



**FREE eBook**

# LEARNING google-chrome- devtools

Free unaffiliated eBook created from  
**Stack Overflow contributors.**

#google-  
chrome-  
devtools

# Table of Contents

About.....	1
<b>Chapter 1: Getting started with google-chrome-devtools .....</b>	<b>2</b>
Remarks.....	2
Examples.....	2
Search by filename.....	2
Go to member (javaScript function/CSS rule).....	2
Accessing Chrome Dev Tools.....	2
<b>Chapter 2: Editable Content.....</b>	<b>4</b>
Introduction.....	4
Examples.....	4
Making content editable.....	4
<b>Chapter 3: Print stack traces.....</b>	<b>5</b>
Examples.....	5
Error.stack.....	5
console.trace().....	5
console.assert().....	6
<b>Credits.....</b>	<b>8</b>

---

# About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [google-chrome-devtools](#)

It is an unofficial and free google-chrome-devtools ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official google-chrome-devtools.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to [info@zzzprojects.com](mailto:info@zzzprojects.com)

---

# Chapter 1: Getting started with google-chrome-devtools

## Remarks

This section provides an overview of what google-chrome-devtools is, and why a developer might want to use it.

It should also mention any large subjects within google-chrome-devtools, and link out to the related topics. Since the Documentation for google-chrome-devtools is new, you may need to create initial versions of those related topics.

## Examples

### Search by filename

While in the sources tab use `CTRL+O` (`⌘+O` for Mac) to search by filename.

### Go to member (javaScript function/CSS rule)

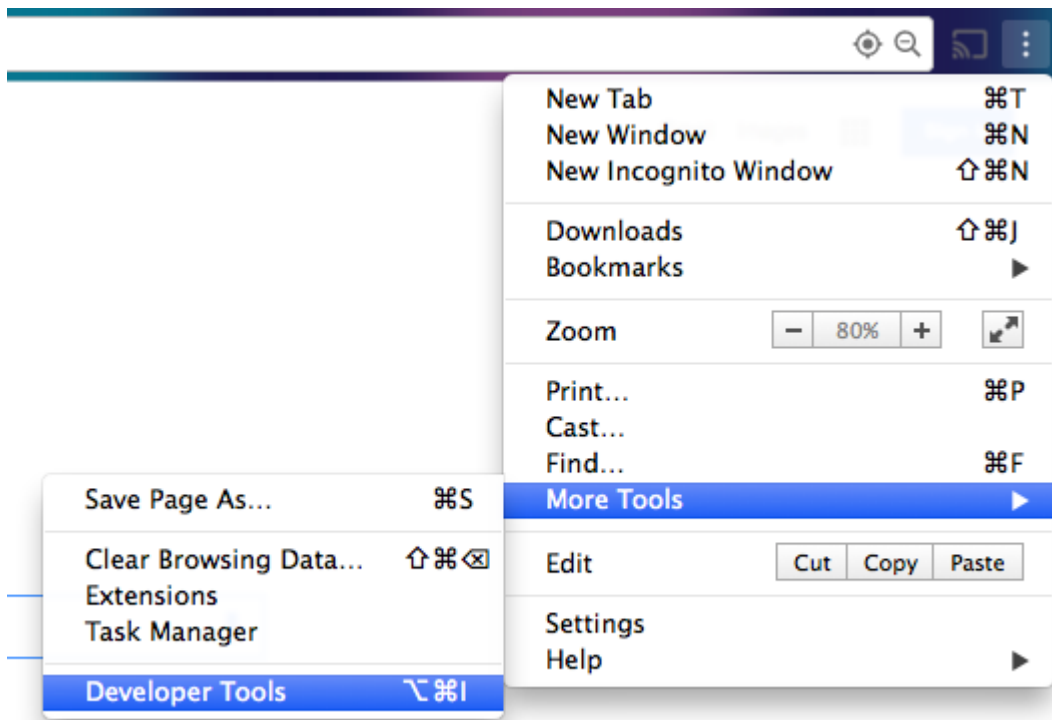
Use `CTRL+SHIFT+O` (`⌘+SHIFT+O` for mac) to navigate to a javaScript function/CSS rule when viewing a file.

