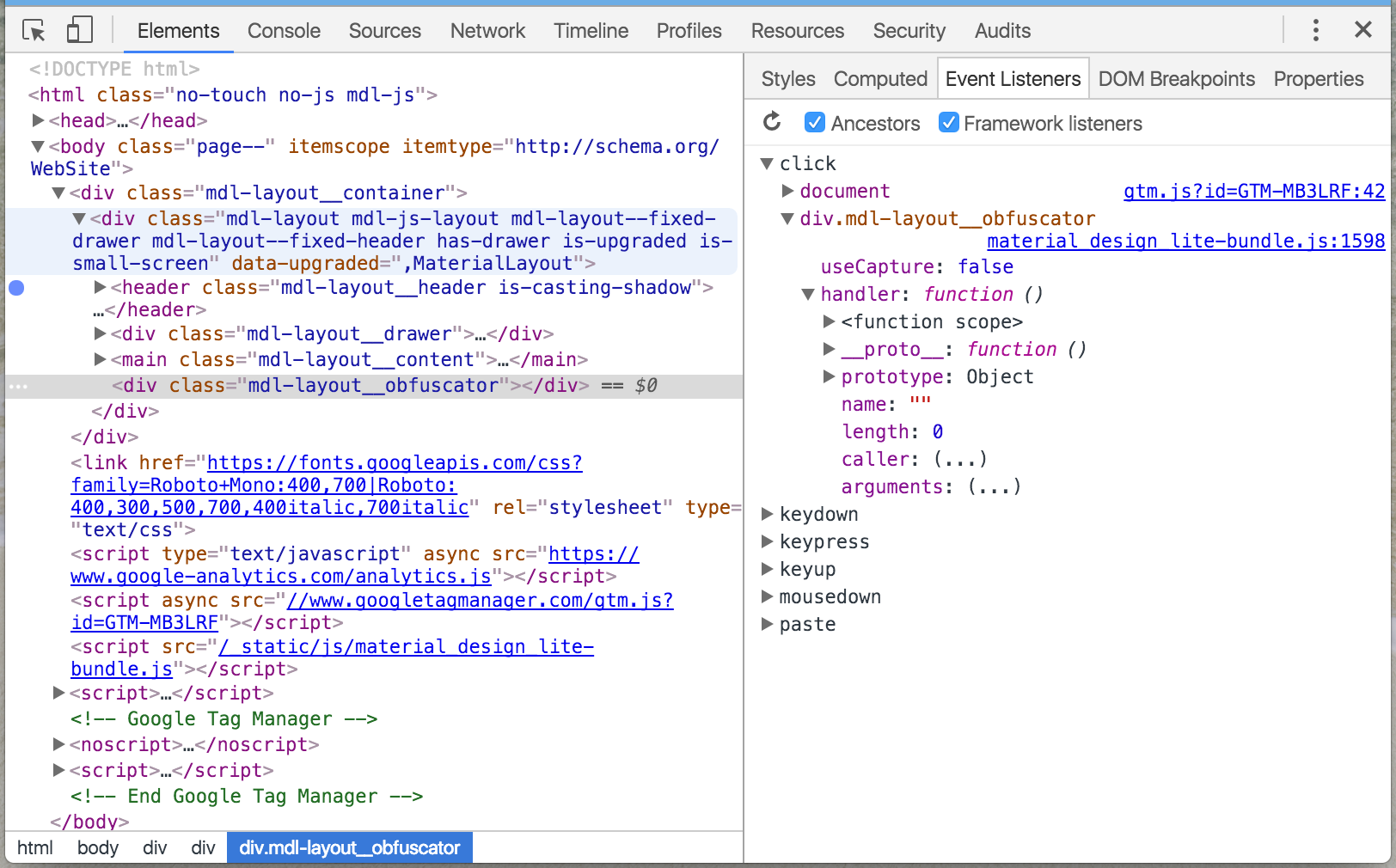


将鼠标悬停在Element面板的节点上，以显示在页面上的元素的相应位置

点击一个元素的元素面板将其选中。

切换复选框来启用或禁用断点。

* + - * 观察元素的事件监听队列（View element event listeners）
        1. Element面板的Event Listeners区域



