

07: Web Dev Studio

Review

Semantic HTML

CSS Selectors and Styling

CSS Box Model

Layouts with Flexbox and Grids

Working with External Libraries

External CSS Libraries

- **What are CSS libraries?**
 - Pre-written collections of styles, effects, and animations.
- **Benefits of CSS libraries**
 - Speed up development.
 - Ensure consistency across designs.
 - Offer responsive, cross-browser-compatible styles.
- **Popular libraries we'll cover:** Bootstrap, Tailwind CSS, Animate.css, and Hover.css.

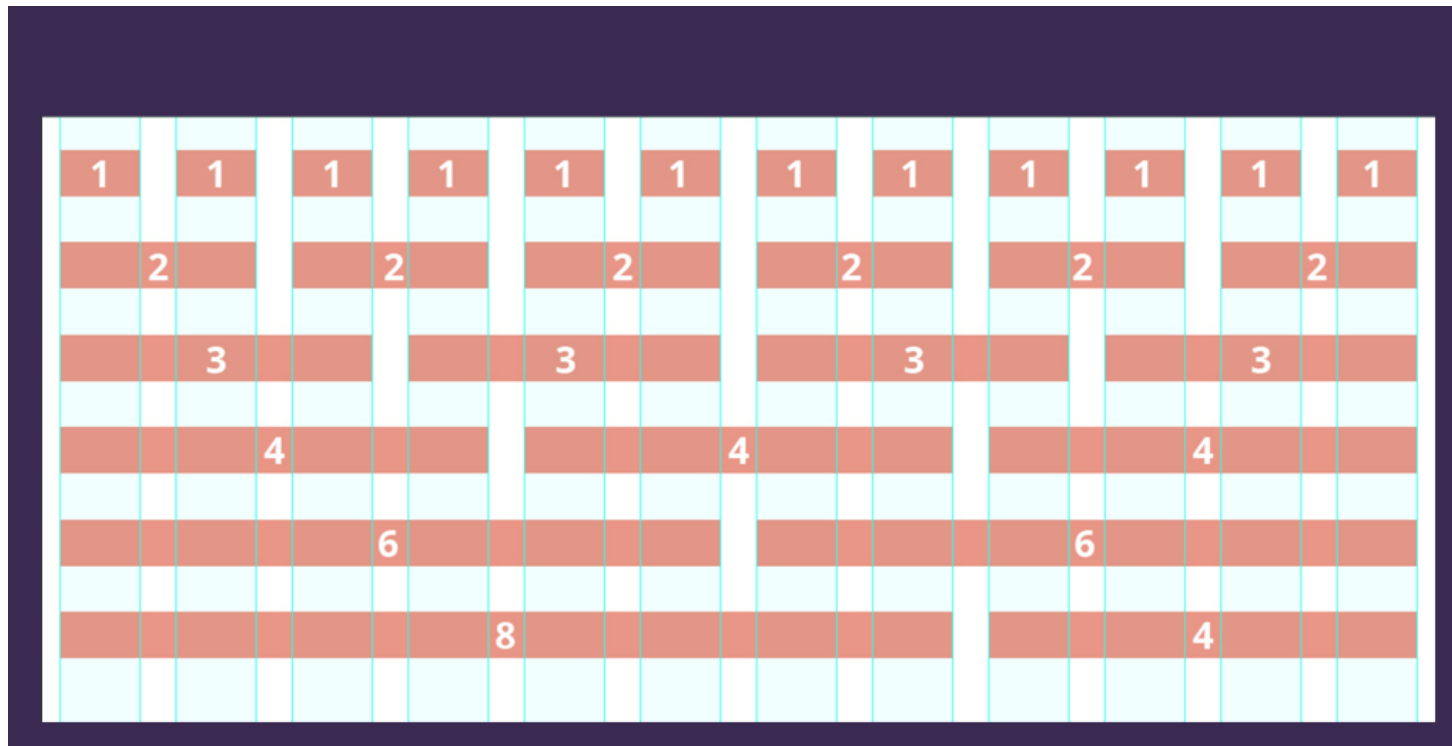
Including External CSS

- External CSS can be linked with the <link> tag just like our local CSS files
- Instead, most commonly we point the source to a CDN that's hosting the file

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">
```

Bootstrap

- Bootstrap is one of the most popular CSS libraries for creating consistent components and grid-based layouts
- It uses a 12-column grid system to create layouts



Bootstrap Components

- Bootstrap also offers a pre-defined set of styles and components with its library, along with variants of each component

Primary

Secondary

Success

Danger

Warning

Info

Light

Dark

[Link](#)

TailwindCSS

- Tailwind is a utility-based approach to defining styles
- It offers a short-hand way to write regular CSS properties within the HTML
- These classes are then compiled to output vanilla CSS
- Documentation: <https://tailwindcss.com/>



TailwindCSS

