

Web Performance Hacks 🧐

@saurabhdaware



Saurabh Daware



- Workz at da Student
- Open Source Fanboi 🌻
- Previously (Intern) -



Bew Digital



@saurabhcodes



/saurabhdaware



saurabhdaware.in

Ready?

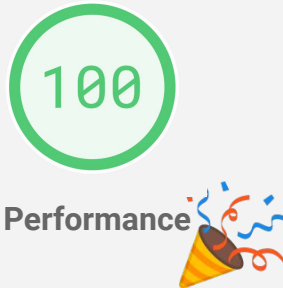


Hack #1



Replace your whole code with...

```
<html>  
  <body>Hello, World!</body>  
</html>
```



How does the web work?



https://saurabhdaware.in

Server

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="Hi, I'm Saurabh Daware. A JavaScript Developer ba
projects, articles, and other information about me. I have created projectman, pwainit,
  <meta name="keywords" content="portfolio site, saurabh, saurabh daware, daware, per
  <meta name="mobile-web-app-capable" content="yes">
  <meta name="author" content="Saurabh Daware" />
  <meta name="copyright" content="Saurabh Daware" />
  <meta name="robots" content="follow"/>

  <link rel="icon" sizes=192x192 href="assets/images/logo-192.png">
  <meta name="theme-color" content="#011f41" />
  <link rel="manifest" href="manifest.json">

  <!-- OPEN GRAPH -->
  <meta property="og:type" content="website" />
  <meta property="og:title" content="Saurabh Daware">
  <meta property="og:url" content="https://saurabhdaware.in">

  <meta name="twitter:card" content="summary_large_image"></meta>
  <meta name="twitter:creator" content="@saurabhcodes"></meta>
```

Parsing &
Rendering

Hi,
I'm **Saurabh Daware**
I make random things, mostly using
JavaScript.



Scroll Down



HTML Parsing & Rendering

(a zoomed out view)



HTML



Tokenization



DOM Tree

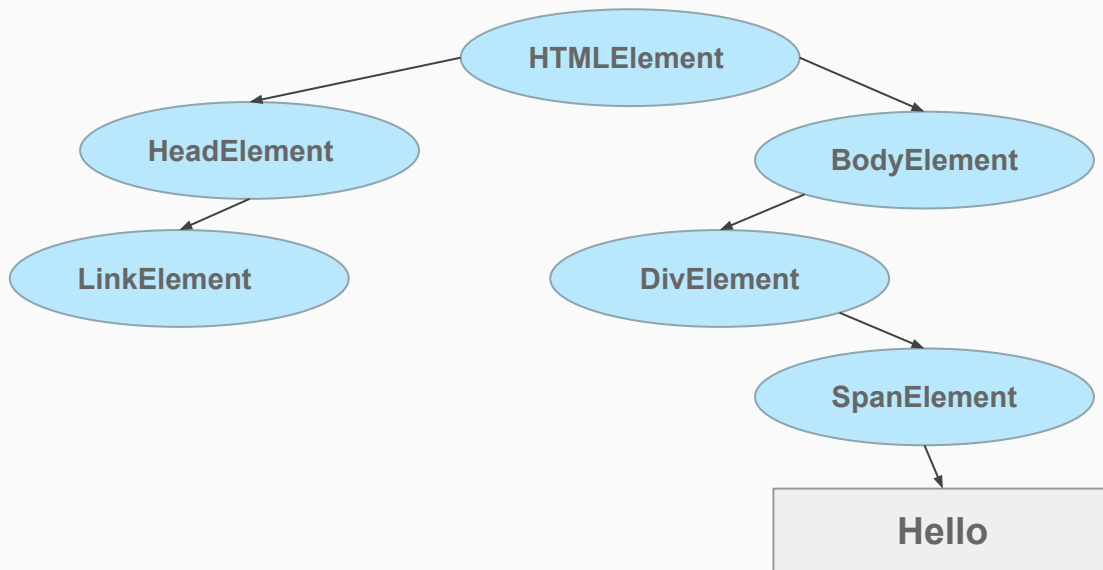
```
<html>
```

