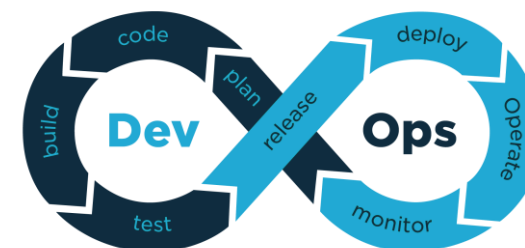


# Programação Web

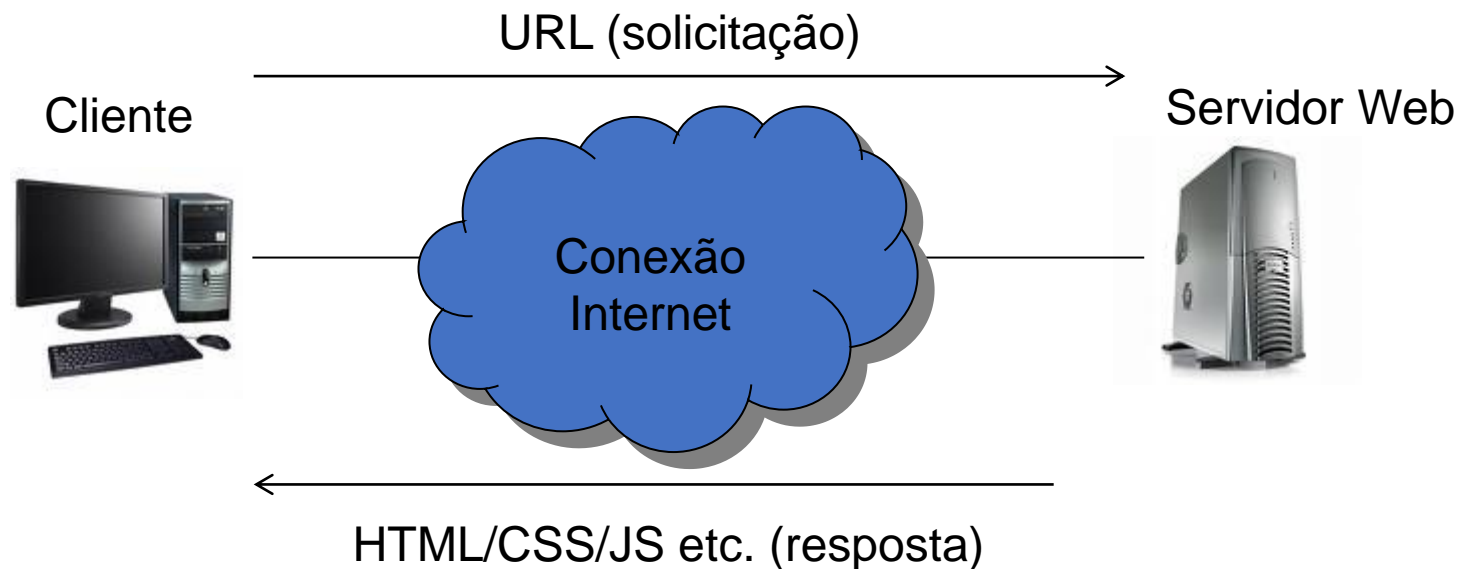
- ✓ Introdução: Conceitos Iniciais
- ✓ Exemplos



E tem mais...



- Pensando nos elementos que fornecem informações através da Web e seu funcionamento básico, temos:



- ✓ É a abreviação de **Uniform Resource Locator**, seria o endereço para um recurso da Internet.
- ✓ É composto por:

1. Identificador do Protocolo
2. Nome do recurso

- Pode conter
  1. Nome do host
  2. Nome do arquivo
  3. Porta do serviço
  4. Referência

**protocolo**:://**domínio**:**porta**/**caminho**/**recurso**?**query\_string**#**referência**

- ✓ Exemplos:  
ftp://www.cruzeirodosul.edu.br/aula.exe  
http://www.cruzeirodosul.edu.br/index.html  
https://www.cruzeirodosul.edu.br/pesquisa.jsp?q=java  
http://www.cruzeirodosul.edu.br/index.html#introducao

- ✓ O que preciso saber?

