

Theming : Faites peau neuve en 50 minutes top chrono

RivieraDev — Theming

Add status update



View 1 + New view

Filter by keyword or by field

Discard

Save

Todo 9

This item hasn't been started



Draft

Rework section layout on mobile viewport



Draft

Add padding in footer between name and links



Draft

Switch from strong to b in headings



Draft

Find better title



Draft

Add Favicon



Draft

Add abbrs



Draft

add idiomatic tags to idiomatic texts



Draft

Fix typo in "criteriaons"



Draft

Do stuff

In Progress 1

This is actively being worked on



rivieradev-theming #1007

Implement dark theme

Done 9

This has been completed



rivieradev-theming #3

Add pictures alternating sides



rivieradev-theming #4

Add gradient to main heading



rivieradev-theming #5

Add background image to landing



rivieradev-theming #6

Add links to Jake Blucker



rivieradev-theming #7

Add license



rivieradev-theming #8

Add newsletter



rivieradev-theming #9

Add contact form



rivieradev-theming #10

Rework newsletter button to fuse with input



rivieradev-theming #11

Add calls to action

Implement dark theme #1007

Edit

Open Mintoo200/rivieradev-theming Public

Maria opened 2 days ago

No description provided.

Mintoo200 added this to RivieraDev — Theming 2 days ago

Mintoo200 converted this from a draft issue 2 days ago

Add a comment

Write Preview

Use Markdown to format your comment

Paste, drop, or click to add files

Close issue Comment

Assignees
No one - Assign yourself

Labels
No labels

Projects

RivieraDev — Theming

Status: Todo

Milestone
No milestone

Development
Create a branch for this issue or link a pull request.

Notifications
Unsubscribe

You're receiving notifications because you're subscribed to this thread.

- Participants
- Transfer issue
 - Lock conversation
 - Pin issue

Merci

Merci



From Maria:

Hey, est-ce que tu pourrais ajouter un bouton, ou un dropdown, pour...





Maria:

Hey, est-ce que tu pourrais ajouter un bouton, ou un dropdown, pour sélectionner le thème directement sur le site ?

```
:root {  
  background-color: var(--background-color);  
  color: var(--text-color);  
}
```

```
:root  
  <body data-theme="dark">  
    ...  
  </body>  
/:root
```

The diagram illustrates the resolution of CSS variables. An arrow points from the `var(--background-color)` property in the `:root` selector to the `<body data-theme="dark">` element. Another arrow points from the `data-theme="dark"` attribute to the `[data-theme]` selector, indicating that the value of the `data-theme` attribute is used to select the appropriate CSS rules.

```
[data-theme] {  
  --background-color: black;  
  --text-color: white;  
}
```

Merci



Cyclic Dependency Space Toggles



Over the past few years, I wanted to be able to select a value from a list in CSS by toggling a single custom property. We have the “space toggle” for booleans, and hopefully, one day, we’ll get style queries, but what about today? In this article, I present you with the technique I recently discovered.

Table of Contents

1. The Technique
 - What is Going On?
 - Multi-Select
 - Was This Always Possible?
2. Space Toggles Logic
 - Buffer and an Impossible Inverter
 - AND, NAND, OR, NOR, XOR, XNOR
 - Fallback Values
3. Do We Still Need Style Queries?
4. Limitations
 - Inheritance
 - Animations
 - Naming
5. Use Cases

The Technique

Let me start¹ by showing you the code, and then I will explain the technique (which I would refer to as “Cyclic Toggles”). Look:

```
.info {
  --level: var(--level--default);

  --level--default: var(--level,);
  --level--success: var(--level,);
  --level--warning: var(--level,);
  --level--error:   var(--level,);

  background:
    var(--level--default, lavender)
    var(--level--success, palegreen)
```

All this builds up on the famous “space toggle” technique, alongside other inspirations — I’ll talk about them [later in the article](#).

The experiments that led to this technique came after [CSSDay](#), where there was a lot of excitement about the potential of style queries. I’ll write up about this in more detail in one of my future articles.

Merci

Merci



From Maria:

Concernant les blocs de code à la fin, est-ce que tu peux forcer...









