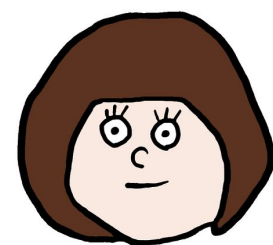
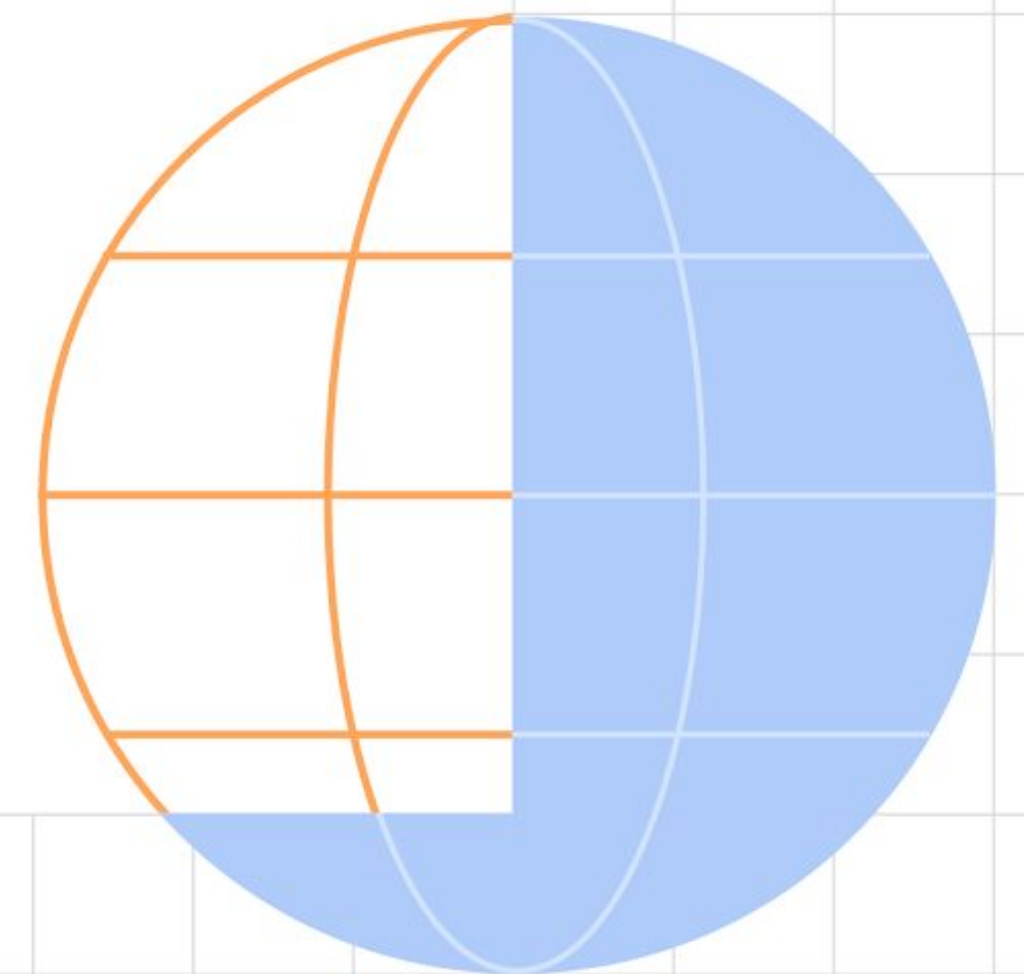