Cliente



HTML/CSS  
JavaScript  
XML  
JSON

...

**Disciplina:**  
Aplicações para Internet  
**Programação Web**

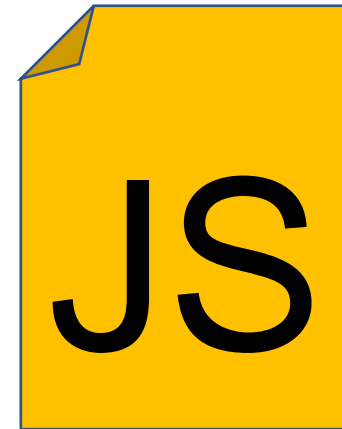
Servidor



Banco de Dados  
Linguagens como JSP,  
PHP, tec. .net etc.

As demais disciplinas  
do curso irão abordar  
esse modelo.

- ✓ HTML (Hypertext Markup Language)
- ✓ CSS (Cascading Style Sheet)
- ✓ JavaScript



- ✓ Hypertext Markup Language  
ou  
Linguagem de Marcação de Hipertexto
- ✓ O HTML é uma linguagem que se propõe em distribuir informação globalmente e pode ser entendido por diversos meios de acesso.
- ✓ Desenvolvido originalmente por Tim Bernes-Lee (criador da World Wide Web, 1989)
- ✓ Um documento HTML é um documento texto que pode ser produzido utilizando qualquer editor de texto. É utilizado por praticamente todas as páginas WEB disponíveis na Internet.
- ✓ O conteúdo HTML de uma página é processado por um Navegador (Internet Explorer, Chrome, Firefox etc.)
- ✓ O HTML define um conjunto de elementos para a marcação de uma página Web: cabeçalho, parágrafo, lista, tabelas, entre outros. Cada elemento possui sua função específica e são comumente chamados de tag (marca ou marcadores).



**Tim Bernes-Lee 1955 - ...**





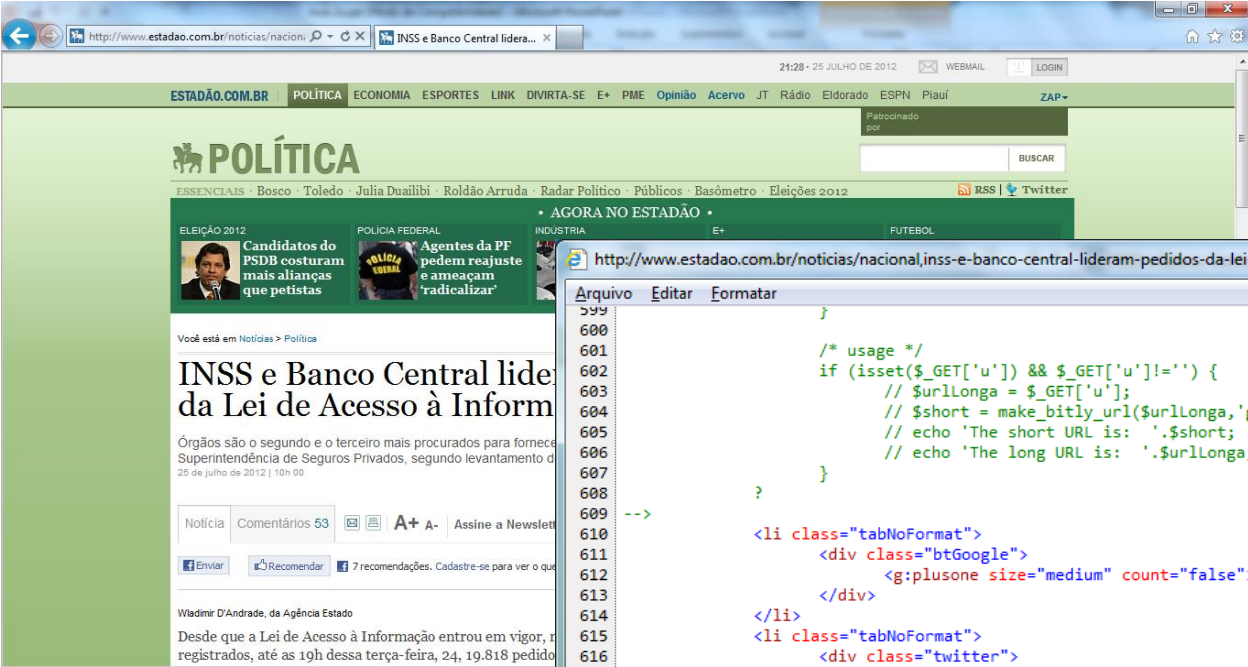
**Eu: Vou aprender JavaScript em 1 mês ser FullStack**