```
<head>...</head>
```

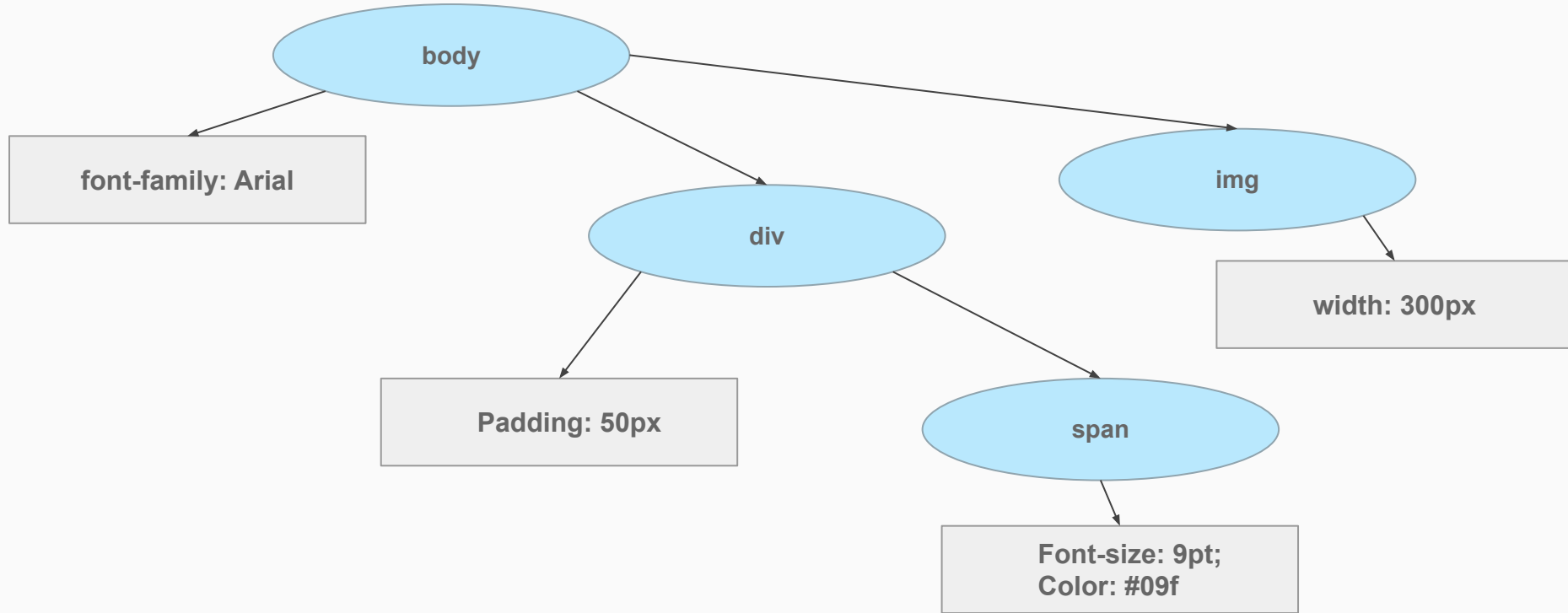
```
<body><div><span>Hello</span></div></body>
```

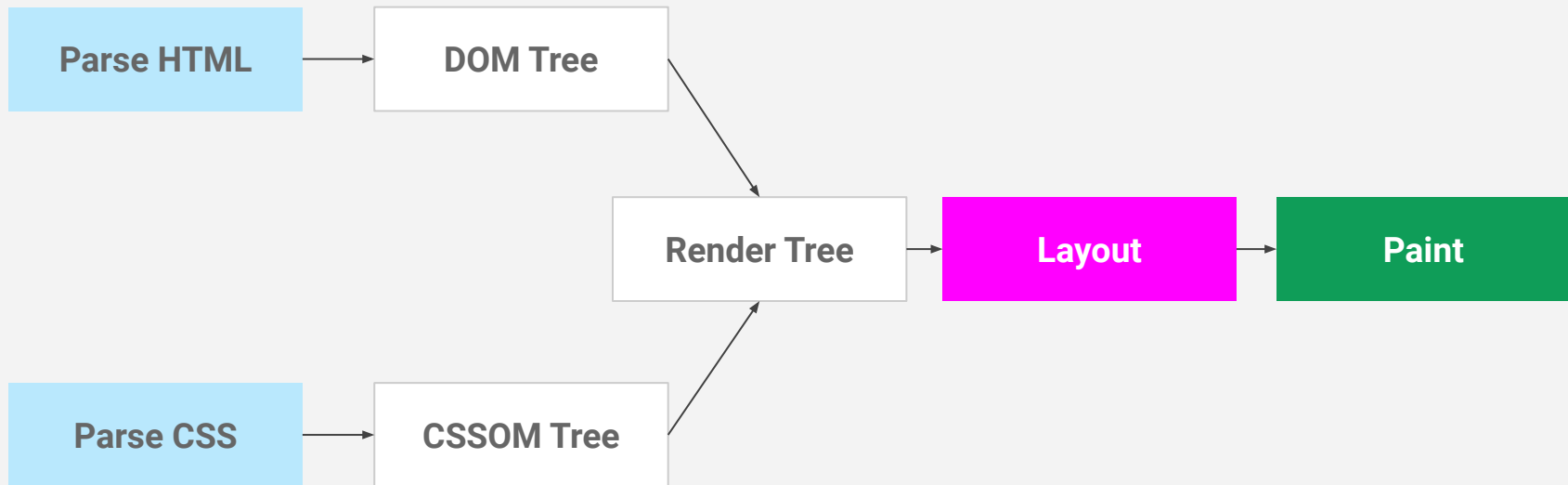
```
</html>
```

[Start: **HTML**] [Start: **HEAD**] ... [End: **HEAD**] [Start: **BODY**] [Start: **DIV**] [Start: **SPAN**]



CSSOM Tree





Parsing halts when it comes across `<link>`, `<script>`, and `<style>` tags.



- **CSS**
- **JavaScript**
- **Images**



'A quick First Paint' hack



Lets Code!!!!

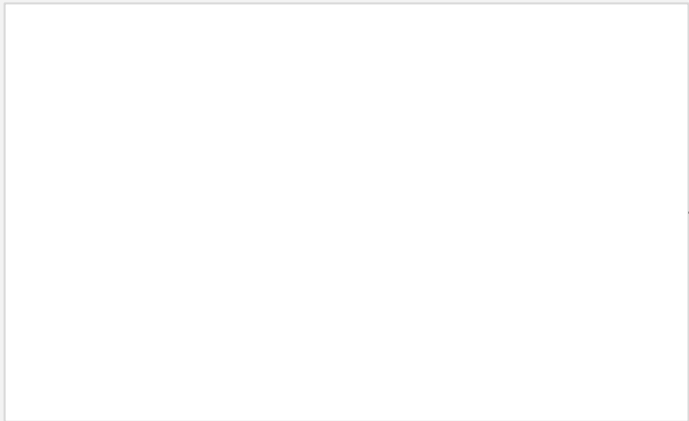


Without CSS