# TechFeed Experts Night #14

今までああ書いてたアレ、これからはこう書けそう



よしこ  
Google Developer Experts for Web  
@yoshiko\_pg



# About Speaker

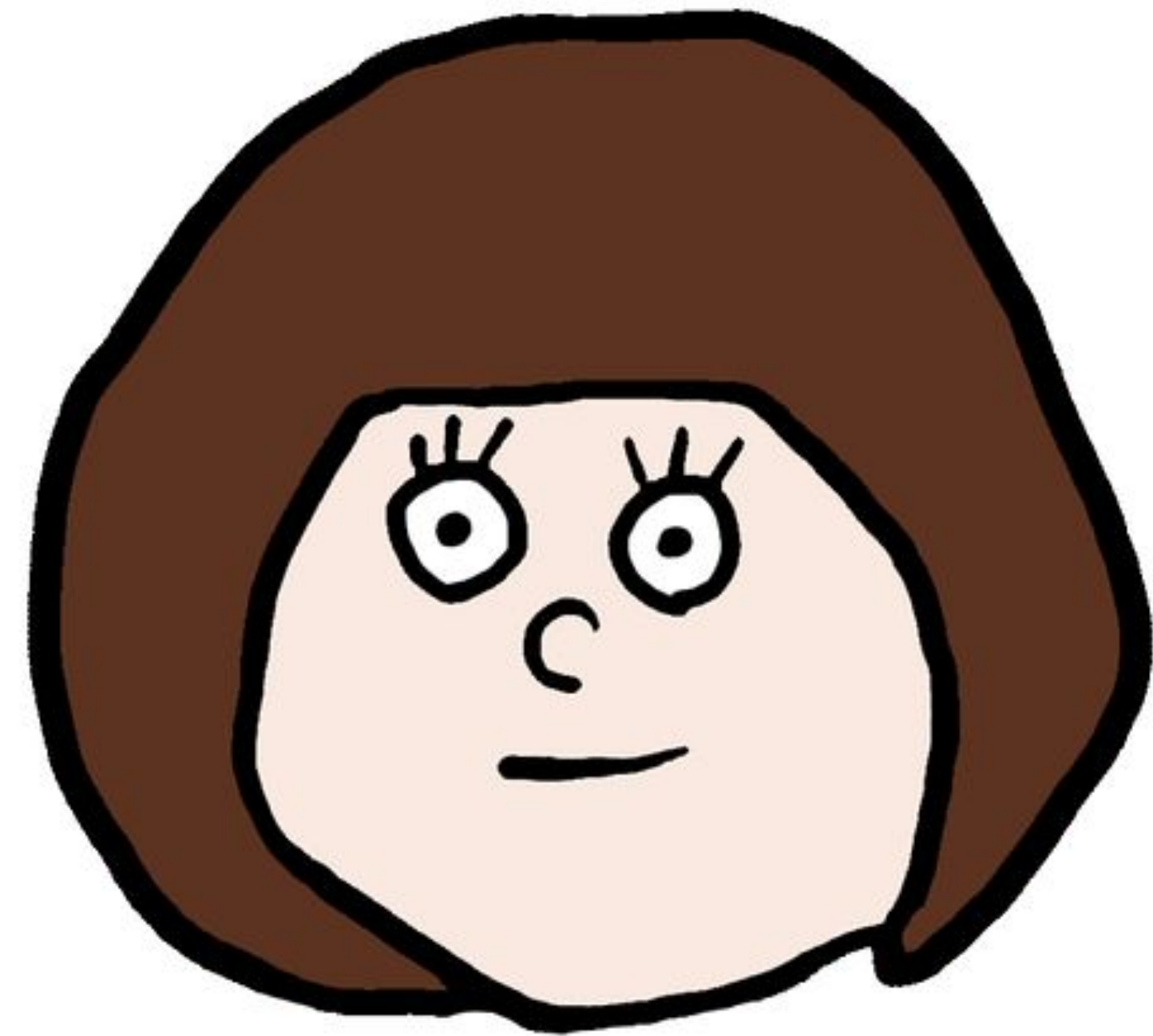
よしこ - @yoshiko\_pg

株式会社ナレッジワーク  
フロントエンドエンジニア

SPAでGUIツールを作るのが好きです。

Zennでフロントエンド設計の記事群を公開しています。

<https://zenn.dev/yoshiko>



# About Session

本日はDevFest 2022のWeb Sessionの再演という形で  
CSSだけ抜き出したものを8分間のLT形式でお話しします！

## このセッションについて

Webフロントエンド開発で頻出するパターンの中で、  
そのパターンを実現するための専用の仕様が追加されたよ！というものを紹介します。

今回は最新モダンブラウザで実装済みのものから抜粋してご紹介します。

実際に使えるかどうかは各アプリケーションのサポート環境次第ですが、  
IE11も今年の6月にサポートが終了し、一般的なサポート環境の水準にも変化が予想されるので  
状況がマッチしたときに引き出せる、知識の引き出しのひとつになれば幸いです。