**Javascript:**





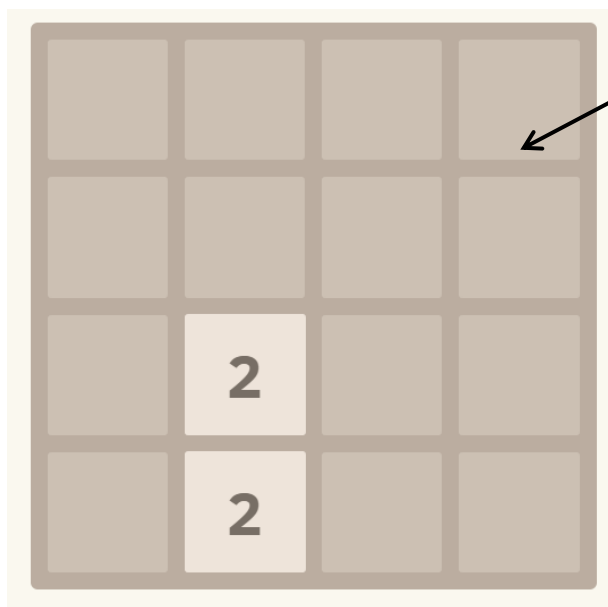
← Página WEB



Código Fonte →

```
599 }
600
601 /* usage */
602 if (isset($_GET['u']) && $_GET['u']!='') {
603     // $urlLonga = $_GET['u'];
604     // $short = make_bitly_url($urlLonga,'grupoestado','R_58294e4bf2c7a5eee0cdccc0fd57dd6','xml');
605     // echo 'The short URL is: '.$short;
606     // echo 'The long URL is: '.$urlLonga;
607 }
608
609 -->
610 <li class="tabNoFormat">
611     <div class="btGoogle">
612         <g:plusone size="medium" count="false"></g:plusone>
613     </div>
614 </li>
615 <li class="tabNoFormat">
616     <div class="twitter">
617         <a href="http://twitter.com/share" class="twitter-share-button" data-text="" data-count="horizontal" data-
618 via="estadao" id="linkTweet">Tweet</a>
619         <script type="text/javascript">$('#linkTweet').attr('data-text', title); </script>
620         <script type="text/javascript" src="http://platform.twitter.com/widgets.js"></script>
621     </div>
622 </li>
623 </ul>
624 </div>
625 <div class="box-compartilhe">
626
627     <!-- div class="facebook">
628         <a name="fb_share" type="button" href="http://www.facebook.com/sharer.php"></a>
```