* + - * 1. 事件监听器属性

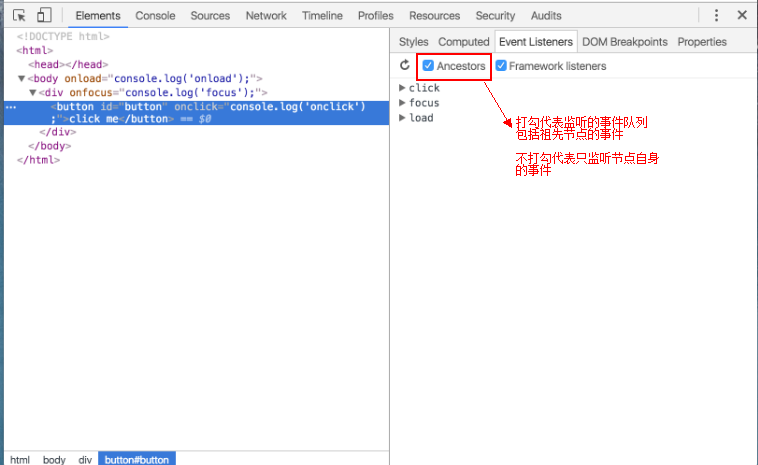
Handler: 包含一个回调函数。在handler上右键单击，然后选择show function definition查看函数的定义在哪里（如果源代码可用）。

useCapture: 一个布尔值，说明addEventListener事件是否设置useCapture参数。默认值为false,代表事件在捕获阶段不激活；如果useCapture为true则代表事件在冒泡阶段不激活

参考文章：<http://javascript.ruanyifeng.com/dom/event.html>

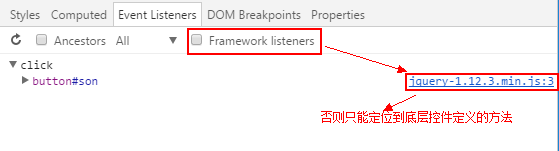
注意：若要想去除chrome浏览器自身携带的事件监听，则要开启chrome的隐身模式。

* + - * 1. 查看祖先事件监听器 （Ancestors）



* + - * 1. 查看框架监听器





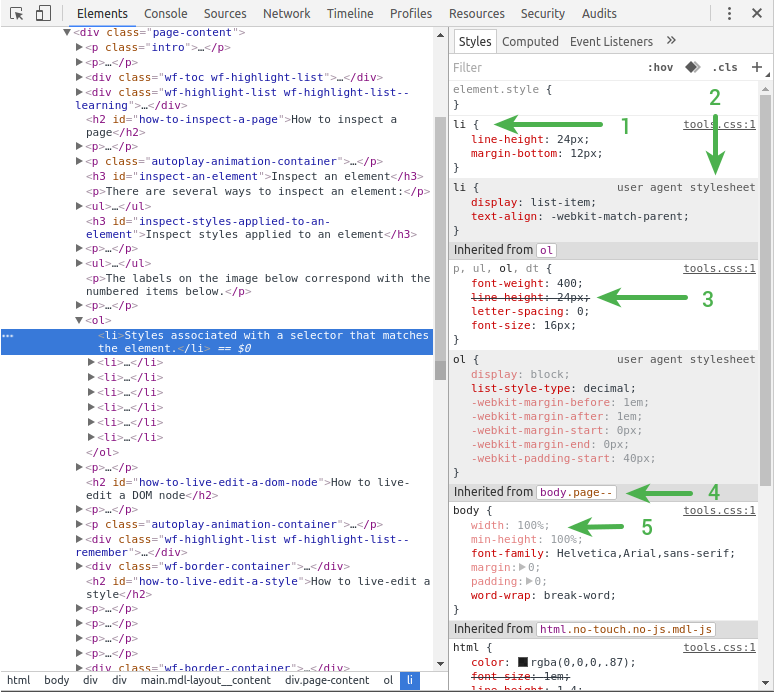
* + 实时编辑元素样式（Live-edit a style）
    - 检查应用于元素的样式 (Inspect styles applied to an element)
      * 在样式面板显示应用到选定的元素，从最高优先级到最低的CSS规则：

在顶部element.style。这些样式要么使用样式属性直接施加到元件（例如，<p style="color:green">），或在DevTools施加。

浏览器默认的样式，任何CSS规则,。例如，在所选择的元件下面的截图接收line-height:24px从所定义的规则tools.css。

再就是继承的样式，其中包括符合所选元素的祖先，任何可继承样式规则。例如，在选定的元素下面的规则继承于user agent stylesheet的display:list-item属性。