# :focus-visible

## CSS Selectors Level 4

フォーカスの場所を伝えたほうがよいとブラウザが判断した場合に適用される。たとえば、クリックしたときはoutlineが出ないがタブキーで移動したときはoutlineが出る、という振る舞いを期待できる

```
/* before */
```

```
/* クリックしたときにも線が出てしまい、うるさい。  
   が、消すとタブキーでの移動が困難になってしまう */
```

```
:focus {  
  outline: 2px solid blue;  
}
```

```
/* after */
```

```
/* クリックしたときには線が表示されず、  
   タブキーでフォーカス移動しているときだけ表示される */
```

```
:focus-visible {  
  outline: 2px solid blue;  
}
```

# overscroll-behavior

## [CSS Overscroll Behavior Module Level 1](#)

スクロール可能な領域同士が重なっている場合、スクロール操作中の領域のスクロール終点まで辿り着いたときに後ろのスクロール領域がスクロールされはじめてしまう問題をCSSで解消できる

```
/* before */  
/* スクロールを止めたい領域のoverflowに一時的に  
   hiddenを指定することでスクロールできなくしている。  
   スクロール位置がリセットされてしまう問題がある */  
.overflow-scroll-area {  
  overflow-y: auto;  
}  
  
body {  
  overflow: hidden;  
}  
  
/* after */  
.overflow-scroll-area {  
  overflow-y: auto;  
  overscroll-behavior-y: contain;  
}
```



# accent-color

## CSS Basic User Interface Module Level 4

一部のform control要素のアクセントカラーを変更できる  
アクセントカラーの上に重なる要素  
(チェックボックスのチェックなど)の色はブラウザがコントラストを保ってくれる

```
/* before */
```

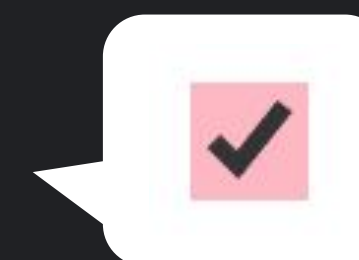
```
/* デフォルトのcheckboxは非表示にし、自分でスタイリング  
したcheckboxの見た目の要素を表示していた */
```

```
.checkbox input[type=checkbox] {  
  visibility: hidden;  
}
```

```
.checkbox span {  
  /* 好きな色のcheckboxな見た目の実装 */  
}
```

```
/* after */
```

```
input[type=checkbox] {  
  accent-color: pink;  
}
```



# translate, scale, rotate

## [CSS Transforms Module Level 2](#)

transformプロパティの関数であるこれら3つのアクションがCSSプロパティとして指定できるようになるので、個別に上書きやアニメーションができるようになる

```
/* before */
.big-box {
  transform: scale(2);
}
.big-box.downward {
  transform: scale(2) rotate(180deg);
}
```

```
/* after */
.big-box {
  scale: 2;
}
.big-box.downward {
  rotate: 180deg;
}
```

# L/S/D Viewport Units

## CSS Values and Units Module Level 4

vh/vw/vmin/vmaxに対してl/s/dの prefixをつけた単位が追加。URL バーなどで画面領域が変動しうる場合、l(arge)が最大～s(mall)が最小領域を表し、d(ynamic)は動的に現在の領域の値を表す。

Supported Version	Chrome Edge	Firefox	Safari	Android Chrome	iOS Safari
	108	101	15.4	108	15.4

```
/* before-1 */
/* 動的に変わるブラウザバーの高さを考慮してくれない */
body {
  height: 100vh;
}

/* before-2 */
/* 画面高さの何%、のような指定はできない */
body {
  height: -webkit-fill-available;
}

/* after */
body {
  height: 100dvh;
}
```



# @layer

## CSS Cascading and Inheritance Level 5

CSSのカスケードにおける新しい概念。記述順や詳細度よりも優先されるので、記述の仕方ではなく開発者の意図したレイヤー構造ベースでスタイルの重ねがけルールを定義できる。

```
/* before */
/* 定義順に気をつけてカスタム定義を後勝ちさせる、
   詳細度で負けたら!importantで無理やり勝たせるなど... */
@import url("ui-library.css");

.item {
  /* カスタマイズしたい内容 */
}

/* after */
@layer framework, custom;

@layer custom {
  .item {
    /* カスタマイズしたい内容 */
  }
}

@import url("ui-library.css") layer(framework);
```

# @media range syntax 🎉

## [CSS Media Queries Level 4](#)

比較演算子を使ってメディアクエリを定義できる。min-width、max-widthだけでは素直に記述できなかった未満・超過の指定も、不等号を使うことで実現できる。

```
/* before */
@media (min-width: 700px) and (max-width: 900px) {
  /* viewport幅が700px以上900px以下なら */
}

@media (max-width: 299px) {
  /* viewport幅が300px未満なら */
}

/* after */
@media (700px <= width <= 900px) {
  /* viewport幅が700px以上900px以下なら */
}

@media (width < 300px) {
  /* viewport幅が300px未満なら */
}
```