### Accessing Chrome Dev Tools

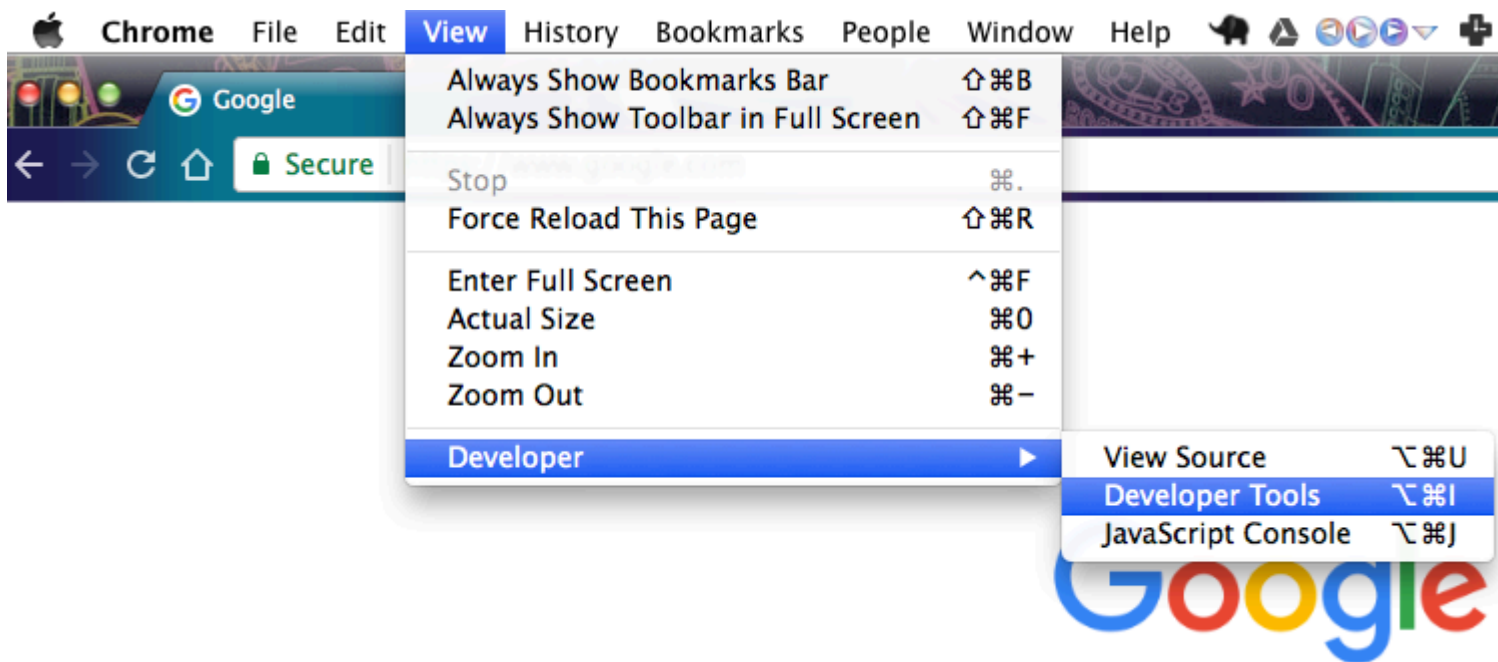
While chrome browser is open to any tab (except welcome tabs) you have three options to open Chrome Dev Tools:

**Keyboard:** Type `command ⌘+option+i`

**Browser Menu:** Click 'Menu' > 'More Tools' > 'Developer Tools'



**Program Menu** (at top of your screen): Click 'View' > 'Developer' > 'Developer Tools'



Read Getting started with google-chrome-devtools online: <https://riptutorial.com/google-chrome-devtools/topic/1423/getting-started-with-google-chrome-devtools>

---

# Chapter 2: Editable Content

## Introduction

You can use `document.body.contentEditable = true` or `document.designMode = 'on'` to edit content in browser.

## Examples

### Making content editable

Open console:

- `Ctrl+Shift+J` (Windows/Linux)
- `Cmd+Opt+J` (Mac)

Insert `document.body.contentEditable = true` or `document.designMode = 'on'` and press `ENTER`