* + - * Demo



1、与选择元素相匹配的样式。

2、清楚地标明浏览器默认样式，并经常覆盖您网页上的CSS。

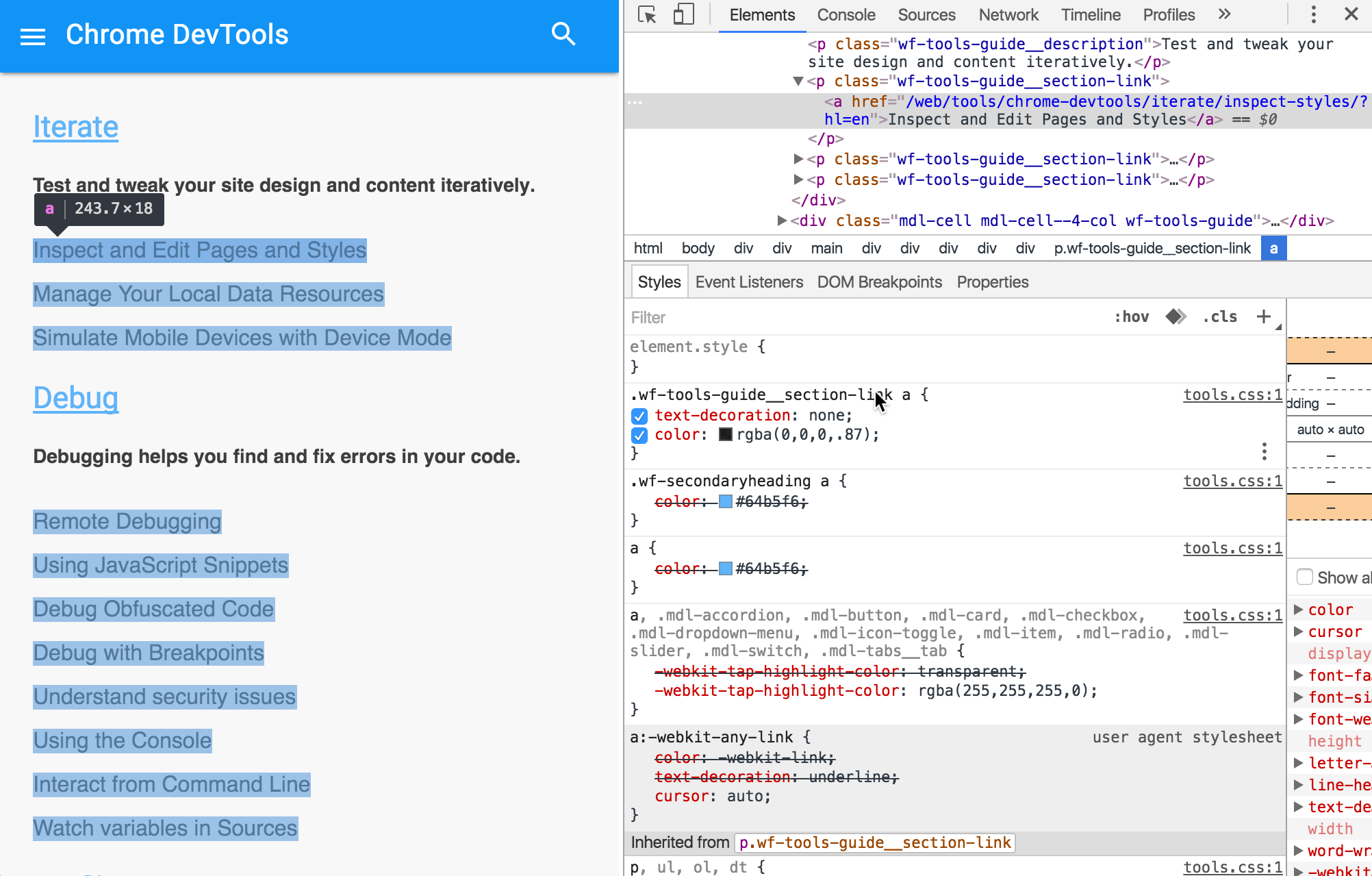
3、如果级联规则被重写则带有删除文本

4、继承的样式下的“从继承显示为一组<NODE>头”。点击标题的DOM节点导航到其在DOM树视图位置。（在CSS 2.1属性表显示哪些属性可以继承。）

5、样式中灰色条目是指没有定义而在运行时生效的样式。

* + - 检查受选择器影响的元素(Inspect elements affected by a selector)