```
<html>
  <head>
    <!-- <link rel="stylesheet" href="styles/main.css" /> -->
  </head>
  <body>
    <header>Wubba Luba Dub Dub!</header>
    <main>...</main>
  </body>
</html>
```





CoolWebsite

Wubba lubba dub dub!

Random Form

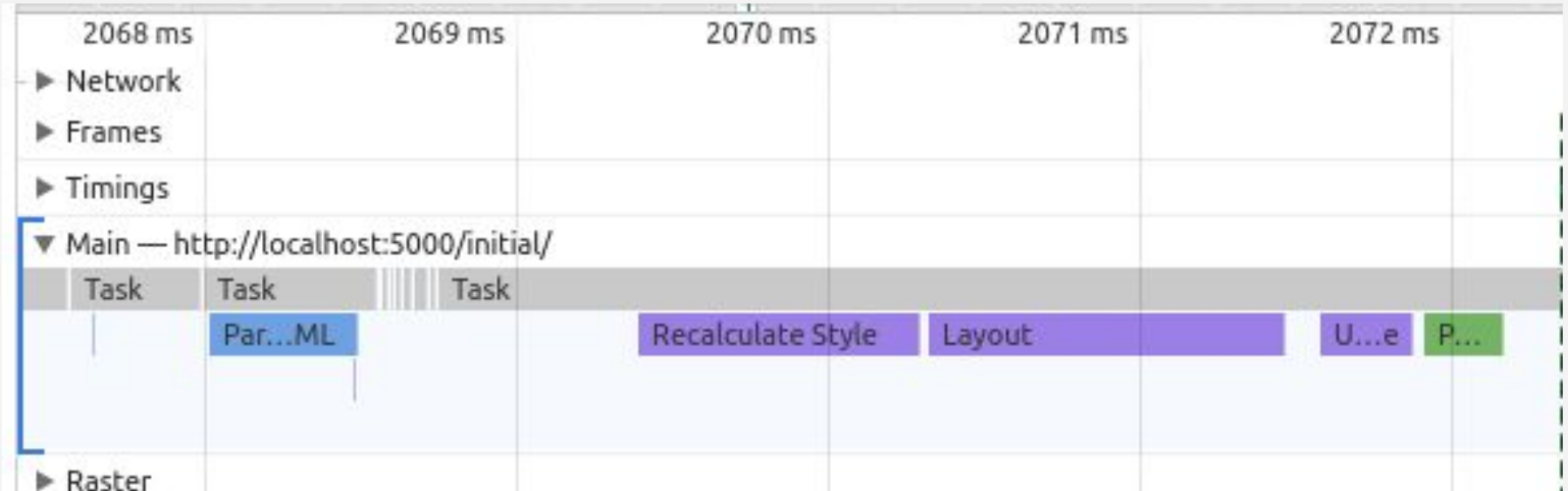
Name:

Email:

About

Lorem ipsum dolor sit amet consectetur adipisicing elit. Ex, quibusdam volupta
dolorum natus hic illum iste! Obcaecati incidunt odit rerum dolor ipsum similique
Explicabo impedit earum fuga praesentium?

2s



With CSS

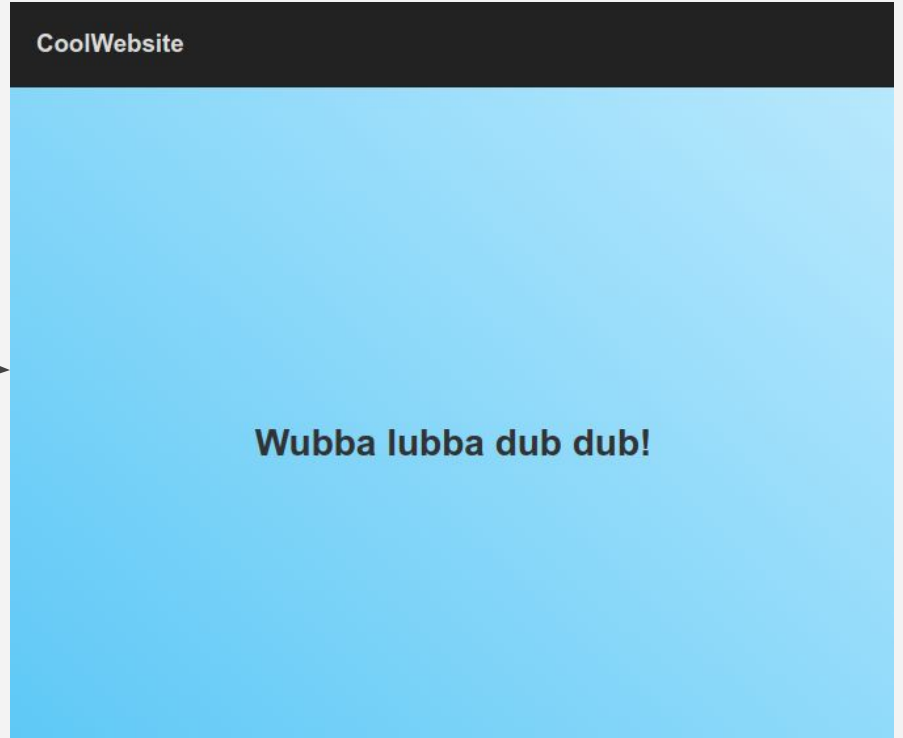