```
<button class="bg-blue-500 hover:bg-blue-700 text-white font-bold py-2 px-4 rounded">
```

Button

```
</button>
```

Animate.CSS

- Easy to use pre-defined animations
- Documentation: <https://animate.style/>

Animate.css

Just-add-water CSS animations

tada ▼

Animate it

[Download Animate.css](#) or [View on GitHub](#)

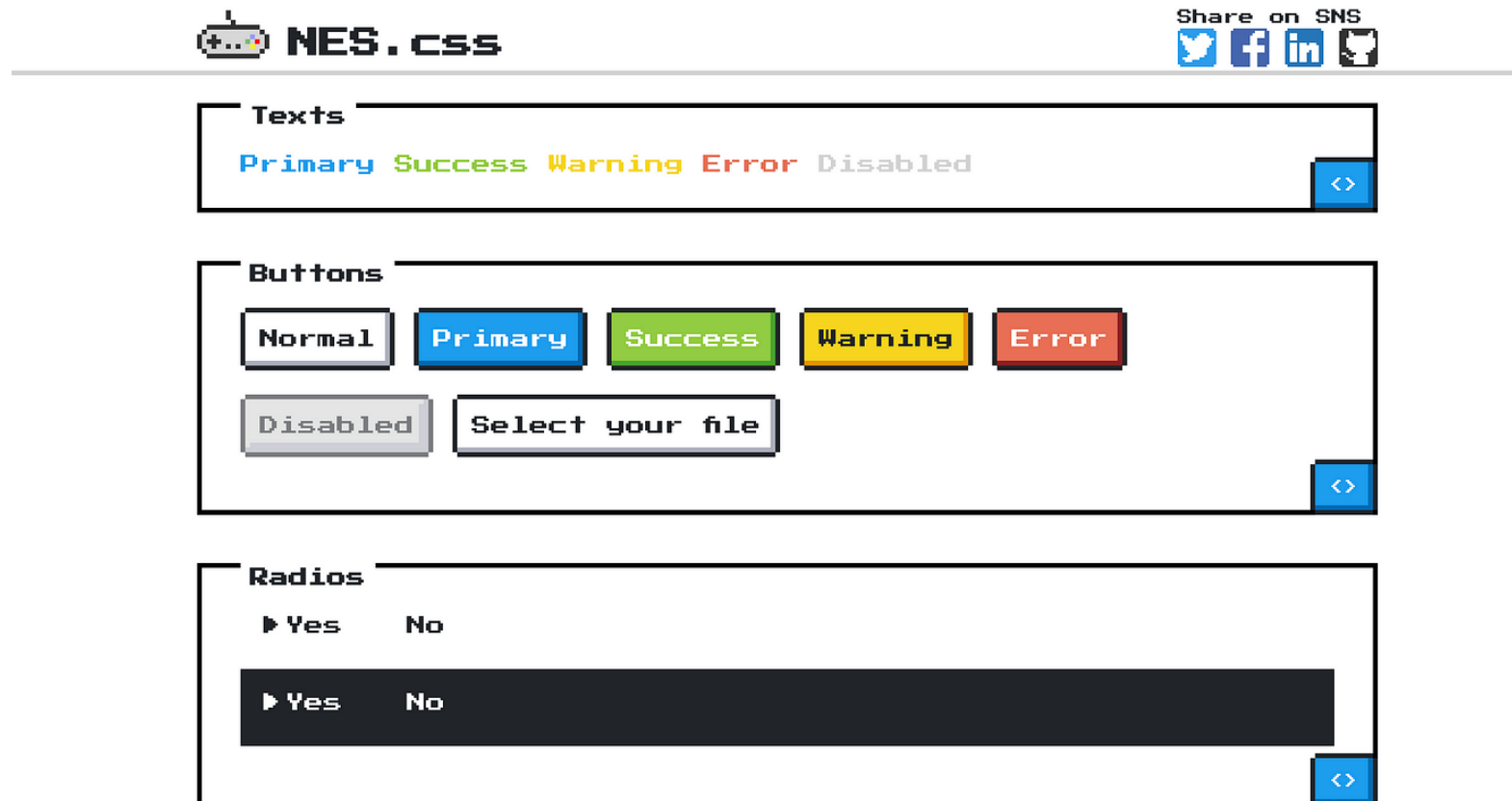
Hover.css

- Hover effects and transitions
- Documentation: <https://ianlunn.github.io/Hover/>

Button Hover

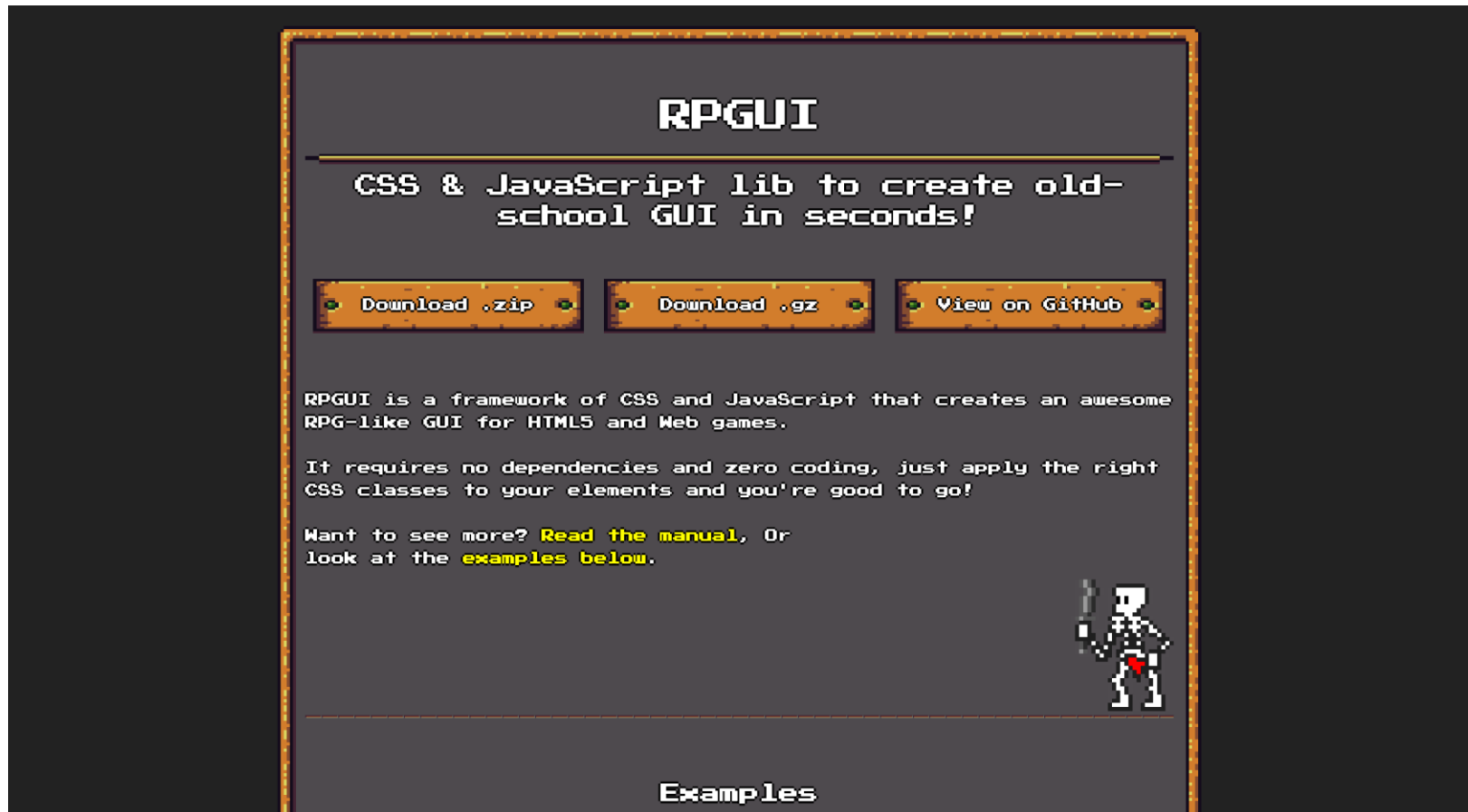
Specialized CSS Libraries

- NES.css - <https://nostalgic-css.github.io/NES.css/>



Specialized CSS Libraries

- RPGUI - <https://ronenness.github.io/RPGUI/>



Specialized CSS Libraries

- 98.css - <https://jdan.github.io/98.css/>



Specialized CSS Libraries

- XP.css - <https://botoxparty.github.io/XP.css/>

The screenshot displays the XP.css website. On the left, a sidebar lists components: Button, Checkbox, OptionButton, Tabs, GroupBox, TextBox, Slider, Dropdown, Window (with sub-items Title Bar, Window contents, Status Bar), TreeView, and ProgressBar. Below this is a link to 'Issues, Contributing, etc.'