<http://gabrielecirulli.github.io/2048/>



Jogo visualizado no Chrome

```
function Grid(size, previousState) {
  this.size = size;
  this.cells = previousState ? this.fromState(previousState) : this.empty();
}

// Build a grid of the specified size
Grid.prototype.empty = function () {
  var cells = [];

  for (var x = 0; x < this.size; x++) {
    var row = cells[x] = [];

    for (var y = 0; y < this.size; y++) {
      row.push(null);
    }
  }

  return cells;
};

Grid.prototype.fromState = function (state) {
  var cells = [];

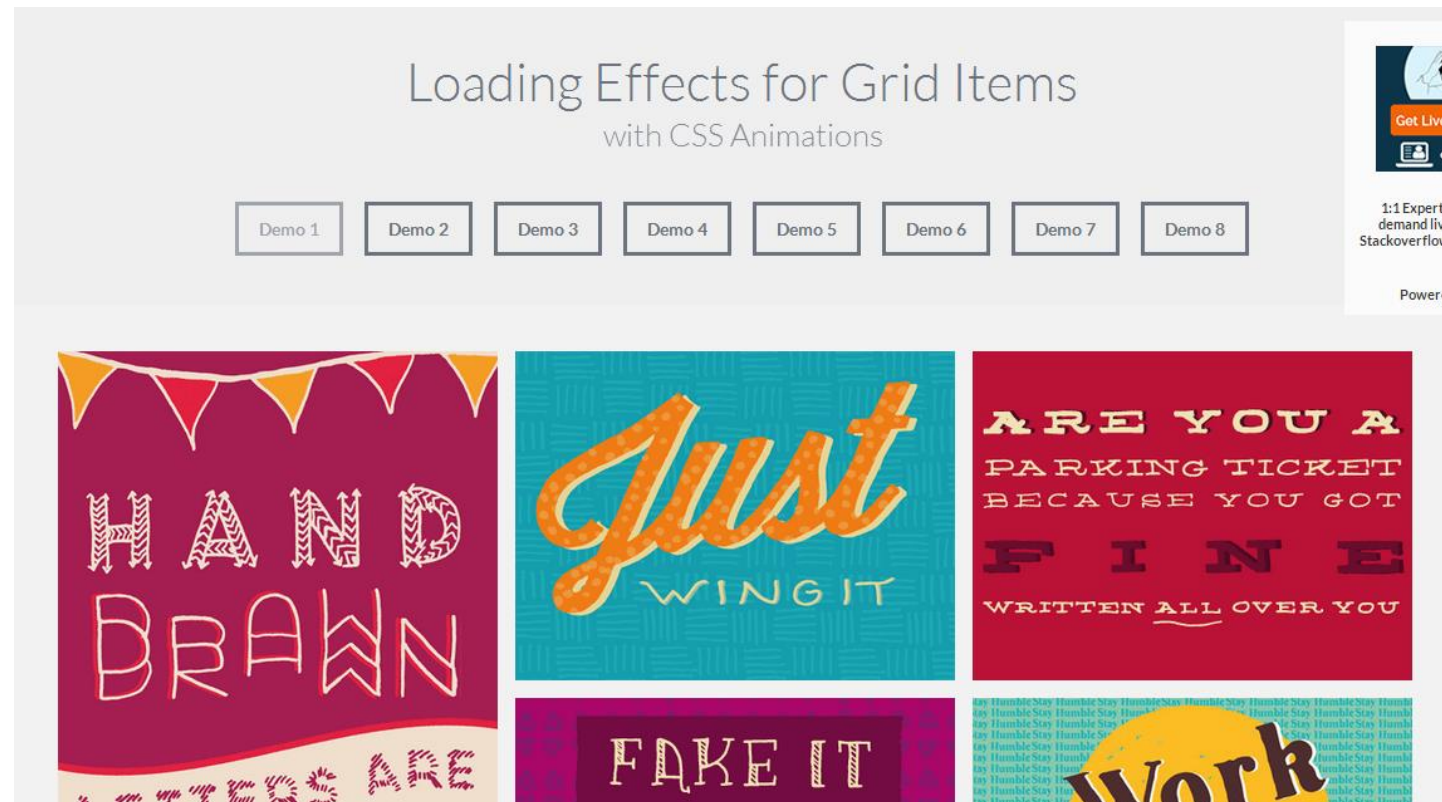
  for (var x = 0; x < this.size; x++) {
    var row = cells[x] = [];

    for (var y = 0; y < this.size; y++) {
      var tile = state[x][y];
      row.push(tile ? new Tile(tile.position, tile.value) : null);
    }
  }

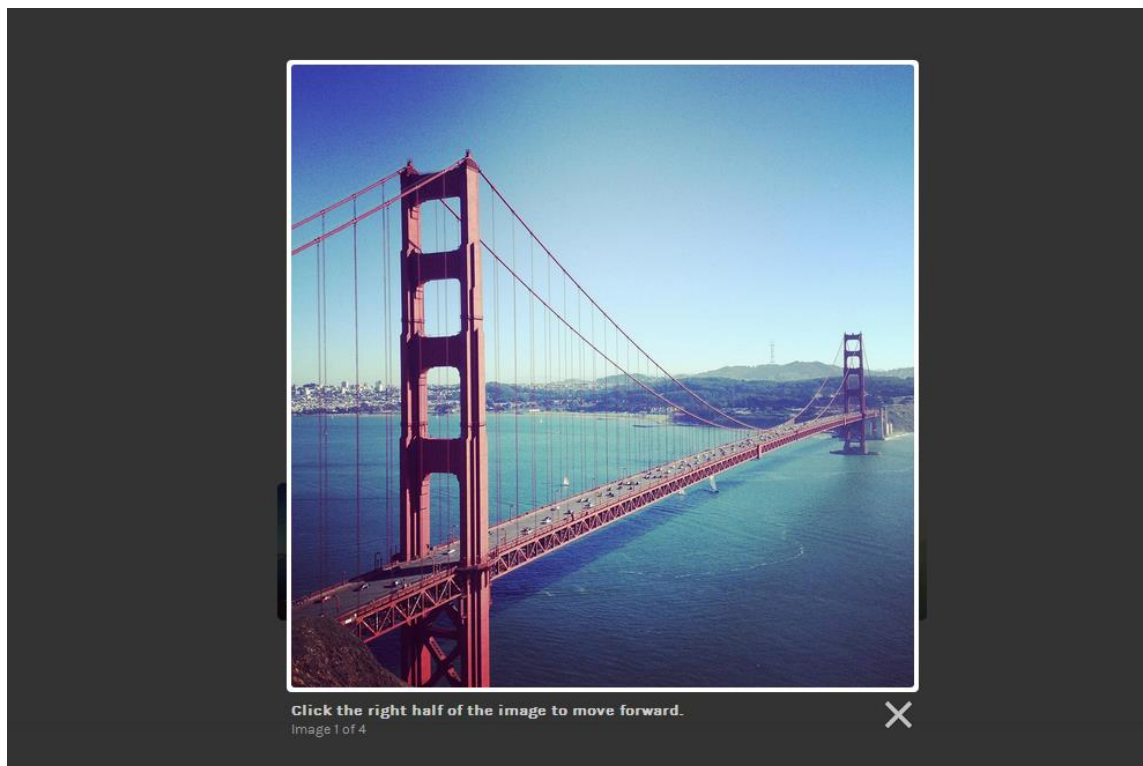
  return cells;
};
```