把鼠标放在Element面板的节点上，浏览器页面所对应的标签会有所影响。



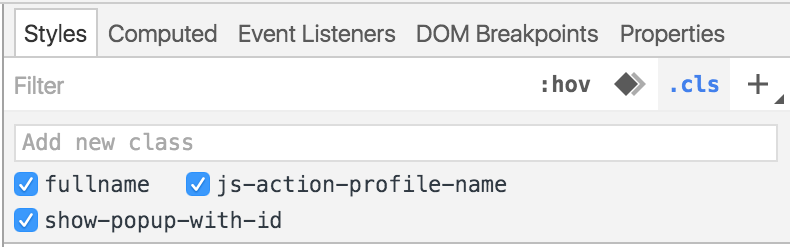
注意：此功能仅突出了视区的元素; 浏览器视区之外的其他元素也受到选择。右键—scroll into view

* + - 添加，启用和禁用CSS类 (Add, enable, and disable CSS classes)

点击.CLS按钮来查看相关联的所有当前选定元素的CSS类。从那里，你可以：

启用或禁用当前​​关联的元素的类。

新类加入的元素。



* + - 编辑现有属性名称或值 (Edit an existing property name or value)

点击一个CSS属性名称或值进行编辑。名称或值就会突出显示，按Tab向前移动到下一个属性，名称。按住Shift，然后按Tab向后移动。

* + - 添加一个新的属性(Add a new property declaration)

点击可编辑CSS规则中的空白区域来创建一个新的规则。点击Enter，或粘贴到CSS的**样式**面板中。新添加的属性和它们的值被解析并应用到正确的字段。

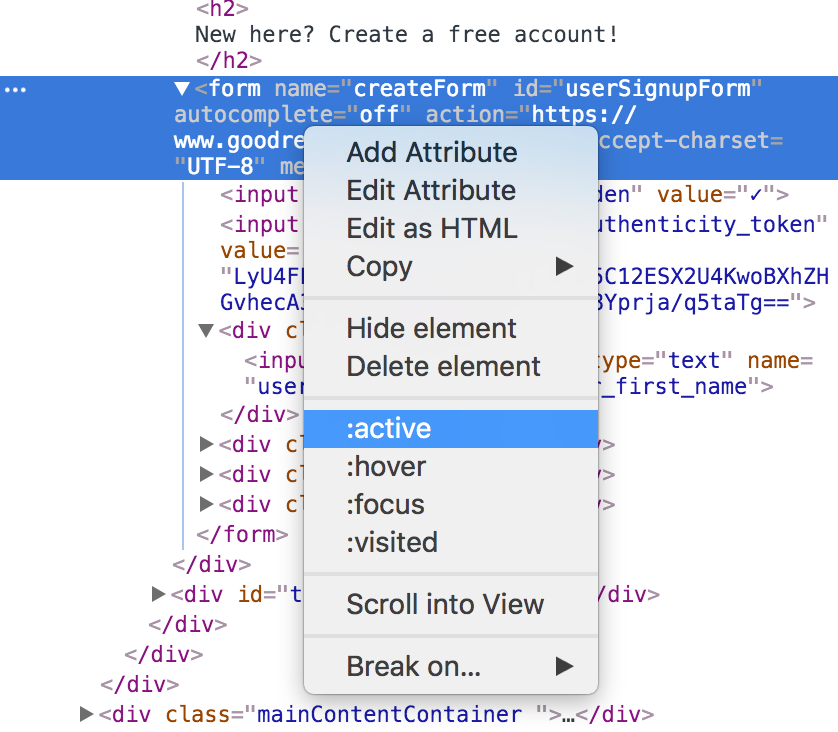
* + - 添加样式规则 (Add a style rule)
      * 单击新建样式规则按钮，添加一个新的CSS规则。
      * 可以选择新定义的样式规则应用到哪些节点



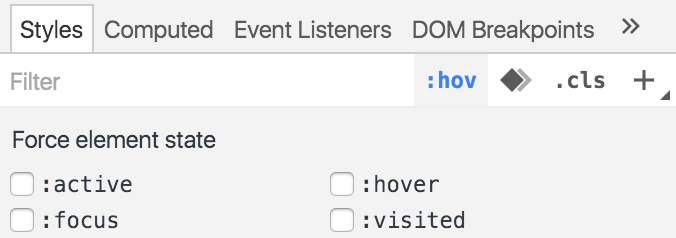
* + - 添加或删除动态样式（伪类）(Add or remove dynamic styles (pseudo-classes))

有两种方法模拟动态样式（伪类）：

1、在选中的元素上单击鼠标右键**菜单**面板，然后从菜单中选择目标伪类启用或禁用它。



2、在**Element**面板中选择在一个元素，单击**：HOV**按钮，在**style**面板，并使用复选框来启用或禁用当前所选元素的选择



* + - 添加背景颜色或颜色到样式规则 (Add background-color or color to a style rule)