Maria:

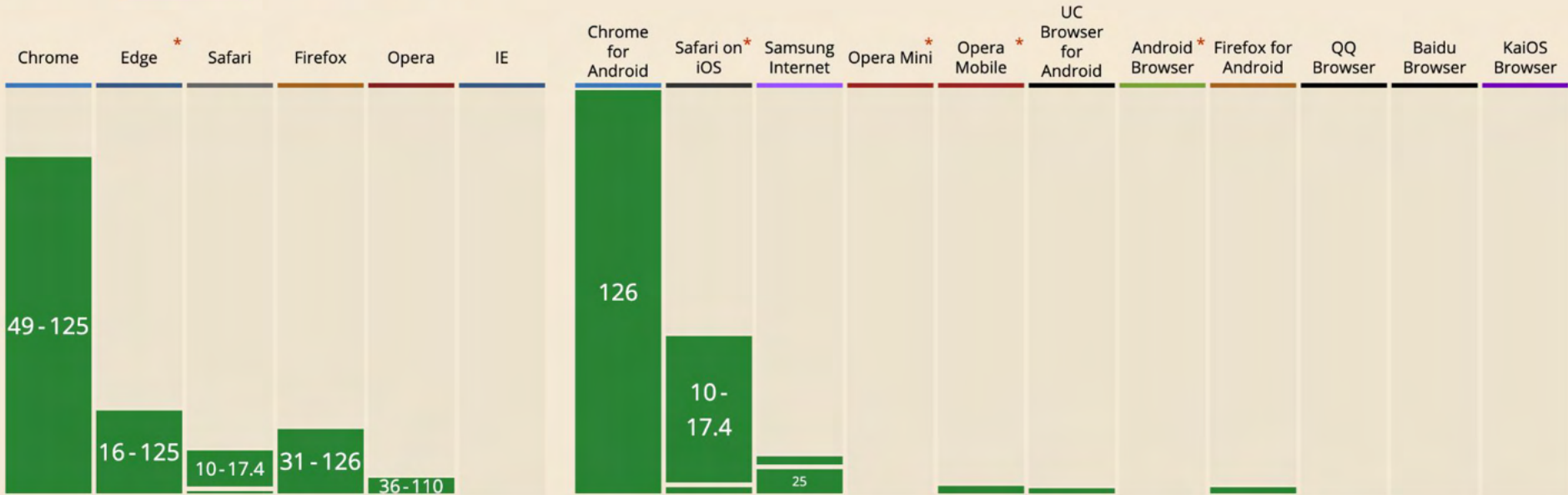
Concernant les blocs de code à la fin, est-ce que tu peux forcer le thème sombre sur l'un et le thème clair sur l'autre ?

CSS Variables (Custom Properties) - CR

 **Baseline** Widely available across major browsers  

Permits the declaration and usage of cascading variables in stylesheets.

Current aligned **Usage relative** Date relative Filtered **All** 



Notes Test on a real browser Known issues (4) Resources (6) Feedback

Since April 2017, this feature works across the latest devices and major browser versions ([learn more about Baseline](#))

Usage % of all tracked  ?
• France 99.74% + 0.03% = 99.77%
Global 98.95% + 0.03% = 98.99%

Feature summary

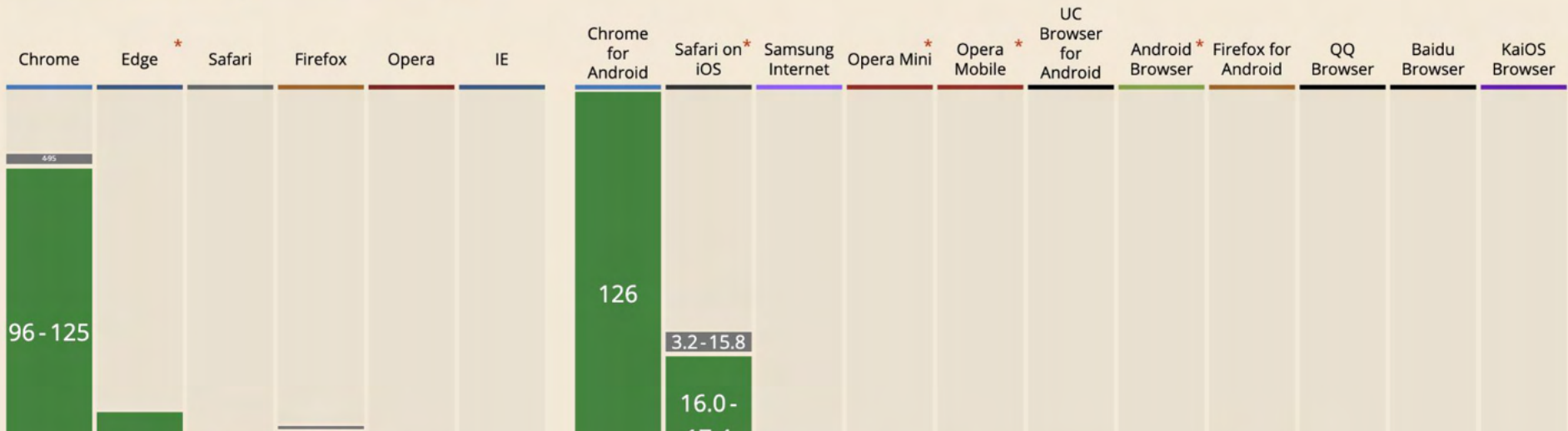
Summary table of the following features:

- 1. dataset & data-* attributes
- 2. CSS 2.1 selectors
- 3. CSS at-rule: `@media`: `prefers-contrast` media feature
- 4. CSS at-rule: `@media`: `forced-colors` media feature
- 5. types: `<color>`: System colors
- 6. prefers-color-scheme media query
- 7. CSS property: color-scheme

Summary type: Combined Intersection Union Average score

Intersection: Each browser version is only considered supported if **all features** are supported.

Current aligned Usage relative Date relative Filtered All



Usage % of all tracked

- France 94.83%
- Global 92.91%

Feature summary

Usage % of all tracked ⬇ ?

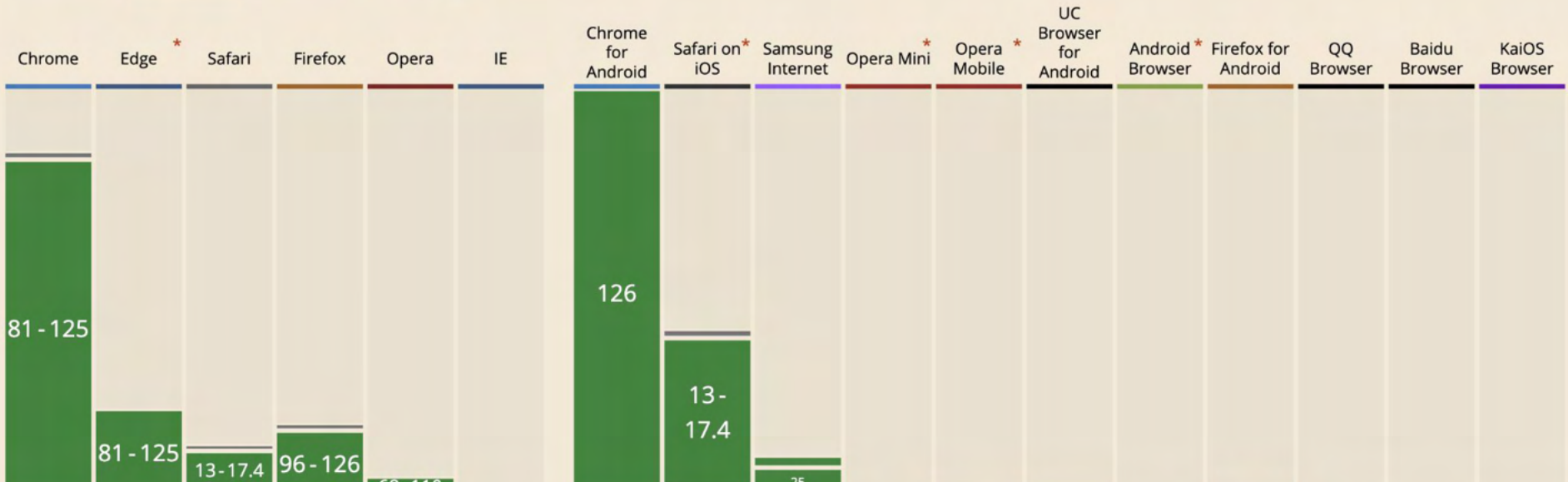
Summary table of the following features:

- 1. dataset & data-* attributes
- 2. CSS 2.1 selectors
- 3. types: `<color>`: System colors
- 4. prefers-color-scheme media query
- 5. CSS property: color-scheme

Summary type: Combined **Intersection** Union Average score

Intersection: Each browser version is only considered supported if **all features** are supported.