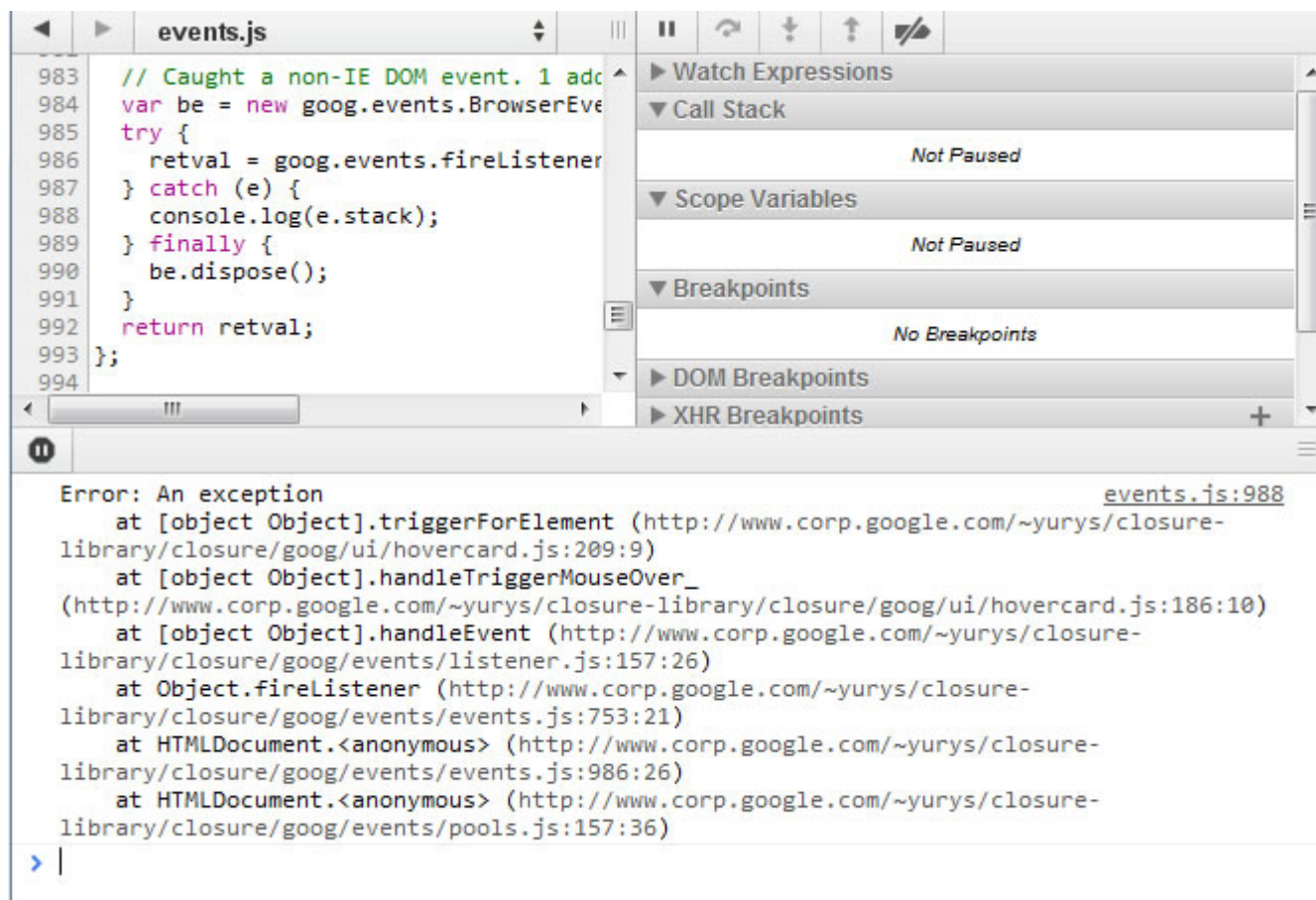
Read Editable Content online: <https://riptutorial.com/google-chrome-devtools/topic/9293/editable-content>

# Chapter 3: Print stack traces

## Examples

### Error.stack

Each Error object has a string property named stack that contains the stack trace:



### console.trace()

Instrument your code with `console.trace()` calls that print current JavaScript call stacks:

The screenshot shows a web browser's developer console. At the top, there are tabs for 'positioning.js', 'events.js', and 'hovercard.js\*'. The 'hovercard.js' tab is active, showing a JavaScript function definition for `goog.ui HoverCard.prototype.triggerForElement`. The function starts with `console.trace();` followed by an `if` statement checking if `anchorElement` is the same as `this.currentAnchor_`. If true, it calls `this.clearHideTimer();` and `return;`. The code is highlighted in a light orange background.

Below the code editor, the console panel is open, showing a list of messages. The first three messages are 'Recompilation and update succeeded.' in blue. The fourth message, marked with a blue circle containing the number 4, is a console trace for `console.trace()`. The trace shows the following call stack:

- `goog.ui HoverCard.triggerForElement`
- `goog.ui HoverCard.handleTriggerMouseOver_`
- `goog.events.fireListener`
- `goog.events.handleBrowserEvent_`
- `(anonymous function)`

A blue arrow points to the right below the trace.

## console.assert()

Place assertions in your JavaScript code by calling `console.assert()` with the error condition as the first parameter. When this expression evaluates to false, you will see a corresponding console record:



positioning.js    events.js    **hovercard.js\*** x

```
217
218
219 console.assert(this.currentAnchor_ == undefined, '
220
221 if (anchorElement == this.currentAnchor_) {
222     // Element is already showing, just make sure it
223     this.clearHideTimer();
224
225 }
```

Line 225, Column 4

Console    Search    Emulation    Rendering

<top frame>    <page context>

Recompilation and update succeeded.

**✖ Assertion failed:**

goog.ui.HoverCard.triggerForElement  
goog.ui.HoverCard.handleTriggerMouseOver\_  
goog.events.fireListener  
goog.events.handleBrowserEvent\_  
(anonymous function)

Recompilation and update succeeded.  
Recompilation and update succeeded.

>

Read Print stack traces online: <https://riptutorial.com/google-chrome-devtools/topic/3148/print-stack-traces>

---

# Credits

S. No	Chapters	Contributors
1	Getting started with google-chrome-devtools	<a href="#">CD..</a> , <a href="#">cdrini</a> , <a href="#">Community</a> , <a href="#">Kara Carrell</a> , <a href="#">UserNotFoundException</a>
2	Editable Content	<a href="#">zucker</a>
3	Print stack traces	<a href="#">UserNotFoundException</a>