

アクセシビリティの高いUIを作ろう

このLTの目標

- アクセシビリティ高いUIはどんなUIなのかわかるようになる
- WAI-ARIAの使い方をちょっとだけわかるようになる

アクセシビリティが低い？高い？

```
HTML
1 <div class="accordion stack">
2   <div class="accordion_item">
3     <div class="accordion_question js-toggleButton">アコーディオン見出し</div>
4     <div class="accordion_answer">Lorem, ipsum dolor sit amet consectetur
      adipiscing elit. Pariatur assumenda itaque nobis eius corrupti enim
      fugit, ullam iusto corporis amet ut praesentium, repudiandae nesciunt
      voluptatem tenetur fugiat nulla error incidunt?</div>
5   </div>
6   <div class="accordion_item">
7     <button type="button" class="accordion_question js-toggleButton">アコー
      ディオン見出し</button>
8     <div class="accordion_answer">Lorem, ipsum dolor sit amet consectetur
      adipiscing elit. Pariatur assumenda itaque nobis eius corrupti enim
      fugit, ullam iusto corporis amet ut praesentium, repudiandae nesciunt
      voluptatem tenetur fugiat nulla error incidunt?</div>
9   </div>
10  <div class="accordion_item">
11    <button type="button" class="accordion_question js-toggleButton"
      aria-controls="hoge" aria-expanded="false">アコーディオン見出し</button>
12    <div id="hoge" class="accordion_answer">Lorem, ipsum dolor sit amet
      consectetur adipiscing elit. Pariatur assumenda itaque nobis eius
      corrupti enim fugit, ullam iusto corporis amet ut praesentium, repudiandae
      nesciunt voluptatem tenetur fugiat nulla error incidunt?</div>
13  </div>
14 </div>
```

アコーディオン見出し

アコーディオン見出し

アコーディオン見出し

[Codepen: アコーディオンのアクセシビリティ比較](#)

それぞれのアコーディオンメニューの違い

- 1つ目は、キーボード操作できない
- 2つ目は、キーボード操作はできるが、スクリーンリーダーユーザーは開閉状態が認識できない
- 3つ目は、キーボード操作ができ、スクリーンリーダーユーザーが開閉状態を認識できる

キーボード操作するユーザーを想定しよう

- 操作できる要素にはキーボードだけでも操作できるように実装する
- HTMLを適切に使う意識を持つだけでもアクセシビリティを高めることができる
 - 例：クリックやタップを想定している要素（遷移はしない）にはbutton要素
 - フォーカスを当てることができる
 - EnterキーまたはSpaceキーでクリック相当の操作をすることができる

スクリーンリーダーユーザーを想定しよう

- 前提としてキーボード操作ができないとスクリーンリーダーでも操作できない
- 読み上げだけで操作できそうか考える
 - どのように読み上げられるのかを実際に聞いてみる
 - WAI-ARIAで読み上げをどう変化させられるかを知る

WAI-ARIAを使うとできること①

- HTMLだけでは伝えられない「状態（ステート）」を伝えられる
- 状態（ステート）とは「動的な情報」
 - 展開されている・展開されていない
 - 押されている・押されていない
 - 選択されている・選択されていない
- aria-expanded
 - 展開されている（true）・展開されていない（false）を伝える属性

WAI-ARIAを使うとできること②

- HTMLだけでは伝えられない「性質（プロパティ）」を伝えられる
- 性質（プロパティ）とは「静的な情報」
 - この要素の名前はこれ
 - この要素はこの要素を操作する
 - この要素はこの要素を参照する
- aria-controls
 - 操作対象の要素を伝える属性
 - 「この要素はあの要素を操作・コントロールするよ」みたいな感じ

まとめ

- Webの使い方は多様であることを頭に入れておこう
 - キーボード操作
 - スクリーンリーダー
 - 他にもいろいろ
- HTMLを適切に使う意識を持とう
- HTMLだけでは伝えられない情報がある場合は、WAI-ARIAの使用を検討しよう