Current aligned **Usage relative** Date relative Filtered **All** ⚙



Aller plus loin

Parlons Cascade

Click Me!

Click Me!

Click Me!

Click Me!

ur app

used to define the visual aspect of a page or website.
to the practice of letting the user choose parts of the
pre-conceived sets of assets. You'll often hear about
es well beyond that with web applications like
d themes with features such as custom gradients,

[Learn more on Wikipedia →](#)

ur app

used to define the visual aspect of a page or website.
to the practice of letting the user choose parts of the
pre-conceived sets of assets. You'll often hear about
es well beyond that with web applications like
d themes with features such as custom gradients,

[Learn more on Wikipedia →](#)

```
[data-theme="dark"] button {  
  border-radius: 30px;  
}
```

```
@media (prefers-color-scheme: dark) {  
  button {  
    border-radius: 30px;  
  }  
}
```

```
@media (prefers-color-scheme: dark) {  
  button {  
    border-radius: 30px;  
  }  
}
```

```
/* ... */
```

```
button {  
  border-radius: 10px;  
}
```

```
@media (prefers-color-scheme: dark) {  
  :root button {  
    border-radius: 30px;  
  }  
}  
  
/* ... */  
  
button {  
  border-radius: 10px;  
}
```


Newsletter

Expect about **0 mail per day**
No useful information whatsoever
100% newsletter content guaranteed

E-mail

Subscribe →

```
button {  
    border-radius: 10px;  
}  
input + button {  
    border-top-left-radius: 0px;  
    border-bottom-left-radius: 0px;  
}
```



```
@media (prefers-color-scheme: dark) {  
  :root button {  
    border-radius: 30px;  
  }  
}  
  
/* ... */  
  
input + button {  
  border-top-left-radius: 0px;  
  border-bottom-left-radius: 0px;  
}
```

```
@media (prefers-color-scheme: dark) {  
  button {  
    border-radius: 30px;  
  }  
}
```

```
/* ... */
```

```
button {  
  border-radius: 10px;  
}
```

```
button {  
  border-radius: 10px;  
}  
input + button {  
  border-top-left-radius: 0px;  
  border-bottom-left-radius: 0px;  
}  
  
/* ... */  
  
@media (prefers-color-scheme: dark) {  
  button {  
    border-radius: 30px;  
  }  
}
```

```
button {  
  border-radius: 10px;  
}  
input + button {  
  border-top-left-radius: 0px;  
  border-bottom-left-radius: 0px;  
}  
  
/* ... */  
  
[data-theme="dark"] button {  
  border-radius: 30px;  
}
```

```
button {  
  border-radius: 10px;  
}  
input + button {  
  border-top-left-radius: 0px;  
  border-bottom-left-radius: 0px;  
}  
  
/* ... */  
  
:where([data-theme="dark"]) button {  
  border-radius: 30px;  
}
```

```
<code data-theme="light">...</code>
```

```
<body data-theme="light">
  ...
  <code data-theme="dark">
    <button>...</button>
  </code>
</body>
```

```
[data-theme="dark"] button {
  border-radius: 30px;
}
[data-theme="light"] button {
  border-radius: 10px;
}
```



```
<body data-theme="light">  
  ...  
  <code data-theme="dark">  
    <button>...</button>  
  </code>  
</body>
```

```
[data-theme="dark"] button {  
  border-radius: 30px;  
}  
[data-theme="light"] button {  
  border-radius: 10px;  
}
```

```
<body data-theme="light">
  ...
  <code data-theme="dark">
    <button>...</button>
  </code>
</body>
```

```
[data-theme="dark"] button {
  border-radius: 30px;
}
[data-theme="light"] button {
  border-radius: 10px;
}
```

```
<body data-theme="light">
  ...
  <code data-theme="dark">
    <button>...</button>
  </code>
</body>
```

```
[data-theme="dark"] *,
[data-theme] [data-theme="dark"] * {
  border-radius: 30px;
}

[data-theme="light"] *,
[data-theme] [data-theme="light"] * {
  border-radius: 10px;
}
```

```
<body data-theme="dark">
  ...
  <pre data-theme="light">
    <code data-theme="dark">
      <button>...</button>
    </code>
  </pre>
</body>
```

```
[data-theme="dark"] *,
[data-theme] [data-theme="dark"] * {
  border-radius: 30px;
}

[data-theme="light"] *,
[data-theme] [data-theme="light"] * {
  border-radius: 10px;
}
```




```
<body data-theme="light">
  ...
  <code data-theme="dark">
    <button>...</button>
  </code>
</body>
```

```
[data-theme="dark"] button {
  border-radius: 30px;
}
[data-theme="light"] button {
  border-radius: 10px;
}
```


```
<body data-theme="light">
  ...
  <code data-theme="dark">
    <button>...</button>
  </code>
</body>
```