. The main content area features the title 'XP.css' with a tagline 'A design system for building faithful recreations of old UIs.' and version information 'npm v0.2.6' and 'minzipped size 39.1 kB'. An 'Intro' section explains that XP.css is an extension of 98.css and provides links to GitHub. A demo window titled 'My First Program' shows a 'Hello, world!' message with 'OK' and 'Cancel' buttons. The page also includes text about semantic HTML, overriding styles, and compatibility with JavaScript frameworks. A 'Windows XP Theme' section at the bottom shows a code snippet for linking the stylesheet.

Intro

Components

- Button
- Checkbox
- OptionButton
- Tabs
- GroupBox
- TextBox
- Slider
- Dropdown
- Window
 - Title Bar
 - Window contents
 - Status Bar
- TreeView
- ProgressBar

Issues, Contributing, etc.

XP.css

A design system for building faithful recreations of old UIs.

npm v0.2.6 minzipped size 39.1 kB

Intro

XP.css is an extension of 98.css. A CSS library for building interfaces that look like old UIs.
See [XP.css on GitHub](#)
See [98.css on GitHub](#)



This library relies on the usage of **semantic HTML**. To make a button, you'll need to use a `<button>`. Input elements require labels. Icon buttons rely on `aria-label`. This page will guide you through that process, but accessibility is a primary goal of this project.

You can override many of the styles of your elements while maintaining the appearance provided by this library. Need more padding on your buttons? Go for it. Need to add some color to your input labels? Be our guest.

This library does not contain any JavaScript, it merely styles your HTML with some CSS. This means 98.css is compatible with your frontend framework of choice.

Here is an example of [XP.css used with React](#), and [an example with vanilla JavaScript](#). The fastest way to use XP.css is to import it from unpkg.

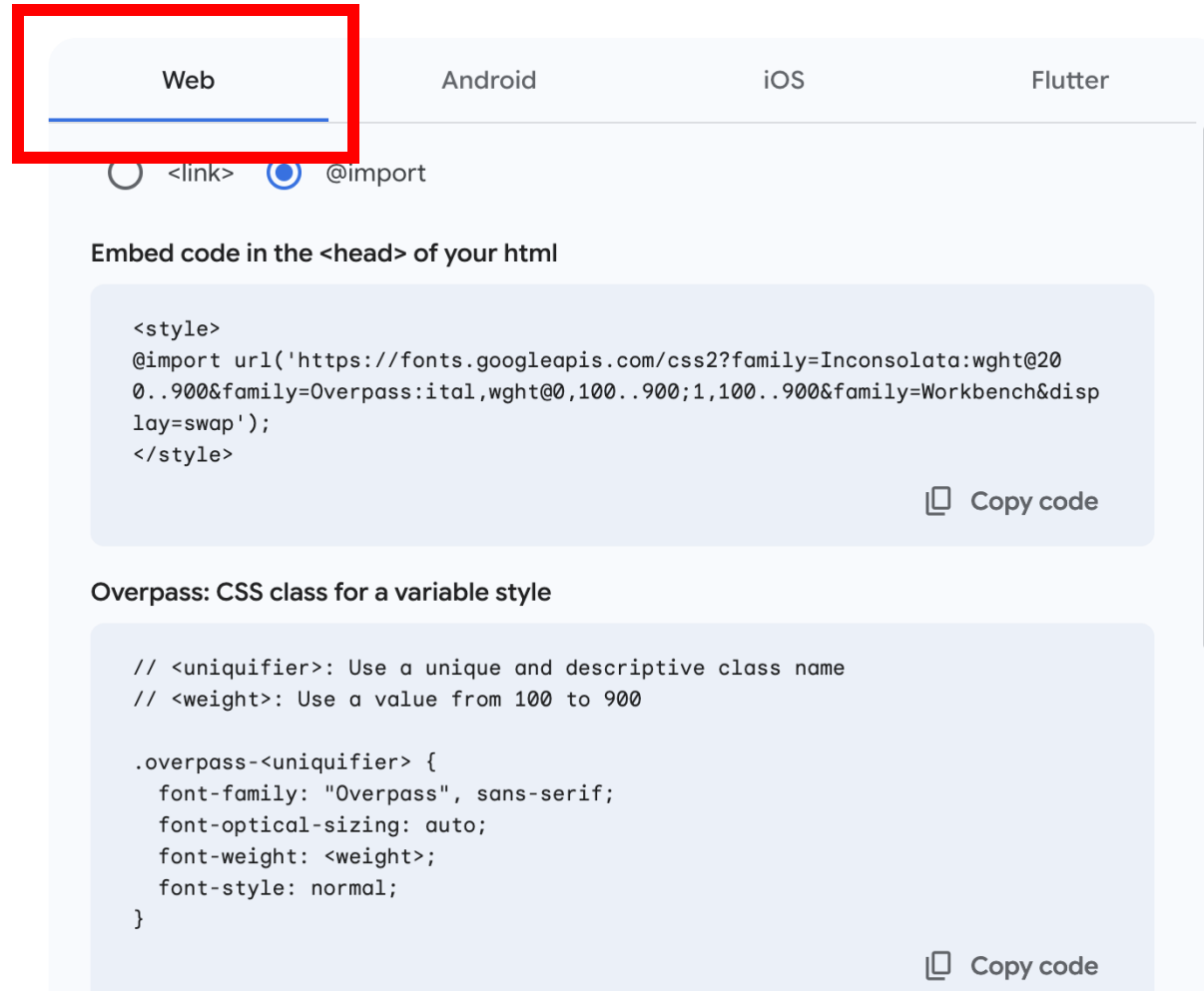
Windows XP Theme