```
<html>
  <head>
    <link rel="stylesheet" href="styles/main.css" />
  </head>
  <body>
    <header>Wubba Luba Dub Dub!</header>
    <main>...</main>
  </body>
</html>
```



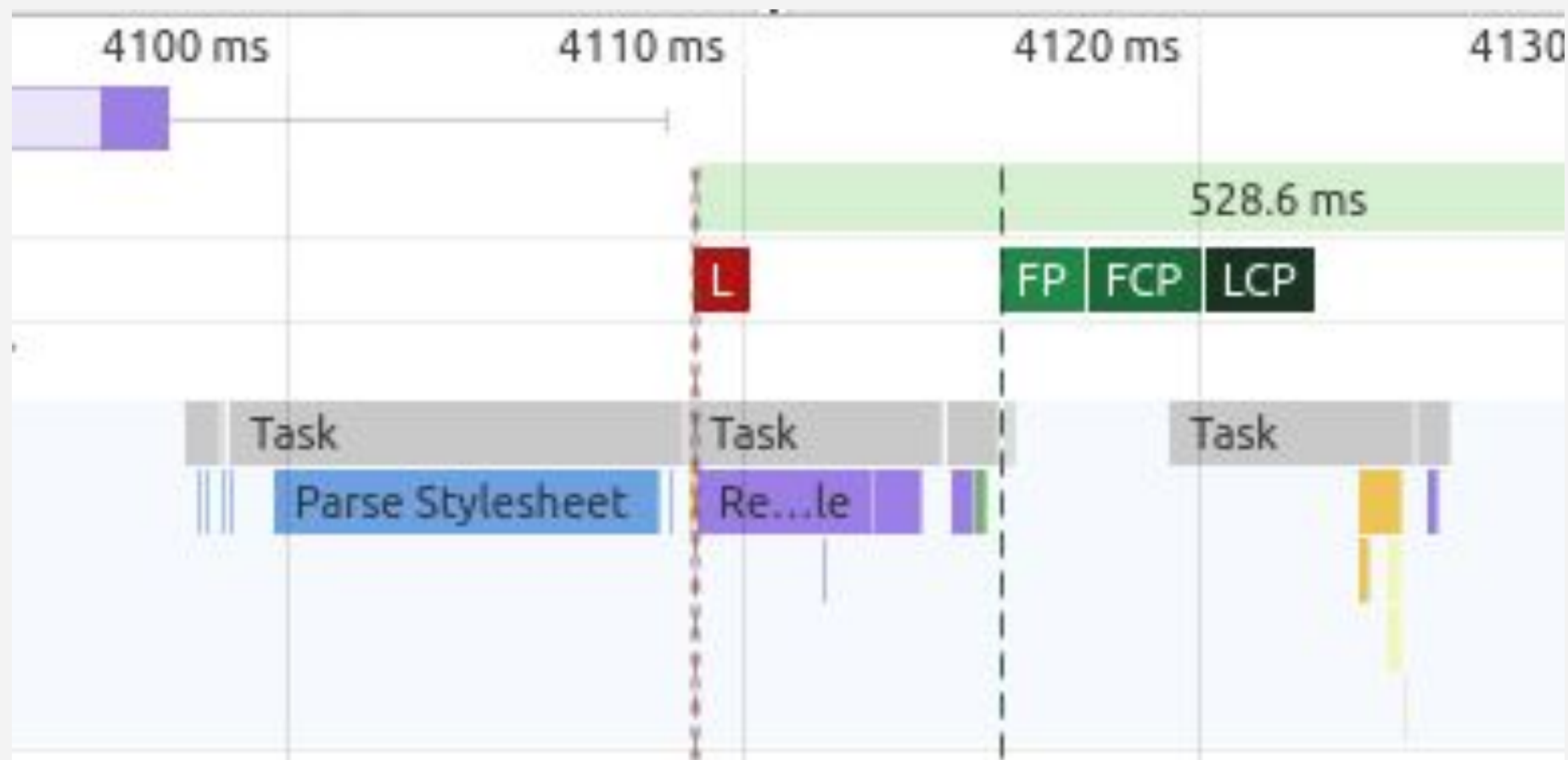


0s - 3.9s



4s





Render waits for CSS to load



**How about rendering the HTML content first
and painting on screen and then applying CSS?**