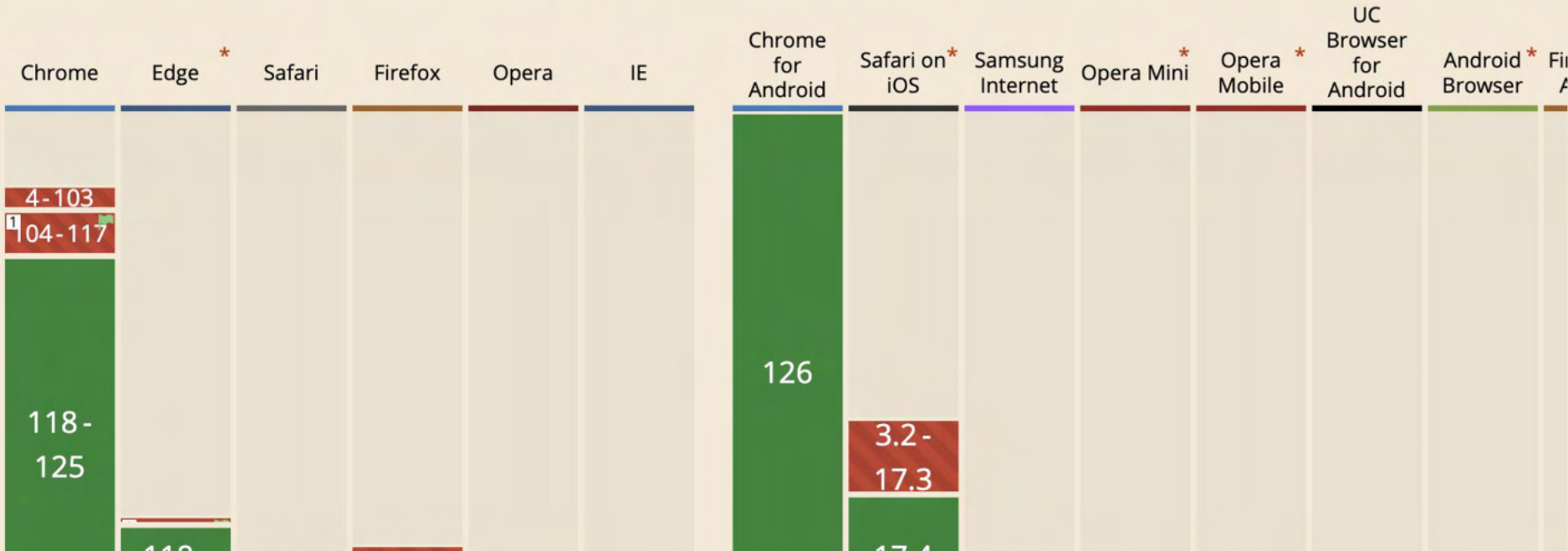
```
@scope ([data-theme="dark"]) {
  button {
    border-radius: 30px;
  }
}
@scope ([data-theme="light"]) {
  button {
    border-radius: 10px;
  }
}
```


Scoped Styles: the @scope rule - WD

 Limited availability across major browsers  

Allows CSS rules to be scoped to part of the document, with upper and lower limits described by selectors.

Current aligned Usage relative Date relative Filtered All 



Usage % of all tracked  ?