```
<link
  rel="stylesheet"
  href="https://unpkg.com/xp.css"
>
```

Using External Webfonts Fonts

- Collections
 - Google Fonts (free)
 - Adobe Fonts (paid)
 - Fontshare (free)
- Independent Font Foundries

Using External Webfonts Fonts



The screenshot shows the Google Fonts 'Web' tab, which is highlighted with a red rectangle. The tab is part of a navigation bar with 'Android', 'iOS', and 'Flutter' options. Below the navigation bar, there are radio buttons for '<link>' and '@import', with '@import' selected. The main content area is titled 'Embed code in the <head> of your html' and contains a code block with the following CSS code:

```
<style>
@import url('https://fonts.googleapis.com/css2?family=Inconsolata:wght@200..900&family=Overpass:ital,wght@0,100..900;1,100..900&family=Workbench&display=swap');
</style>
```

To the right of the code block is a 'Copy code' button. Below this, there is a section titled 'Overpass: CSS class for a variable style' with a code block containing the following CSS code:

```
// <uniquifier>: Use a unique and descriptive class name
// <weight>: Use a value from 100 to 900

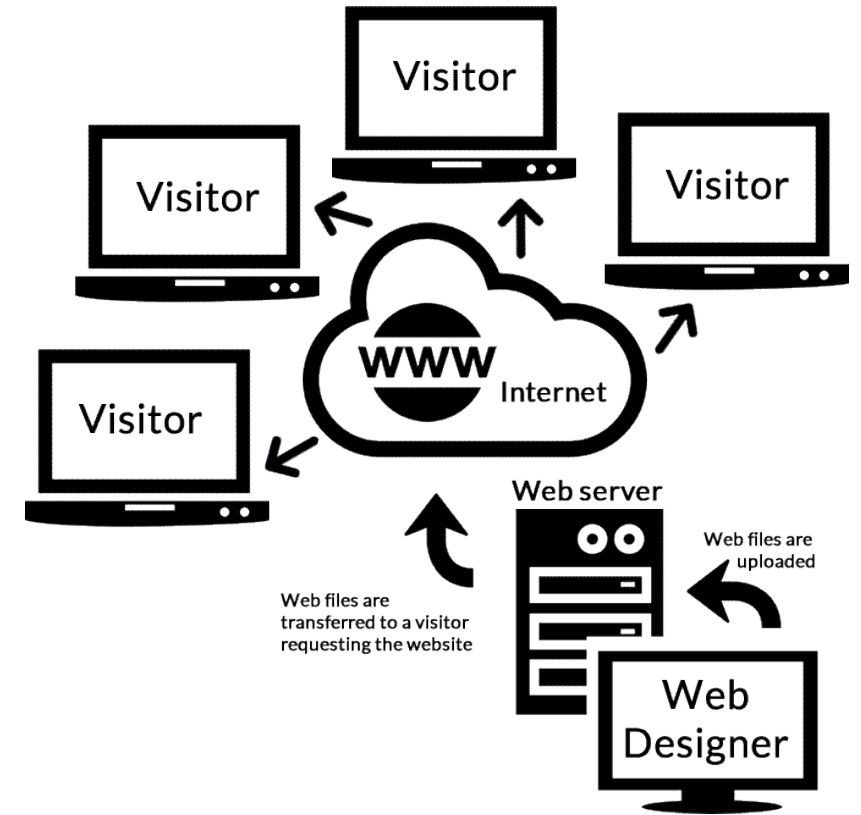
.overpass-<uniquifier> {
  font-family: "Overpass", sans-serif;
  font-optical-sizing: auto;
  font-weight: <weight>;
  font-style: normal;
}
```

To the right of this code block is another 'Copy code' button.

JavaScript & Document Object Model

How does a website work?

- What happens when I visit a website?
- What is 'client'?
- What is 'server'?



Browsers' Role

- Browsers are responsible to render our code as websites to users
- Different browsers use different browser 'engines'
- Engines are an implementation of standard conventions



View Transition example

[Article 1](#) [Article 2](#)

Article 1 content



photo credit: Michael Kirsh

Lorem ipsum dolor sit amet, consectetur adipiscing elit. facilisis vel mauris in, luctus semper turpis. Aliquam lob feugiat. Nulla aliquet ante laoreet enim maximus mollis

View Tr



Limit

Provides a m
between diff
contents in a
transitions, s
is being plan

Current aligned

Usage relative

Date relative

Filtered

Chrome	Edge *	Safari	Firefox	Opera	IE	Chrome for Android	Safari on iOS *	Samsung Internet	Opera Mini *	Opera Mobile *	Browser for Android	Android Browser *	Firefox for Android	QQ Browser	Baidu Browser	Ka Bro
4-110	12-110			10-96				4-22								