```
<html>
  <head>
    <link
      rel="preload" href="styles/main.css" as="style"
      onload="this.rel='stylesheet'; this.onload=null"
    />
  </head>
  <body>
    <header>Wubba Luba Dub Dub!</header>
    <main>...</main>
  </body>
</html>
```





0s - 1.9s

CoolWebsite

Wubba lubba dub dub!

Random Form

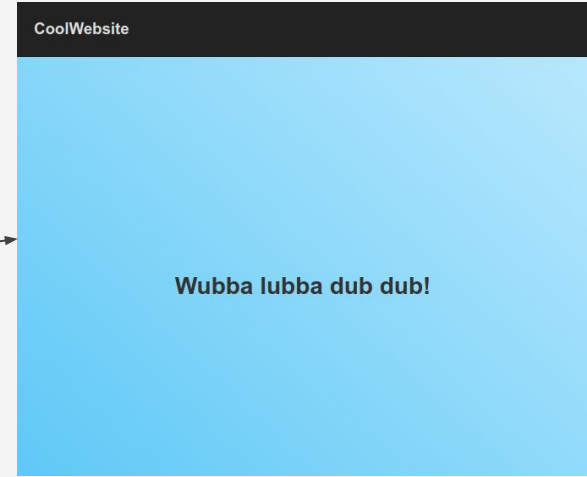
Name:

Email:

About

Lorem ipsum dolor sit amet consectetur adipisicing elit. Ex, quibusdam volupta
dolorum natus hic illum iste! Obcaecati incididunt odit rerum dolor ipsum similique
Explicabo impedit earum fuga praesentium?

2s - 3.9s



4s



Just showing HTML is bad UX



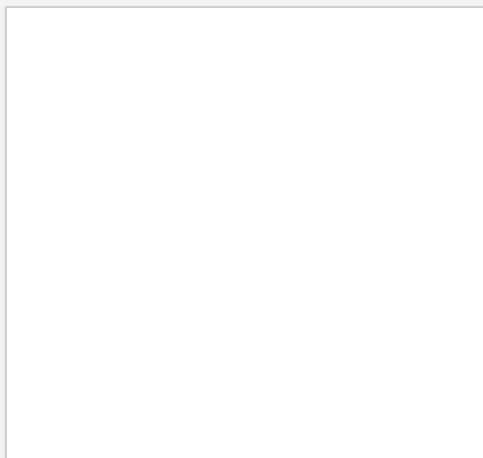
Inlining critical CSS



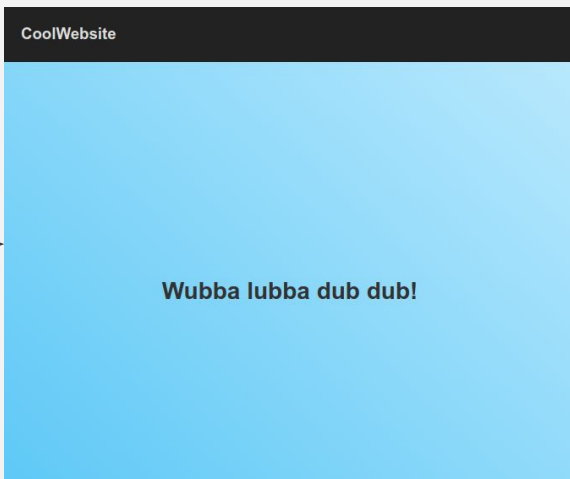