Código fonte

<http://tympanus.net/Development/GridLoadingEffects/index.html>



<http://lokeshdhakar.com/projects/lightbox2/>



<https://swiperjs.com/>

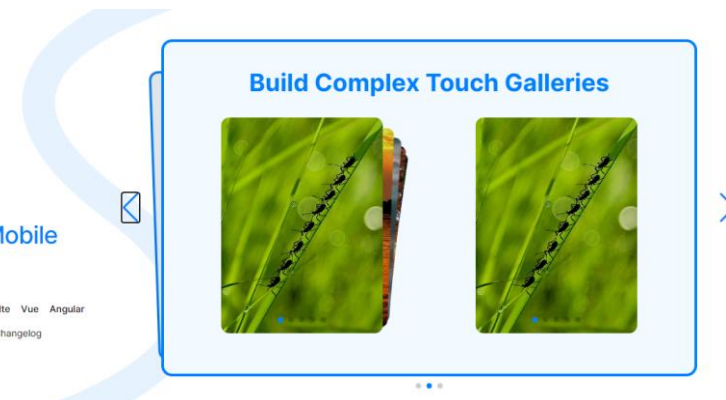


**Swiper**  
The Most Modern Mobile  
Touch Slider


[Get Started](#) [API](#) [Demos](#) [React](#) [Solid](#) [Svelte](#) [Vue](#) [Angular](#)

MIT Licensed, v8.3.2 released on July 26, 2022 [Changelog](#)

32,140 stars



<https://developers.google.com/chart/interactive/docs/gallery?hl=pt-br>



Google Charts

alcidestbjr@gmail.com Sair

Produtos > Google Charts

Google Charts 8+1 10

• Overview

• Chart Gallery

Miscellaneous Examples

Annotation Charts New!