111-128	111-127	3.1-17.6	2-129	97-110	6-10		3.2-17.6	23-24		12-12.1		2.1-4.4.4				2
129	128	18.0	130	111	11	128	18.0	25	all	80	15.5	128	127	14.9	13.52	3
130-132		18.1-TP	131-133				18.1									

Open

Bug 1909173 (dt-view-transition) Opened 2 months ago Updated 2 days ago

[META] Add support for CSS View Transitions level 1 (SPA)

Categories

Product: DevTools ▾

Component: Inspector ▾

Type: task

Priority: Not set Severity: --

Tracking

Status: NEW

► People (Reporter: nchevobbe, Unassigned)

► References (Blocks 1 open bug)

► Details (Keywords: meta)

Bottom ↓

Tags ▾

Timeline ▾



Nicolas Chevobbe [:nchevobbe]

Reporter

Description • 2 months ago

CSS view transition for single page application is going to be implemented in the coming months. Let's have this bug to cover needed in DevTools to avoid issues / developer confusion

Browser Engines

- WebKit by Apple
 - Safari
- Gecko by Mozilla Foundation
 - Firefox
- Blink by Google
 - Google Chrome
 - Microsoft Edge
 - Arc
 - Brave
 - Opera

- Each browser engine is equipped with its own JavaScript engine
- JavaScript engines run along with the rendering engines via the Document Object Model to enable dynamic interactions on the web page

JavaScript Engines

- WebKit by Apple uses JavaScript Core
 - Safari
- Gecko by Mozilla Foundation uses Spider Monkey
 - Firefox
- Blink by Google uses V8
 - Google Chrome
 - Microsoft Edge
 - Arc
 - Brave
 - Opera

V8 engine also powers the NodeJS runtime

JavaScript Intro

A variable is a named spot in code that stores a value you can use or change.

Declaring a variable:

```
let x = 100; // number
```

```
let name = "cci"; // string
```

```
let isThursday = true // boolean
```

JavaScript Intro


let vs const

```
let x = 100;
```

```
x = 200;
```

```
const x = 100;
```

```
x = 200;
```

A red prohibition sign (a circle with a diagonal line through it) is overlaid on the right side of the code block containing 'x = 200;', indicating that this code is invalid or prohibited when using 'const'.

JavaScript Intro

Functions

```
function sayHello() {  
  alert("Hello!");  
}
```

```
sayHello();
```

Running the function (function execution)

Document Object Model

- The **Document Object Model (DOM)** represents an HTML page as a structured tree of nodes.
- JavaScript uses the DOM to access and modify page elements, making web pages interactive.