```
<html>
  <head>
    <link
      rel="preload" href="styles/main.css" as="style"
      onload="this.rel='stylesheet'; this.onload=null"
    />
    <style>
      header{/* styles of header */}
    </style>
  </head>
  <body>
    <header>Wubba Luba Dub Dub!</header>
    <main>...</main>
  </body>
</html>
```



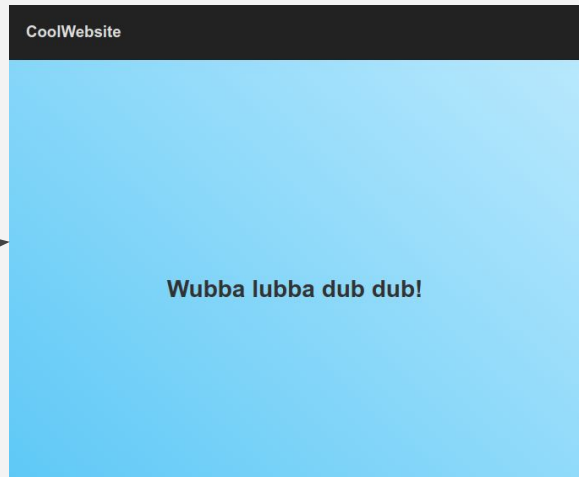


0s - 1.9s



2s - 3.9s

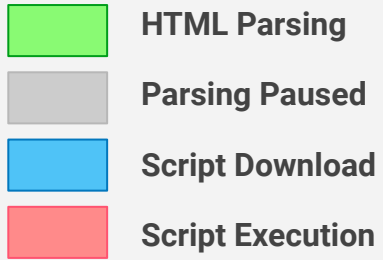
Only header css without
css for other content
outside the viewport



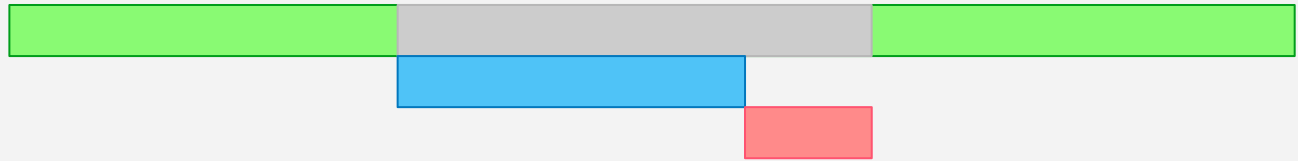
4s

Full CSS

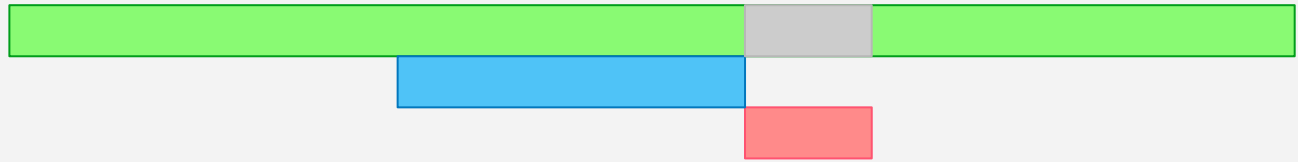




<script>



<script async>



<script defer>



Code-splitting



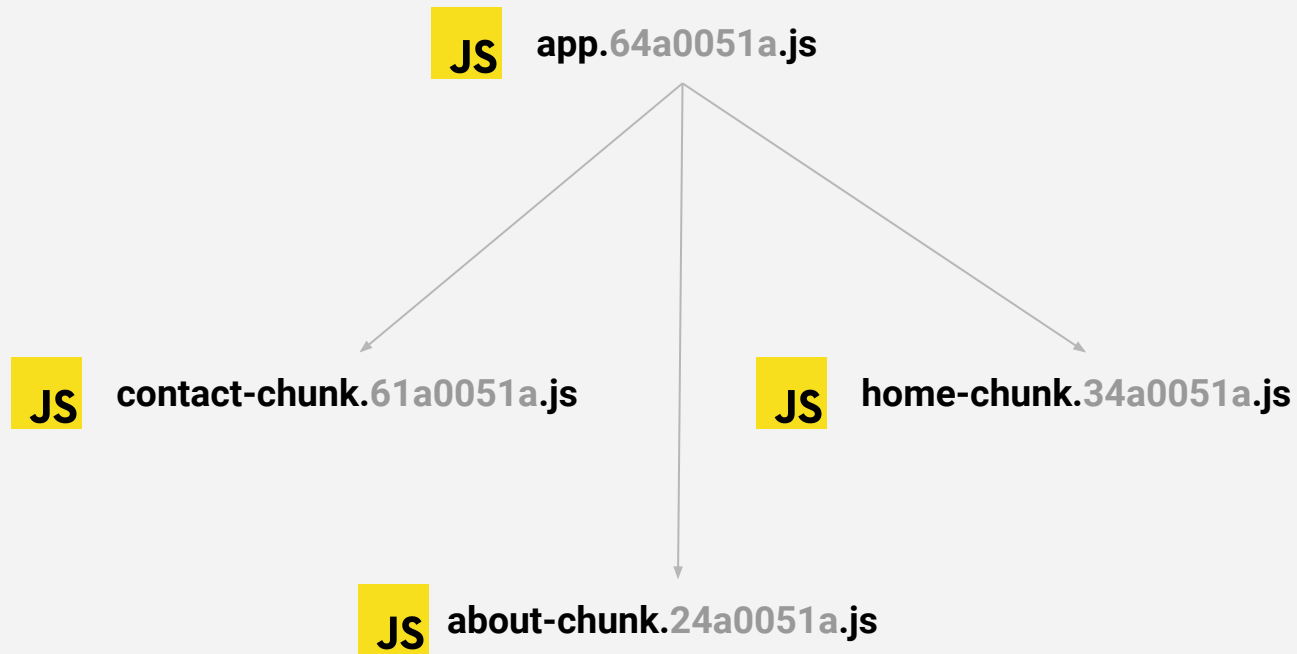


Image Optimization