Area Charts

Bar Charts

Bubble Charts

Calendar Charts

Candlestick Charts

Column Charts

Combo Charts

Diff Charts

Gauge Charts

Geo Charts

Histograms

Intervals

Line Charts

Maps

Org Charts


Pie Charts

## Chart Gallery

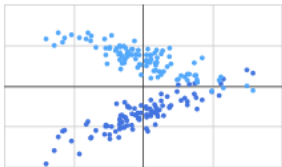
Our gallery provides a variety of charts designed to address your data visualization needs. These charts are based on pure HTML5/SVG technology (adopting VML for old IE versions), so no plugins are required. All of them are interactive, and many are pannable and zoomable. Adding these charts to your page can be done in [a few simple steps](#).

Some additional community-contributed charts can be found on the [Additional Charts](#) page.

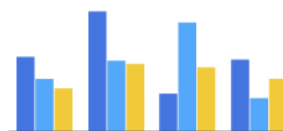
### Geo Chart




### Scatter Chart




### Column Chart




### Histogram



### Bar Chart



### Combo Chart





## Data-Driven Documents

<https://d3js.org/>



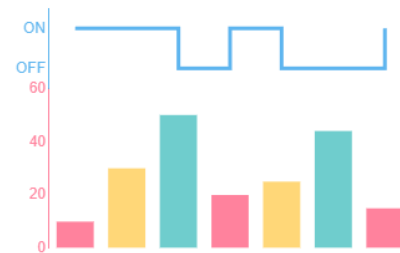


<https://uber.github.io/react-vis/>



## Chart.js

Simple yet flexible JavaScript charting for designers & developers

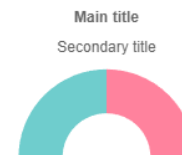
[Get Started](#)[Samples](#)[Ecosystem](#)[GitHub](#)

### New in 3.5 Scale stacking

Layout boxes can be stacked and weighted in groups.