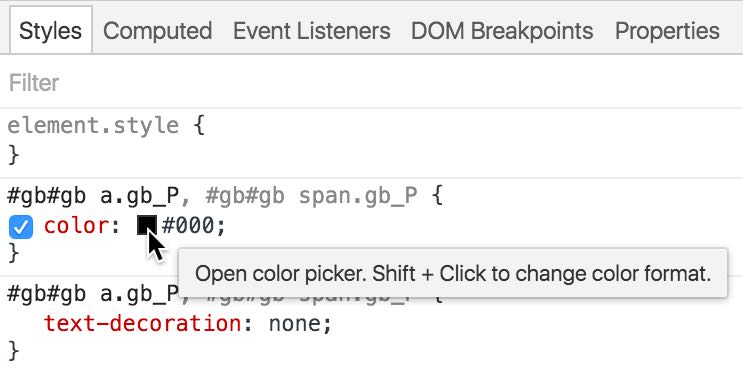
在样式规则的右下角，有一个有三个点的图标。您需要将鼠标悬停在样式规则才能看到它。



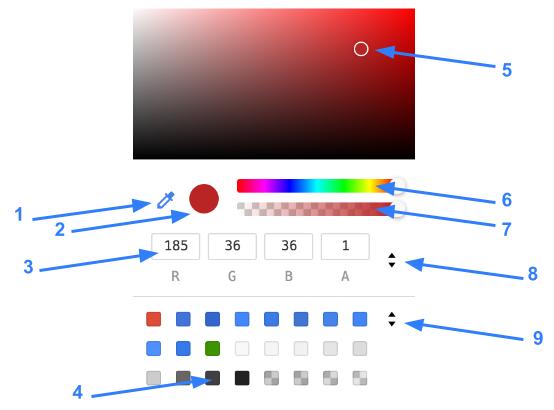
将鼠标悬停在该图标以显示按钮添加一个color属性（增添色彩宣言）或background-color属性（添加背景色声明）。点击这些按钮之一将属性添加到样式规则。

* + - 利用颜色选择器修改颜色 (Modify colors with the Color Picker)
      * 打开颜色选择器

要打开**拾色器**，发现在一个CSS声明**样式**定义的颜色（如面板color: blue）。属性值的左边有一个小的，彩色的正方形。正方形的颜色与页面显示的颜色相匹配。点击这个正方形打开**颜色选择器**。

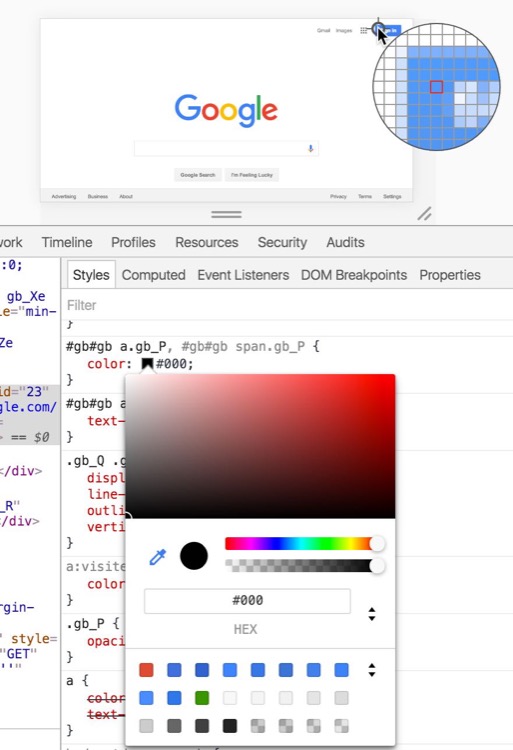


* + - * 颜色选择器的使用



* + - * 1. 吸管。见滴管以获取更多信息。

点击滴管，使其启用（按钮启用滴管），鼠标悬停在页面上的色彩，然后单击设置当前选定的色彩，颜色选择器面板将会出现鼠标刚刚点击的颜色。



* + - * 1. 当前颜色。可视化表示当前颜色值。
        2. 当前值。在十六进制，RGBA，HSL或代表当前颜色。
        3. 调色板。见调色板以了解更多信息。