- France 81.47%
- Global 83.48%

```
<body data-theme="light">
  ...
  <code data-theme="dark">
    <button>...</button>
  </code>
</body>
```

```
[data-theme="dark"] button {
  border-radius: 30px;
}
[data-theme="light"] button {
  border-radius: 10px;
}
```



```
<body data-theme="light">
  ...
  <code data-theme="dark">
    <button>...</button>
  </code>
</body>
```

```
[data-theme="light"] {
  --border-radius: 10px;
}

[data-theme="dark"] {
  --border-radius: 30px;
}

button {
  border-radius: var(--border-radius);
}
```

```
:root, [data-theme] {  
  --style: var(--round);  
  --round: var(--style,);  
  --square: var(--style,);  
}  
button {  
  border-radius:  
    var(--round, 30px)  
    var(--square, 0px);  
}
```

```
:root, [data-theme] {  
  --style: var(--round);  
  --round: var(--style,);  
  --square: var(--style,);  
}  
button {  
  border-radius: var(--round, 30px);  
}  
button {  
  border-radius: var(--square, 0px);  
}
```

```

button, .call-to-action {
  --knob-width: 1ch;
  --knob-border-thickness: 2px;
  --knob-inset: 1ch;
  position: var(--button-magic, relative) var(--button-knobs, relative);
  border-radius: var(--button-magic, 1em) var(--button-knobs, 0);
  z-index: var(--button-magic, 0);
  &::before, &::after {
    position: var(--button-magic, absolute) var(--button-knobs, absolute);
    content: var(--button-magic, "") var(--button-knobs, "");
    aspect-ratio: var(--button-magic, 1);
    border-top: var(--button-magic, 2px solid var(--primary-color)) var(--button-knobs, var(--knob-border-thickness) solid var(--primary-color));
    border-bottom: var(--button-magic, 2px solid var(--primary-color)) var(--button-knobs, var(--knob-border-thickness) solid var(--primary-color));
    border-left: var(--button-knobs, var(--knob-border-thickness) solid var(--primary-color));
    border-right: var(--button-knobs, var(--knob-border-thickness) solid var(--primary-color));
    border-radius: var(--button-magic, 50%);
    transition-duration: var(--button-magic, 300ms) var(--button-knobs, 200ms);
    transition-timing-function: var(--button-magic, linear) var(--button-knobs, linear);
    opacity: var(--button-magic, 0);
    animation: var(--button-magic, rotate linear infinite 2s);
    z-index: var(--button-magic, -1) var(--button-knobs, 1);
    background-color: var(--button-knobs, var(--background-color));
    transform-origin: top left;
  }
  &:hover::before, &:hover::after {
    opacity: var(--button-magic, 1);
  }
  &:hover::before {
    left: var(--button-magic, 50%) var(--button-knobs, calc(100% - var(--knob-width) - var(--knob-inset) - var(--knob-border-thickness) * 2));
  }
  &:hover::after {
    right: var(--button-magic, 50%) var(--button-knobs, calc(100% - var(--knob-width) - var(--knob-inset) - var(--knob-border-thickness) * 2));
  }
  &::before {
    top: var(--button-magic, 50%) var(--button-knobs, 0);
    right: var(--button-magic, 50%);
    bottom: var(--button-magic, 50%);
    left: var(--button-magic, 50%) var(--button-knobs, var(--knob-inset));
    transform: var(--button-magic, translate(-50%, -50%)) var(--button-knobs, translateY(-50%));
    transition-property: var(--button-magic, opacity) var(--button-knobs, left);
    height: var(--button-magic, 160%) var(--button-knobs, .4em);
    width: var(--button-knobs, var(--knob-width));
  }
  &::after {
    top: var(--button-magic, 50%);
    right: var(--button-magic, 50%) var(--button-knobs, var(--knob-inset));
    bottom: var(--button-magic, 50%) var(--button-knobs, 0);
    left: var(--button-magic, 50%);
    transform: var(--button-magic, translate(-50%, -50%)) var(--button-knobs, translateY(50%));
    transition-property: var(--button-magic, opacity) var(--button-knobs, right);
    height: var(--button-magic, 200%) var(--button-knobs, .4em);
    width: var(--button-knobs, var(--knob-width));
    animation-direction: var(--button-magic, reverse);
  }
}

```

```
[data-theme="dark"] *,
[data-theme] [data-theme="dark"] * {
    border-radius: 30px;
}
```