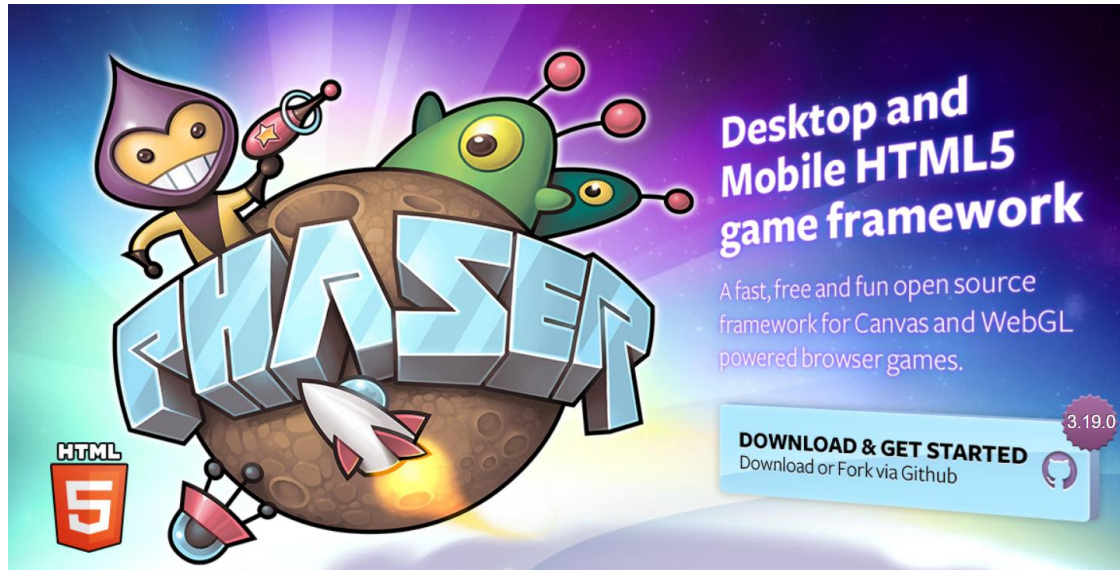
### New in 3.4 Subtitle plugin

A secondary title plugin with all the same options



<https://www.chartjs.org/>





<https://phaser.io/>



<http://www.melonjs.org/>

# ONOFF



★ START GAME ★

CONTROLS

EDITOR

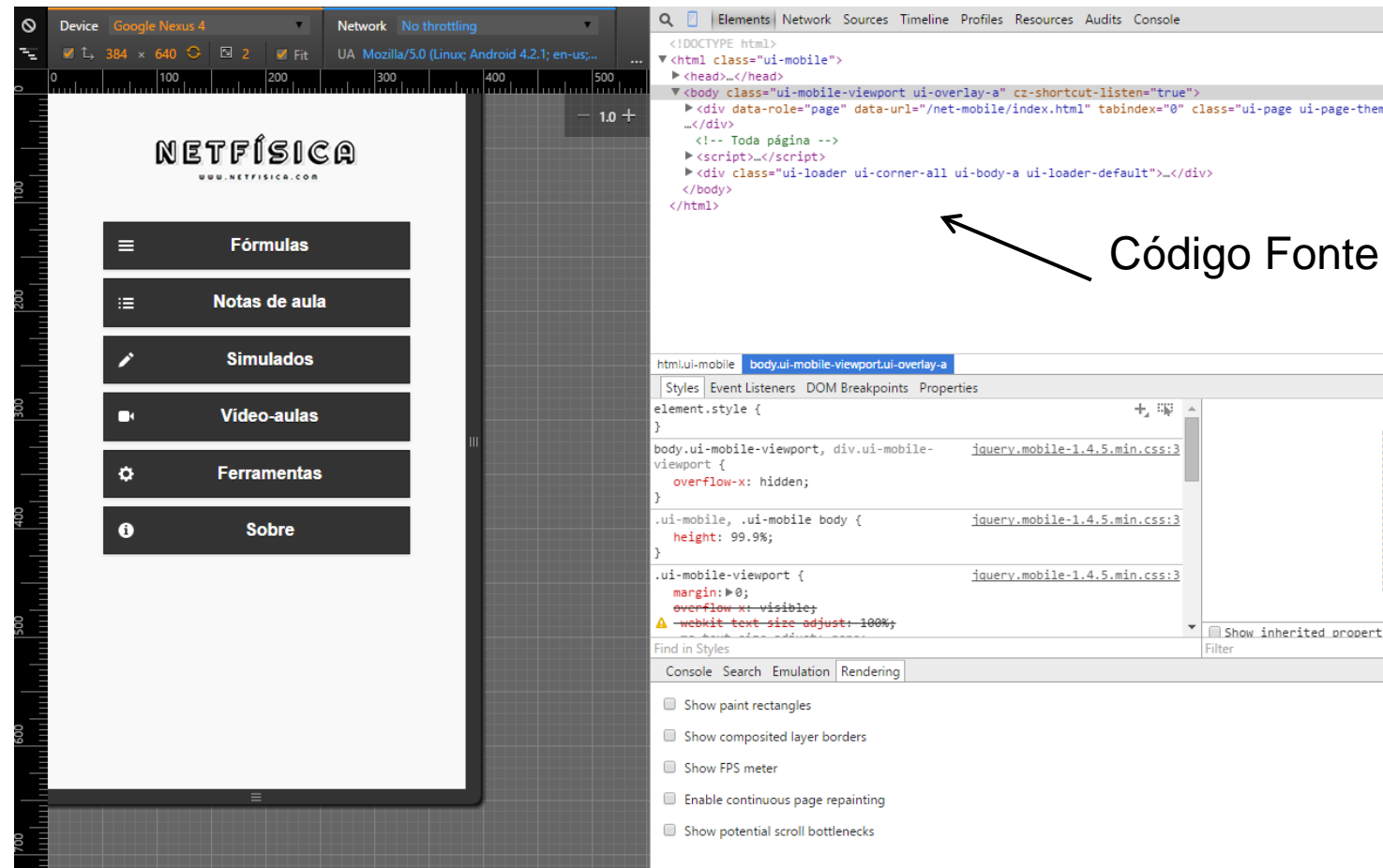
JS13KGAMES 2018 ■ BRAD DUNBAR ■ DANIEL MARINO