# @container 🎉

## [CSS Containment Module Level 3](#)

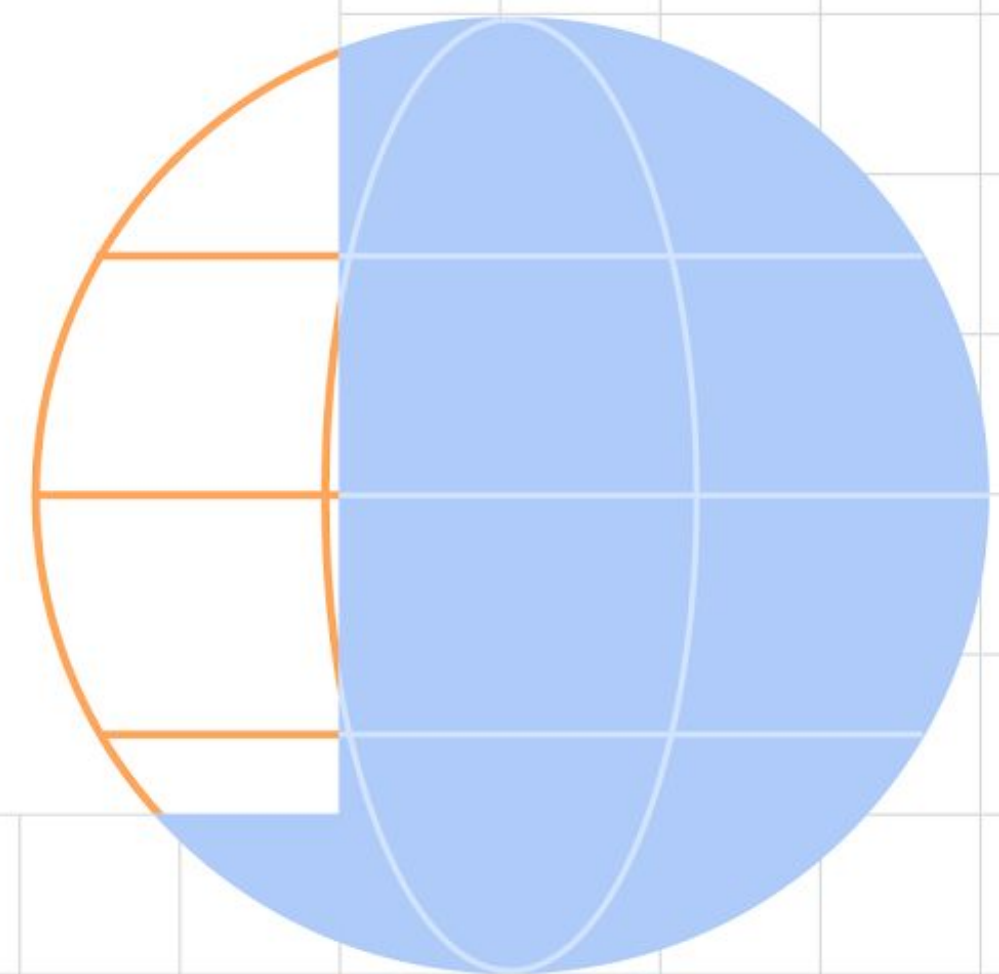
media queryのようにviewportのサイズではなく、コンテナコンテキストとして宣言されたコンテナ要素のサイズを元にクエリで条件を指定してスタイルを分岐させられる。

```
/* before */
@media (max-width: 300px) {
  .cards {
    flex-direction: column;
  }
}

/* after */
.container {
  container-type: inline-size;
}
@container (min-width: 300px) {
  .cards {
    flex-direction: column;
  }
}
```



最後に...



# **\_KNOWLEDGE WORK**

ナレッジワーク

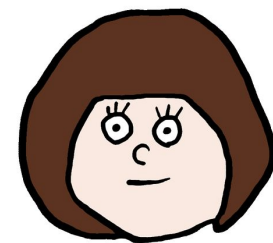
<https://kwork.studio>



Google Developers



# Thank You!



よしこ [@yoshiko\\_pg](#)  
Google Developer Experts for Web