在颜色选择器提供了以下调色板：

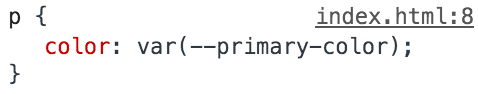
页面颜色。从页面的CSS自动生成的一组颜色。

材料设计。与设计规范的颜色保持一致。

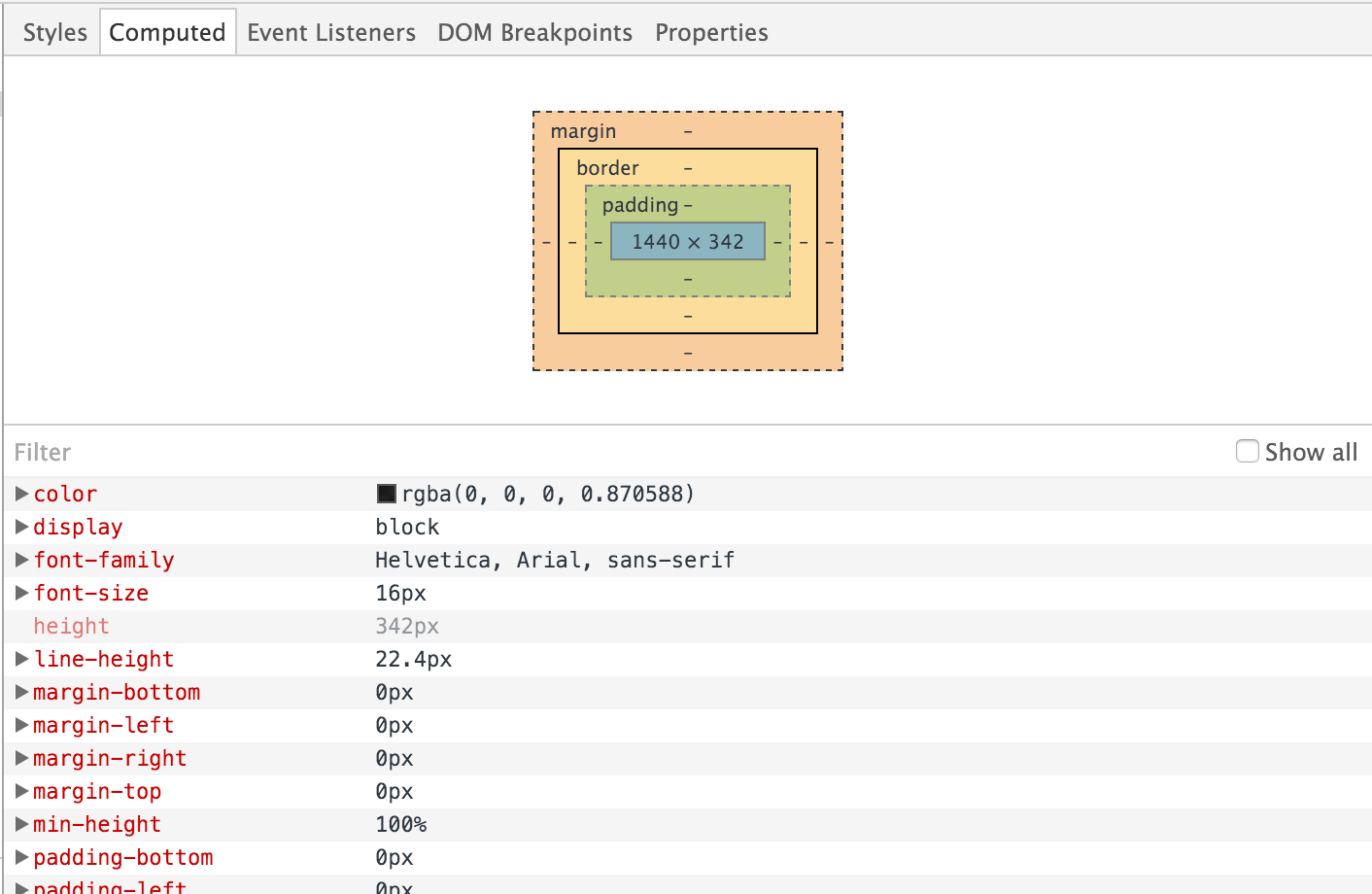
自定义。您选择的任何颜色。DevTools保存您的自定义选项板，甚至是跨页，直到你删除它。

* + - * 1. 色调和颜色选择。
        2. 色调选择。
        3. 透明度选择。
        4. 颜色值选择。点击RGBA，HSL和十六进制之间切换。
        5. 调色板选择。单击选择不同的模板。
    - 查看和编辑CSS自定义属性（CSS变量）(View and edit CSS custom properties (CSS variables))