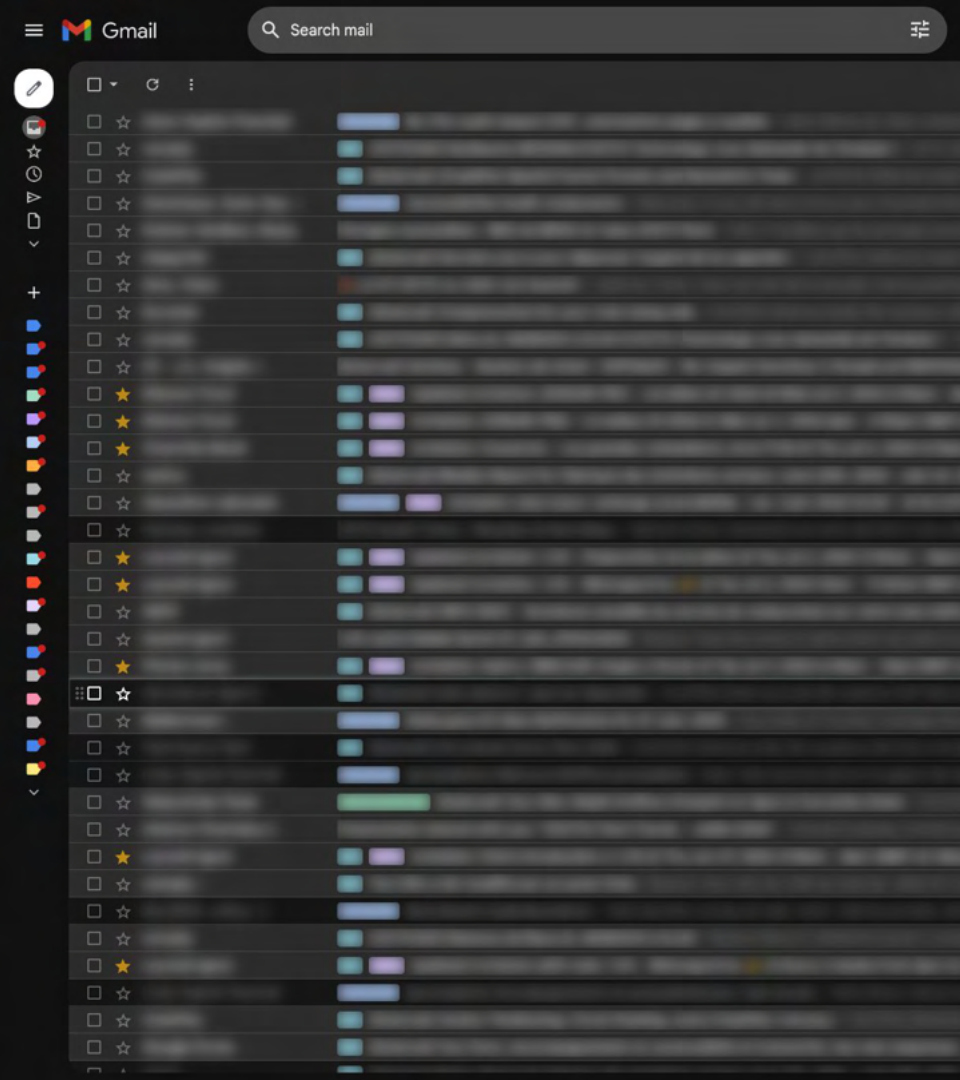
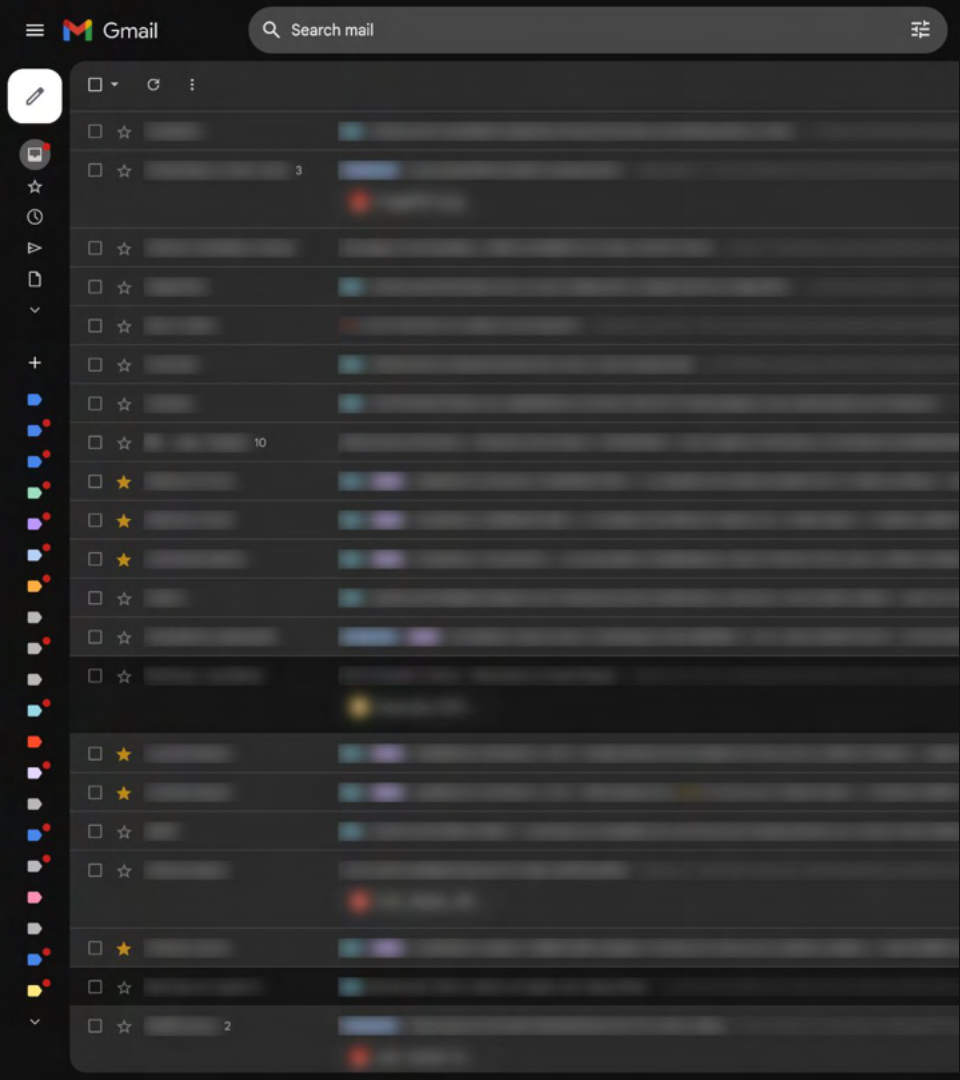
```
[data-theme="light"] *,
[data-theme] [data-theme="light"] * {
    border-radius: 10px;
}
```

```
@scope ([data-theme="dark"]) {
    button {
        border-radius: 30px;
    }
}

@scope ([data-theme="light"]) {
    button {
        border-radius: 10px;
    }
}
```

