


トランジション

など

An abstract background on the left side of the slide, composed of numerous overlapping, semi-transparent triangles in various shades of green and yellow, creating a mosaic-like effect.

CSSのおさらい

CSSのおさらい

1. タグ, #id, .class毎にデザインを指定
2. CSS3から動きもつけれるように
(transition, animation)
3. CSS設計 (BEMなど)
4. 新しい書き方 Sass/Scss

<https://developer.mozilla.org/ja/docs/Web/CSS>

<https://coinbaby8.com/css-how-to-write.html>

Sass/Scssについて

Sass/Scss

変数、インポート、ネスト(&)


などを使える

CSS設計(BEMなど)と相性がいい

<https://mae.chab.in/archives/2553>


VS Code おすすめプラグイン

Live Sass Compiler



サンプル1 タブメニュー

タブメニュー



`` はページ最上部に移動するので `@click.prevent` をつける

クラスをつける条件をつける方法

`:class="{クラス名: 条件}"`

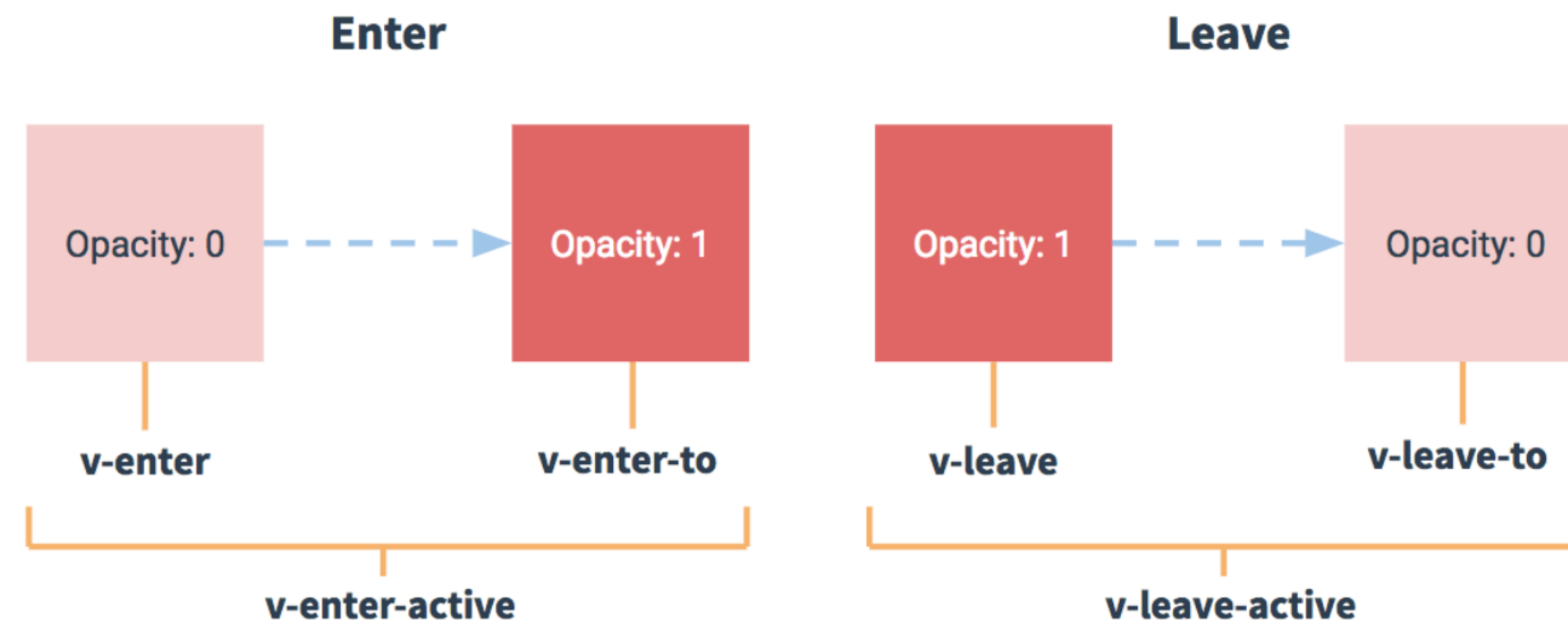
`:class="{active: active === item.id}"`



Transition

<transition> 遷移 移り変わり

CSSのtransitionとは別物
ルールに沿ってclassが自動で設定
ルールに合わせてCSSを書く事で楽につくれる



<transition> 遷移 移り変わり



```
<transition name="fade">
```

```
  <div v-show=""></div>
```

```
</transition>
```

name属性でクラス名を指定

transitionタグの中は単一要素

v-ifやv-showで表示/非表示を切り替え

<transition-group>




オブジェクトや配列をv-forで表示時に使う

```
<transition-group name="fade" tag="p">
```

```
<v-for="" :key="">
```

```
</transition-group>
```

:keyは必須。v-move クラスも使える



サンプル2

モーダルウィンドウ

モーダルウィンドウ

表示非表示の切り替え

< @click=“dataの中の値 = 設定したい値”>

!をつけると否定になり、

trueならfalse, falseならtrueになる

<button @click=“isShow = !isShow”>




Array.splice

Array.spliceは万能の配列関数

Vue.jsのリアクティブシステムを
維持できる配列の関数

- `push()`
- `pop()`
- `shift()`
- `unshift()`
- `splice()`
- `sort()`
- `reverse()`

Array.splice(開始位置、[削除、追加])
引数3つ



サンプル3 カラーセル

カルーセル(スライドショー)




transition表示で配列を使う場合は
:key をつけて特定できるように

setInterval内でthisを使うなら
事前にlet that = this などに変数をつくると
楽



カスタムトランジ ションクラス

既存のCSSライブラリを使う場合



CSSの場合 (Animate.cssなど)

<transition

enter-active-class="animate__animated"

>

JSの場合 (Velocity.js, gasp.js など)

<transition @enter="メソッド名">