<https://js13kgames.com/games/onoff/index.html>



[https://developer.mozilla.org/en-US/docs/Games/Tutorials/2D\\_Breakout\\_game\\_pure JavaScript](https://developer.mozilla.org/en-US/docs/Games/Tutorials/2D_Breakout_game_pure_JavaScript)



- ✔ Quando abrimos uma página HTML em um navegador, o navegador irá interpretar as tags, ou seja, irá fazer uma análise sintática, definindo como cada parte do texto será apresentada.
- ✔ Um problema comum no desenvolvimento de páginas HTML desde o seu surgimento e de outras tecnologias como CSS, é a compatibilidade entre os navegadores, ou seja, uma página desenvolvida em HTML5 pode rodar somente no Chrome, por exemplo, ou aparecer de um jeito no Internet Explorer e de outro no Firefox. Também temos que tomar cuidado quando falamos em dispositivos ou sistemas diferentes, como visualizar nossa página em um celular ou tablet.



```
<!DOCTYPE html>

<html>

<head>
  <meta content="text/html" charset="utf-8">
  <title>Texto na barra de título</title>
</head>

<body>
  <h1>Título tamanho maior</h1>
  <h2>Título tamanho um pouco menor</h2>
  <h4>Título tamanho menor ainda</h4>
  <br/>
  <hr/>
  
</body>

</html>
```



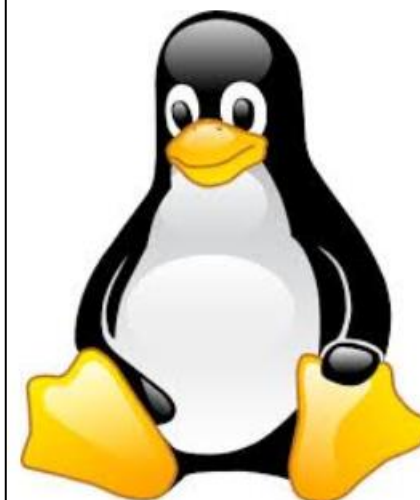
código HTML

**Título tamanho maior**

**Título tamanho um pouco menor**

Título tamanho menor ainda

---



página aberta no navegador

Trabalharemos com HTML + CSS + JavaScript

```
<!DOCTYPE html>
<script>
    console.log("Primeira aula de WEB");
</script>
<style type="text/css">
    body{
        color: #0F0;
        font-size:30pt;
    }
</style>
<html>
<head>
    <title>Título do documento</title>
</head>
<body>
    Todo o conteúdo da página vai dentro da tag BODY.
</body>
</html>
```

Vamos inserir esse código no Visual Studio Code para iniciarmos nosso contato com as linguagens e a ferramenta.

Lembre-se de salvar seu arquivo com a extensão HTML e posteriormente teste seu arquivo em algum navegador.





<https://nodejs.org/en/>

Interpretador JS orientado a eventos. Permite criar aplicações de rede escalonáveis. Migra a programação JS do cliente para o servidor. Baseado no Interpretador V8 (Google, feito em C++). Foi criado por Ryan Dahl em 2009.

PWA - Progressive Web App (aplicativo progressivo para a web)

Experiência instalável, como um aplicativo, em computadores e dispositivos móveis, a diferença é que são criados, entregues e mantidos diretamente pela Web.

<https://electronjs.org/>



Permite construir aplicações HTML, CSS e JS para desktop



Editor de Texto (JS, HTML e CSS)





## Exemplos (códigos) no GitHub

[https://github.com/alcidestbj/Programacao\\_Web](https://github.com/alcidestbj/Programacao_Web)

- ★ Hello World – do GitHub  
<https://guides.github.com/activities/hello-world/>
- ★ GitHub Training & Guides  
<https://www.youtube.com/githubguides>

## O Que é GitHub e Para Que é Usado?

<https://www.hostinger.com.br/tutoriais/o-que-github/>

## Tutorial do GIT Básico – Introdução ao GIT

<https://www.hostinger.com.br/tutoriais/tutorial-do-git-basics-introducao/>