Theming : Faites peau neuve en 50 minutes top chrono

~~Theming~~ : Faites peau neuve en 50 minutes top chrono
Spacing : Faites peau neuve en 50 minutes top chrono



~~Theming~~ : Faites peau neuve en 50 minutes top chrono

~~Spacing~~ : Faites peau neuve en 50 minutes top chrono

Layout : Faites peau neuve en 50 minutes top chrono

~~Theming~~ : Faites peau neuve en 50 minutes top chrono

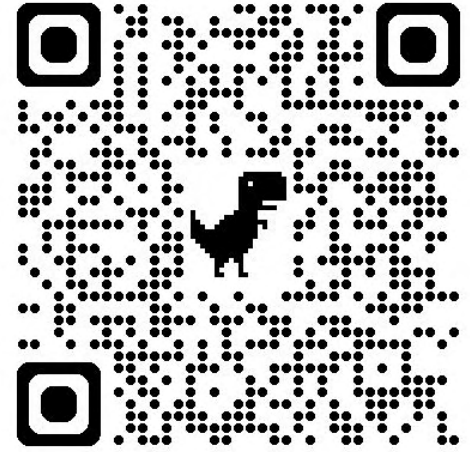
~~Spacing~~ : Faites peau neuve en 50 minutes top chrono

~~Layout~~ : Faites peau neuve en 50 minutes top chrono

~~Animations~~ : Faites peau neuve en 50 minutes top chrono

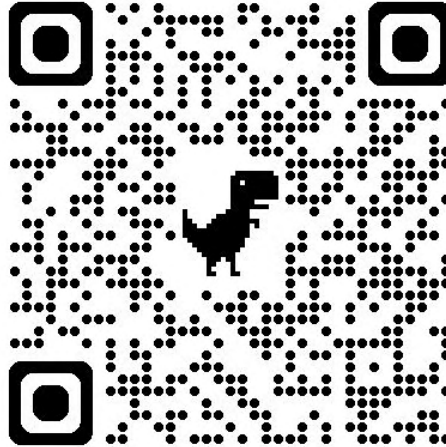
Merci

Code source
par là



github.com/Mintoo200/rivieradev-theming

openfeedback.io/rivieradev24/0/216



Feedback apprécié
par ici

et par là

