

# Fundamentos do Desenvolvimento de Software

Programação Web com JavaScript I

# Agenda

## **Etapas 9:** Preparando-se para o AT.

- Leitura do AT.
- Depuração de Código.



# Leitura do AT

# Depuração de Código

**Depuração** (em inglês: debugging, debug) é o processo de encontrar e reduzir defeitos num aplicativo de software ou mesmo em hardware.

**Erros de Software** incluem aqueles que previnem o programa de ser executado e aqueles que produzem um resultado inesperado.

**Depurador** (em inglês: debugger) é um **programa de computador usado para testar outros programas** e fazer sua depuração, que consiste em encontrar os defeitos do programa.

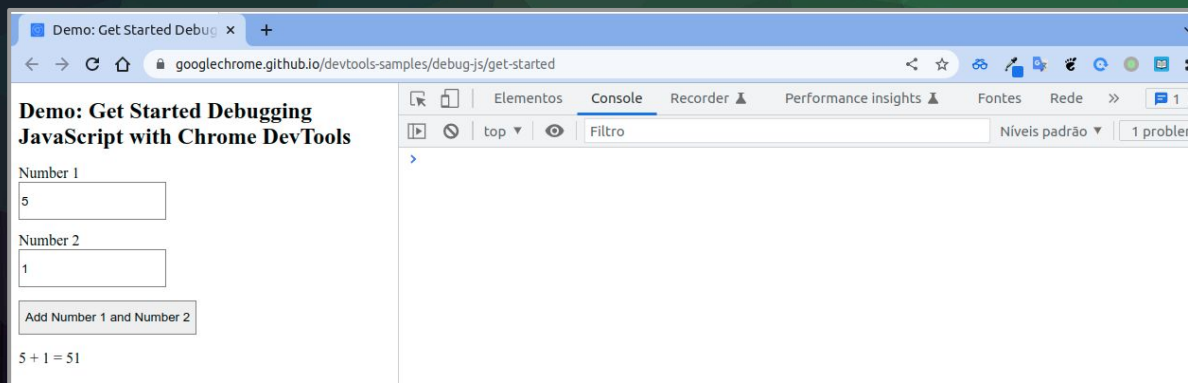
Os **depuradores** oferecem funcionalidades mais sofisticadas como:

- Execução passo-a-passo de um programa;
- Suspensão do programa para examinar seu estado atual, em pontos pré definidos, chamados pontos de parada;
- Acompanhamento do valor de variáveis que podem ser usadas inclusive para gerar uma suspensão, ou ativar um ponto de parada.



**DevTools** é um conjunto de ferramentas de desenvolvedor da Web criadas diretamente no navegador Google Chrome.

**DevTools** pode ajudar a editar páginas em tempo real e diagnosticar problemas rapidamente, o que, em última análise, ajuda a criar sites melhores e mais rápidos.



<https://developer.chrome.com/docs/devtools/>

Demo: Get Started Debugging JavaScript with Chrome DevTools

Number 1  
5

Number 2  
1

Add Number 1 and Number 2

5 + 1 = 51

CTRL + SHIFT + I  
ou  
F12

1

2

The screenshot shows a web browser window with a tab titled 'Demo: Get Started Debug'. The address bar shows the URL 'googlechrome.github.io/devtools-samples/debug-js/get-started'. The page content includes a form with two input fields labeled 'Number 1' and 'Number 2', containing the values '5' and '1' respectively. Below the inputs is a button labeled 'Add Number 1 and Number 2'. The result of the addition is displayed as '5 + 1 = 51'. The Chrome DevTools interface is open, with the 'Console' tab selected. The 'Fontes' (Sources) tab is also visible. A red circle with the number '1' is next to the text 'CTRL + SHIFT + I ou F12'. A red circle with the number '2' is next to the 'Fontes' tab. A large grey arrow points from the shortcut text to the 'Fontes' tab.

<https://googlechrome.github.io/devtools-samples/debug-js/get-started>



$$5 + 1 = 51$$

The screenshot shows the Chrome DevTools interface. The 'Fontes' (Sources) panel is active, displaying the 'get-started.js' file. The file content is as follows:

```
1 <!--  
2 Copyright 2016 Google Inc.  
3  
4 Licensed under the Apache License, Version 2.0 (the "License");  
5 you may not use this file except in compliance with the License.  
6 You may obtain a copy of the License at  
7  
8 http://www.apache.org/licenses/LICENSE-2.0  
9  
10 Unless required by applicable law or agreed to in writing, software  
11 distributed under the License is distributed on an "AS IS" BASIS,  
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or impl  
13  
14 limitations under the License.  
15 -->  
16  
17 <!-- referenced in: https://developers.google.com/web/tools/chrome-dev  
18  
19 <!doctype html>  
20 <html>
```

The 'Pilha de chamadas' (Call Stack) panel is visible at the bottom, showing the current execution context. The status bar at the bottom indicates 'Não pausado' (Not paused).

```
function updateLabel() {  
  var addend1 = getNumber1();  
  console.log('addend1:', addend1);  
  var addend2 = getNumber2();  
  console.log('addend2:', addend2);  
  var sum = addend1 + addend2;  
  console.log('sum:', sum);  
  label.textContent = addend1 + ' + ' + addend2 + ' = ' + sum;  
}
```

Um método comum para depurar um problema como esse é inserir várias instruções `console.log()` no código para inspecionar os valores durante a execução do script.





## Lista de Tarefas

Tarefa:

aaaa

aaaa

Agora é a sua vez!

671px x 639px

Open  
Sandbox

671px x 639px

Elementos Console Fontes Rede Desempenho >> 4 1

Página >> script.js x

```
1 function cadastrar() {
2   const novaTarefa = document.getElementById("novaTarefa")
3   if (novaTarefa.value) {
4     //falsy para validar se tem alguma string
5     const listaTarefas = document.getElementById("listaTa
6     listaTarefas.innerHTML += `<p>${novaTarefa.value}</p>`
7   }
8 }
9
```

Linha 3, coluna 18 Cobertura: N/A

Escopo Observação

Local

- this: Window
- novaTarefa: input#novaTarefa**

Global Window

Pausado no ponto de interrupção

Pontos de interrupção

- ☐ Pausar quando houver exceções não detectadas
- ☐ Pausar nas exceções encontradas

script.js

- ☒ if (novaTarefa.value) { 3

Pilha de chamadas

- cadastrar script.js:3
- onclick (índice):17

Pontos de interrupção de XHR/busca

Pontos de interrupção DOM

Listeners globais

Pontos de interrupção do listener de eventos

Pontos de interrupção da violação de CSP