```

```



Lazy loading via attribute for images & iframes - UNOFF

The `loading` attributing on images & iframes gives authors control over when the browser should start loading the resource.

| Current aligned | | Usage relative | | Date relative | | Apply filters | | Show all | | ? | | | | |
|-----------------|-------|----------------|--------|---------------|-------|---------------|--------|----------|------|-----------------|--------------|--------------------|---------------------|--------|
| IE | Edge | Firefox | Chrome | Safari | Opera | iOS | Safari | Opera | Mini | Android Browser | Opera Mobile | Chrome for Android | Firefox for Android | UC for |
| | | | 4-74 | | | | | | | | | | | |
| | | | 75 | | | | | | | | | | | |
| 6-10 | 12-18 | 2-71 | 76-78 | 3.1-12.1 | 10-63 | 3.2-13.1 | | | | 2.1-4.4.4 | 12-12.1 | | | |
| 11 | 79 | 72 | 79 | 13 | 64 | 13.2 | | all | | 76 | 46 | 79 | 68 | |
| | | 73-74 | 80-82 | TP | | 13.3 | | | | | | | | |



Polyfill:

<https://github.com/mfranzke/loading-attribute-polyfill>



Progressive JPEGs



JPEG



Progressive JPEG

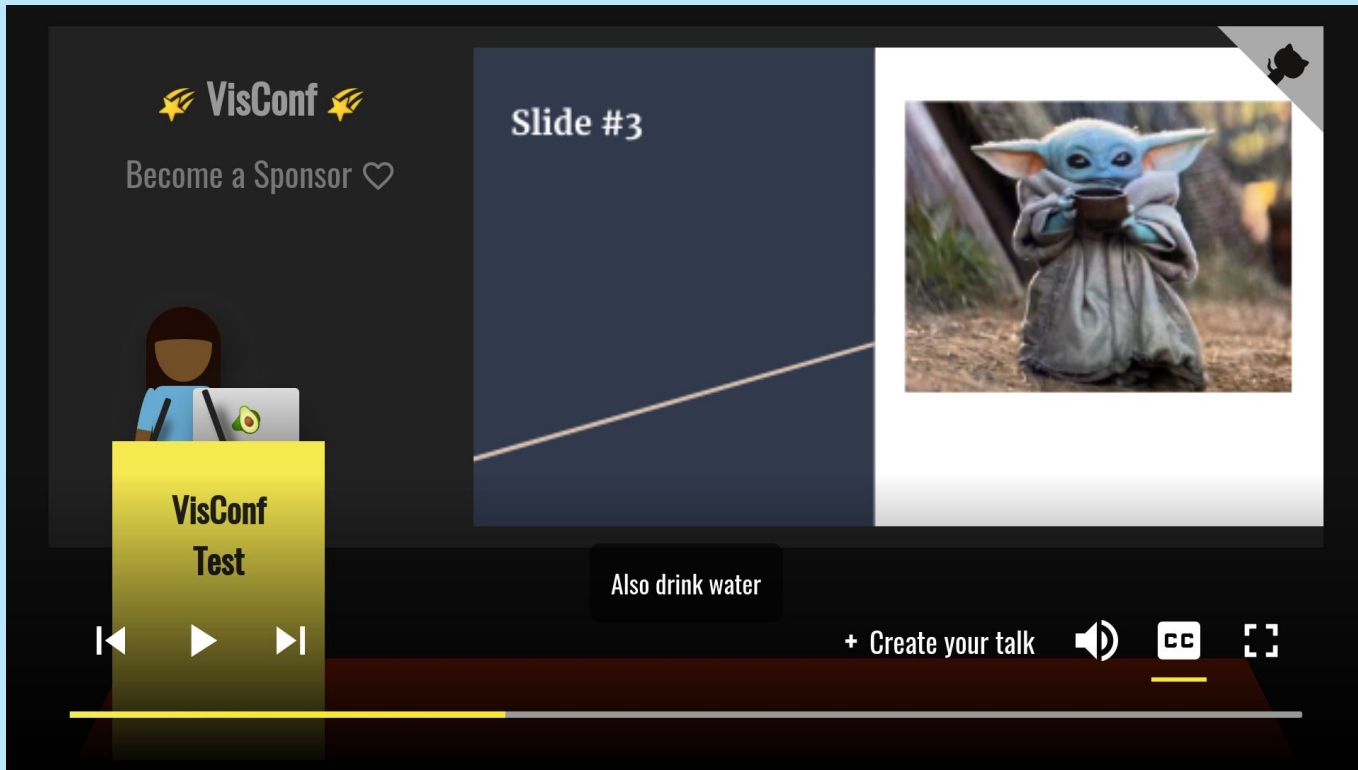


CDN



https://res.cloudinary.com/saurabhdaware/image/upload/c_scale,fl_progressive,w_300/v1526463112/Screenshot_87_baiu9r.jpg







<https://github.com/saurabhdaware/visconf>




Thank You!



 @saurabhcodes

  /saurabhdaware

 saurabhdaware.in



Extra Slides





Fancy Keywords to Google Search:

- Inlining Critical CSS
- QuickLinks (<https://github.com/GoogleChromeLabs/quicklink>)
- InstantClick (<http://instantclick.io/>)
- Web Workers, Web Assembly, OffScreenCanvas
- Service Workers
- Predictive Prefetching

Async vs Defer : <https://stackoverflow.com/questions/10808109/script-tag-async-defer>

Script attributes: <https://eager.io/blog/everything-i-know-about-the-script-tag/>

RenderTree Constructions:

<https://developers.google.com/web/fundamentals/performance/critical-rendering-path/render-tree-construction>

Rendering Behind the scene:

<https://blog.logrocket.com/how-browser-rendering-works-behind-the-scenes-6782b0e8fb10/>

Ryan Seddon:

<https://www.youtube.com/watch?v=SmE4OwHztCc>

JPEG to Progressive JPEG

<https://imagemagick.org/index.php>