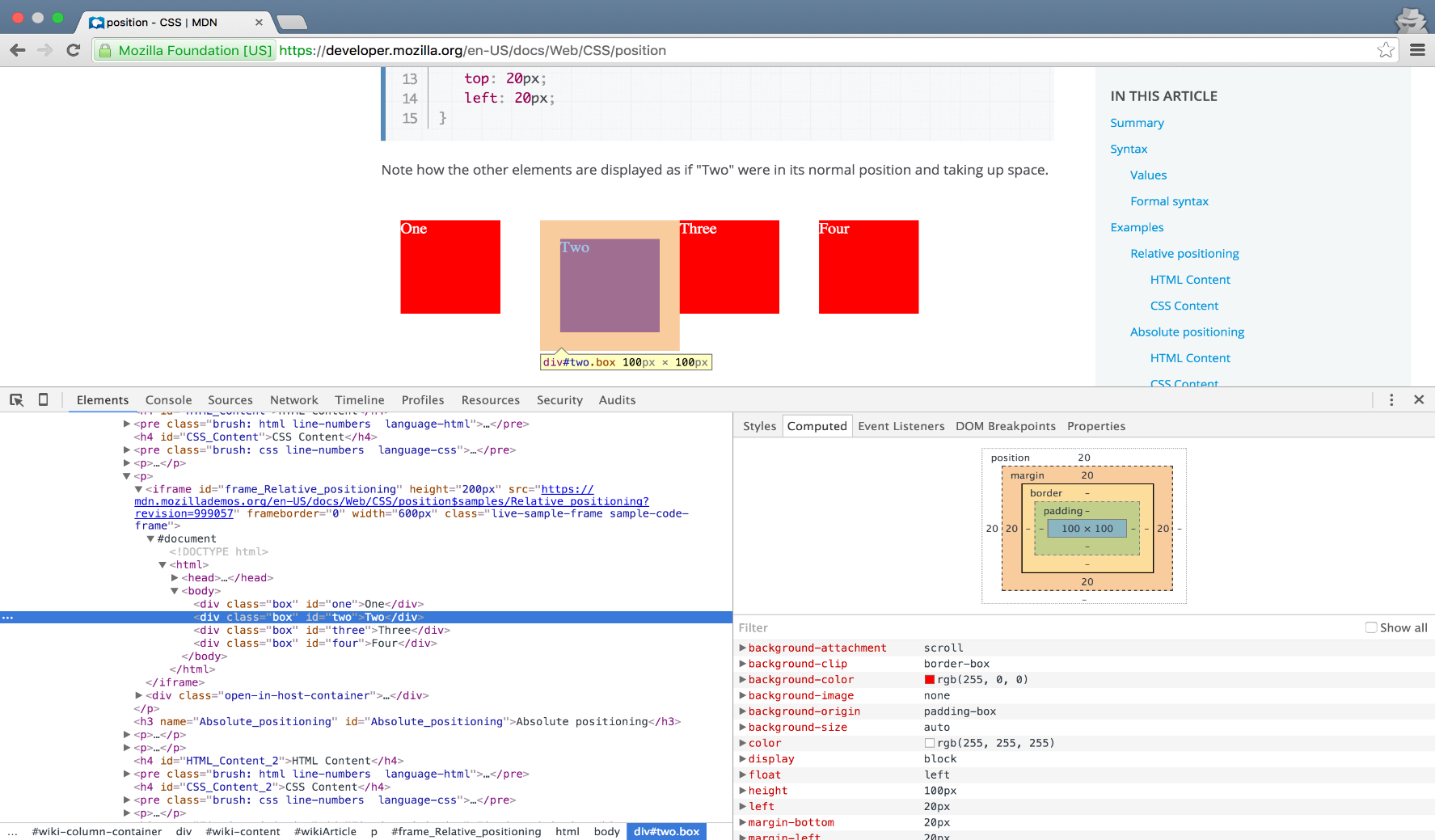
如果你看到这样的声明价值var(--main-color)，如下面的截图，这意味着该声明是使用自定义属性。这些值可以像其他任何声明值一样被编辑。目前还没有办法跳转到自定义属性定义。



* + - 编辑三种预处理器框架Sass, Less 和 Stylus (Edit Sass, Less or Stylus)（未完待续）https://developers.google.com/web/tools/setup/setup-preprocessors
  + 检查和编辑盒子模型参数 （Examine and edit box model parameters）
    - 在Element中的Computed面板中检查并编辑当前元素的盒子模型参数。在盒子模型中的所有值都可以编辑，只需点击他们。如图：