```
<html>

<head>

  <title>A Simple Page</title>

</head>

<body>

  <section>

    <p>A paragraph with a <a>link</a></p>

    <p>A second paragraph</p>

  </section>

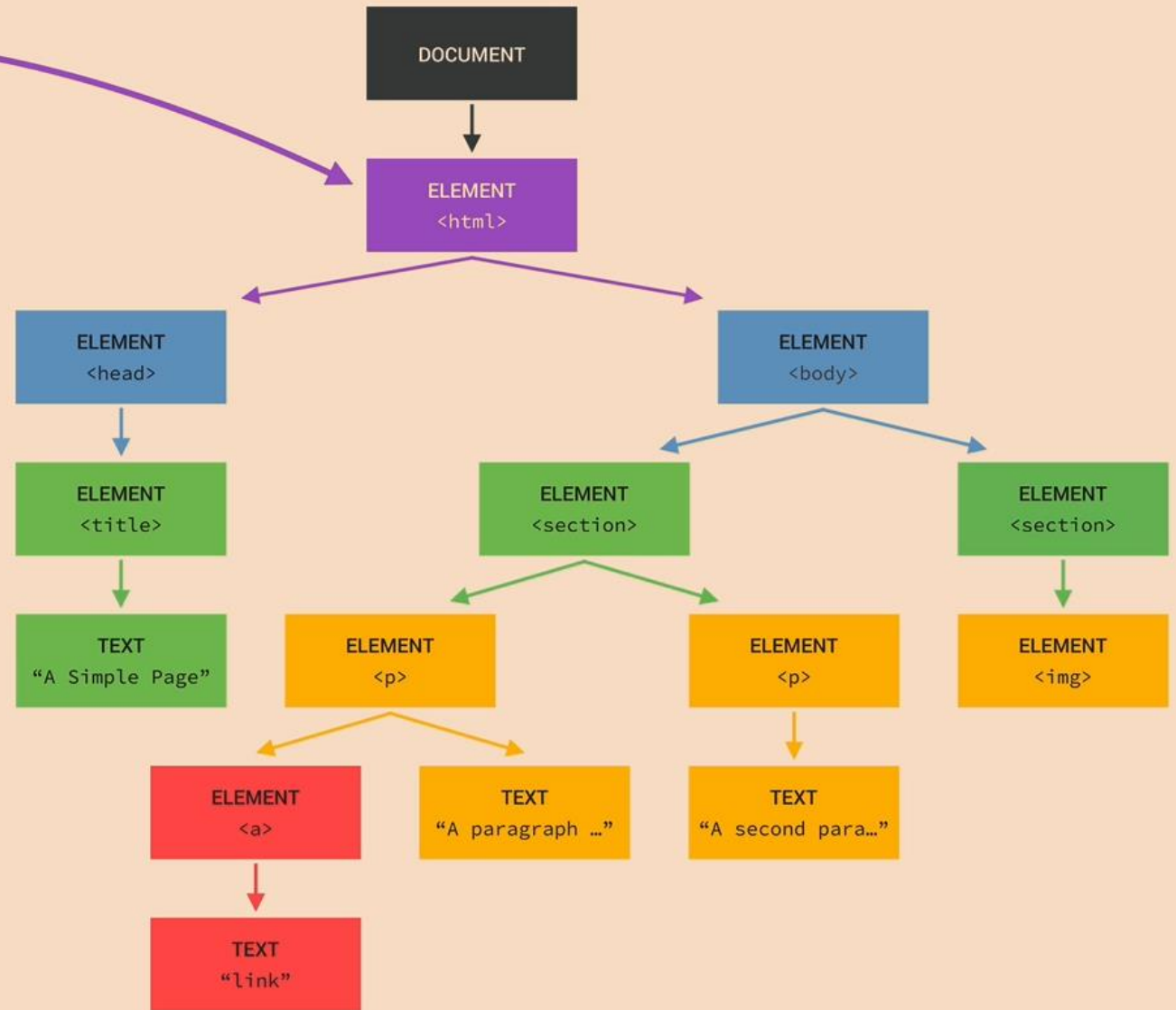
  <section>

  </section>

</body>

</html>
```



```
<html>

<head>

  <title>A Simple Page</title>

</head>

<body>

  <section>

    <p>A paragraph with a <a>link</a></p>

    <p>A second paragraph</p>

  </section>

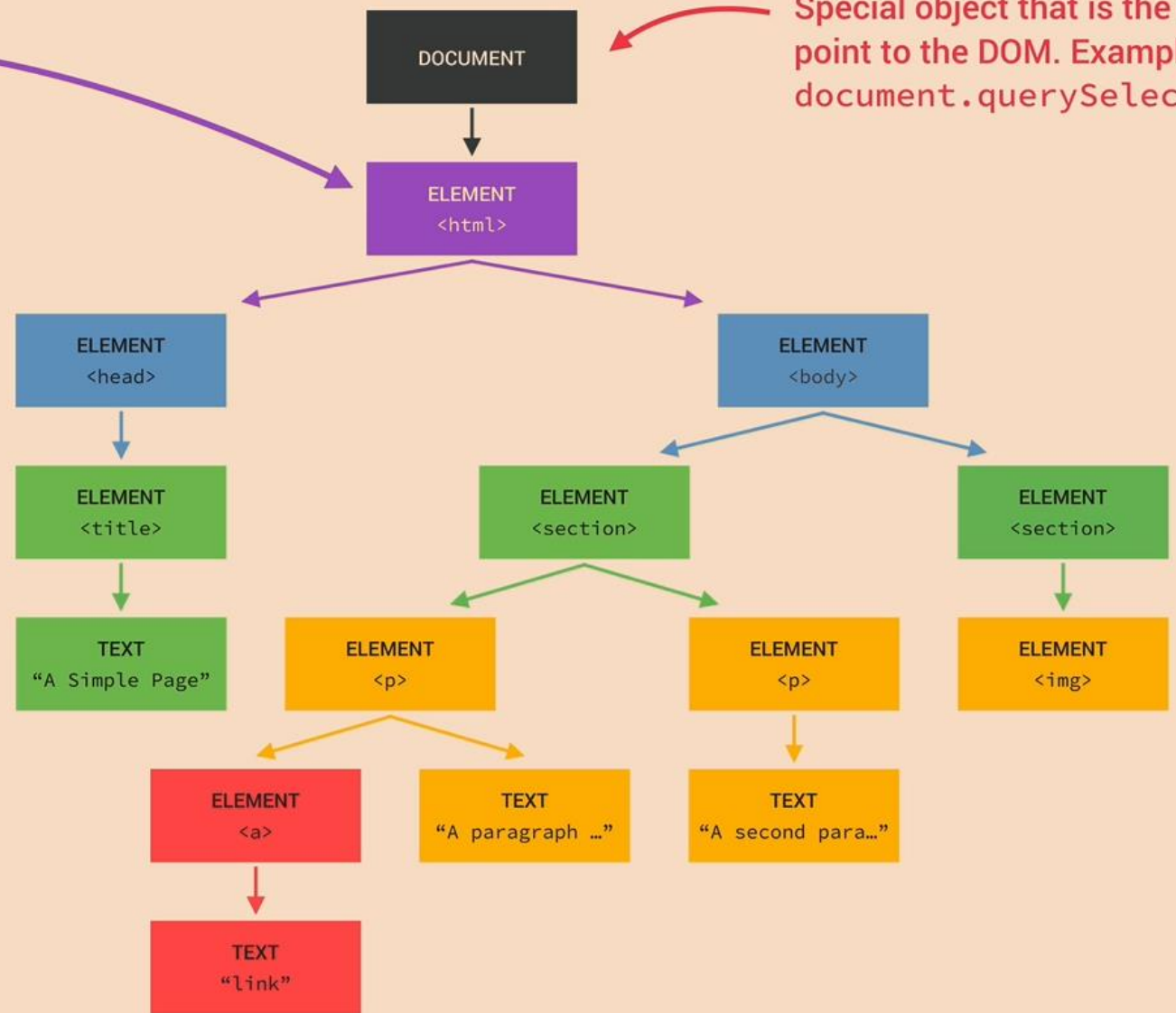
  <section>

  </section>

</body>

</html>
```



Special object that is the entry point to the DOM. Example: `document.querySelector()`

Manipulating Documents

```
document.querySelector("_____")
```

Entry point to the document

Selector for the element

Manipulating Documents

querySelector method takes CSS-like selectors to access an element

for ID

. for classes

Manipulating Documents

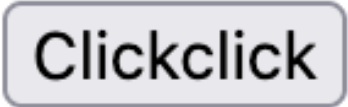
querySelector method takes CSS-like selectors to access an element

for ID

. for classes

HTML

```
<button id="submit-btn">Clickclick</button>
```



JavaScript

```
document.querySelector("#submit-btn")
```

```
const myButton = document.querySelector("#submit-btn");
```

```
myButton.textContent = "Hello World";
```

Hello World

Working with CSS Classes

```
myButton.classList.add('highlight')
```

```
myButton.classList.remove('highlight')
```

Creating New Elements

```
const myNewButton = document.createElement("button")
```

```
myNewButton.textContent = "Hello again"
```

```
document.body.append(myNewButton)
```

Creating New Elements

```
const myNewButton = document.createElement("button")
```

```
myNewButton.textContent = "Hello again"
```

```
document.body.append(myNewButton)
```



Adds the element at the end of **body**

```
<div id="about-section"></div>
```

```
const sectionContainer = document.querySelector("#about-section")
```

```
const aboutHeading = document.createElement('h2')
```

```
aboutHeading.textContent = "About Me"
```

```
sectionContainer.append(aboutHeading)
```

```
<div id="about-section"></div>
```

```
const sectionContainer = document.querySelector("#about-section")
```

```
const aboutHeading = document.createElement('h2')
```

```
aboutHeading.textContent = "About Me"
```

```
sectionContainer.append(aboutHeading)
```

Adds the h2 tag at the end of <div>

Walk-through