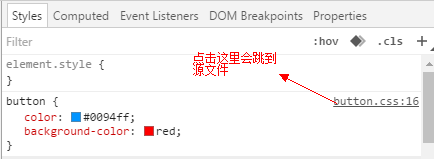


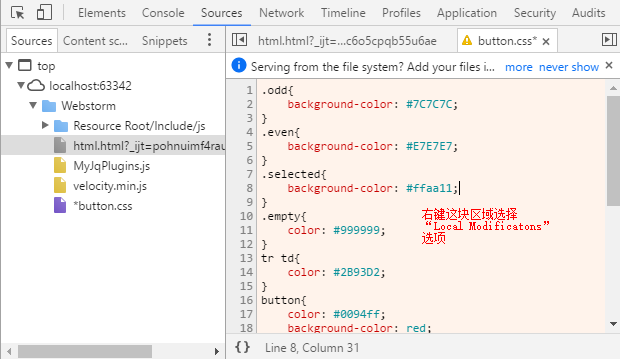
盒子模型包含**顶部**，**底部**，**左**，**右**当前元素的padding**填充**，**border**和**margin**属性。



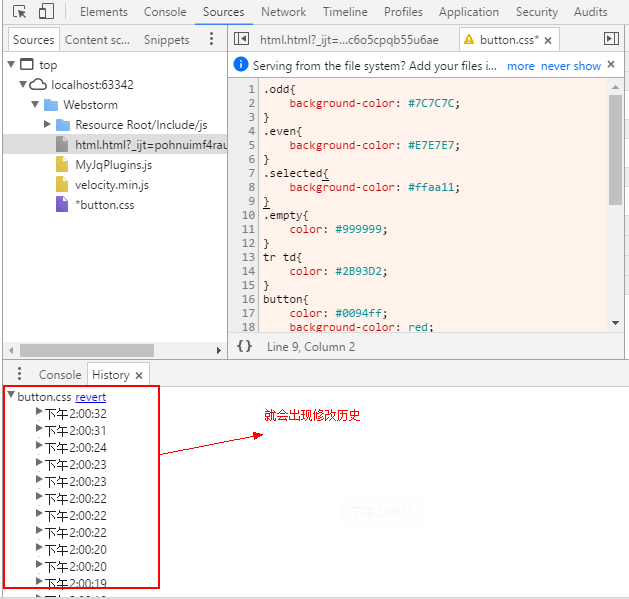
所有的值可以通过双击它们，就像在样式面板中的属性值进行修改;(并不会保存到源文件)

* + 查看本地更改（View local changes）
    - 要查看到您的网页制作实时编辑的历史记录：
      * 在Element面板的style窗口中，单击您修改的文件。DevTools会带您到Sources面板中。
      * 在Sources面板的文件上单击鼠标右键。
      * 选择本地修改(Local modifications)。





Next:



* + - 查看所做的更改：
      * 展开顶层的文件名 ​​来查看在哪个时间改变发生了哪些改变。
      * 展开的第二级项来查看与上一个版本的差异（前和后）。用粉红色背景的行表示更改前的状态，同时背景为绿色的行表示更改后的状态。



* + 撤销更改（Undo changes） 源文件映射到本地文件的章节：https://developers.google.com/web/tools/setup/setup-workflow
    - 如果你还没有把文件映射到本地，你重新加载页面的时候，所有活的编辑都将丢失。
    - 假设你已经设置了相应的映射，撤消更改：
      * 使用Ctrl+ Z（Windows）或Cmd+ Z（Mac）来快速撤消通过元素面板的DOM或样式小的变化。
      * 要撤消对文件所做的所有本地修改，打开源面板，然后选择文件名旁边的revert 键​​。
  + 主题（Topics）
    - 检查动画（Inspect animations）
      * https://developers.google.com/web/tools/chrome-devtools/iterate/inspect-styles/animations?hl=en
    - 键盘与快捷键参考（Keyboard & UI Shortcuts Reference）
      * https://developers.google.com/web/tools/chrome-devtools/iterate/inspect-styles/shortcuts?hl=en
* 编辑样式（[Edit Styles](https://developers.google.com/web/tools/chrome-devtools/iterate/inspect-styles/edit-styles)）
  + 内容上文已经给出
* 编辑DOM ([Edit the DOM](https://developers.google.com/web/tools/chrome-devtools/iterate/inspect-styles/edit-dom))
  + 内容上文已经给出

CONSOLE

SOURCES

NETWORK

TIMELINE

PROFILES

RESOURCES

SECURITY

DEVICE MODE

REMOTE DEBUGGING

SETTINGS