Client-Side Web Development

Class 1.2

Today’s Topics

- Web Accessibility
- **Lab:** Keeping Things Accessible

Any Questions?

Web Accessibility

Web Accessibility is the experience of users who might be outside the narrow range of the "typical" user, who might access or interact with things differently than you expect.

"Non-typical" Users

- Visual Impaired
- Physically Disabled
- Mobile Users
- International Users

Making a site accessible not only helps the non-typical user, it general makes it site easier to use for everyone

Web Content Accessibility Guidelines (WCAG) 2.1

https://www.w3.org/TR/WCAG21/

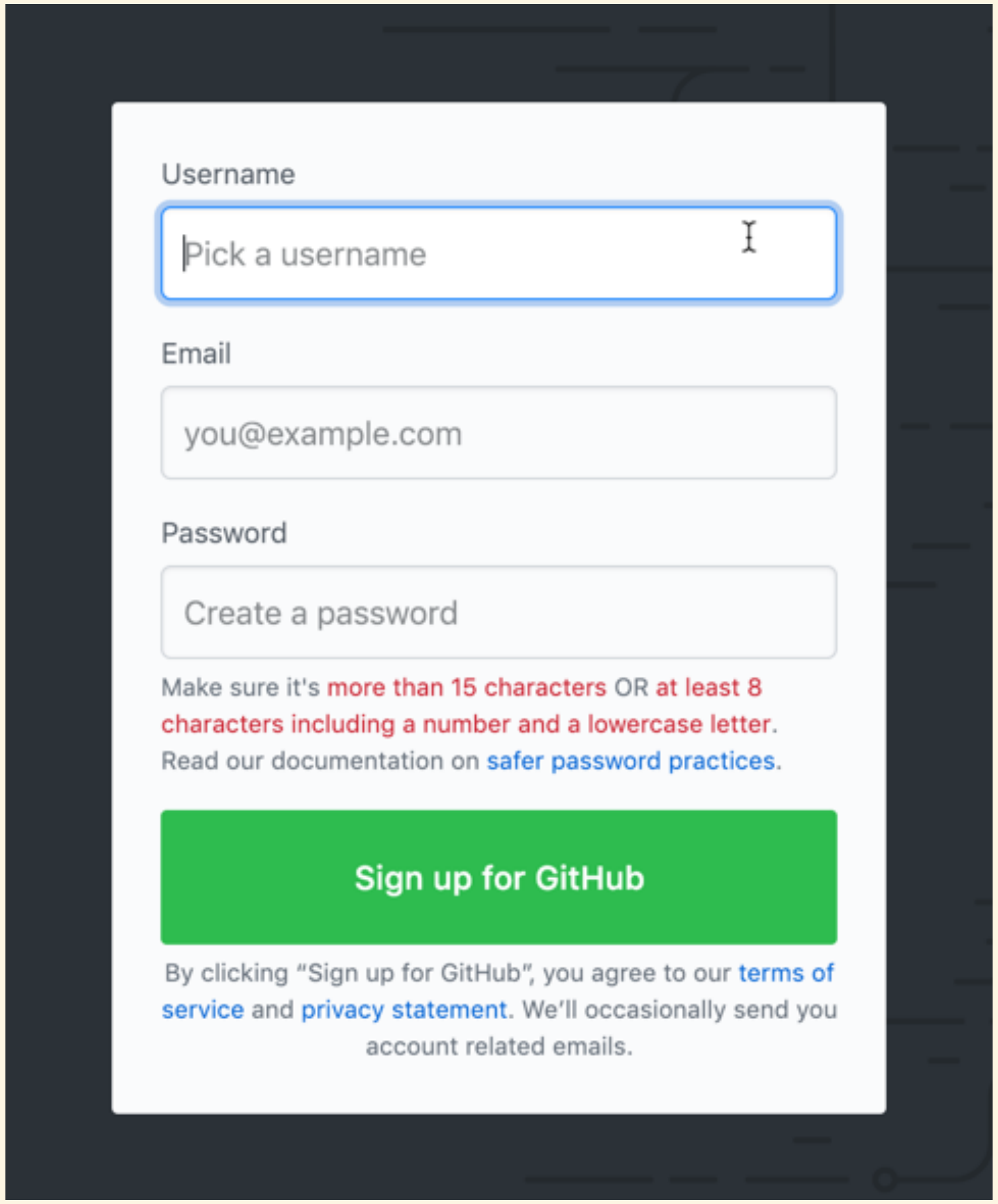
WebAIM WCAG Checklist

https://webaim.org/standards/wcag/checklist

Focus

Focus **determines where keyboard events go in the page at any given moment.**

Focus



Username

Pick a username

Email

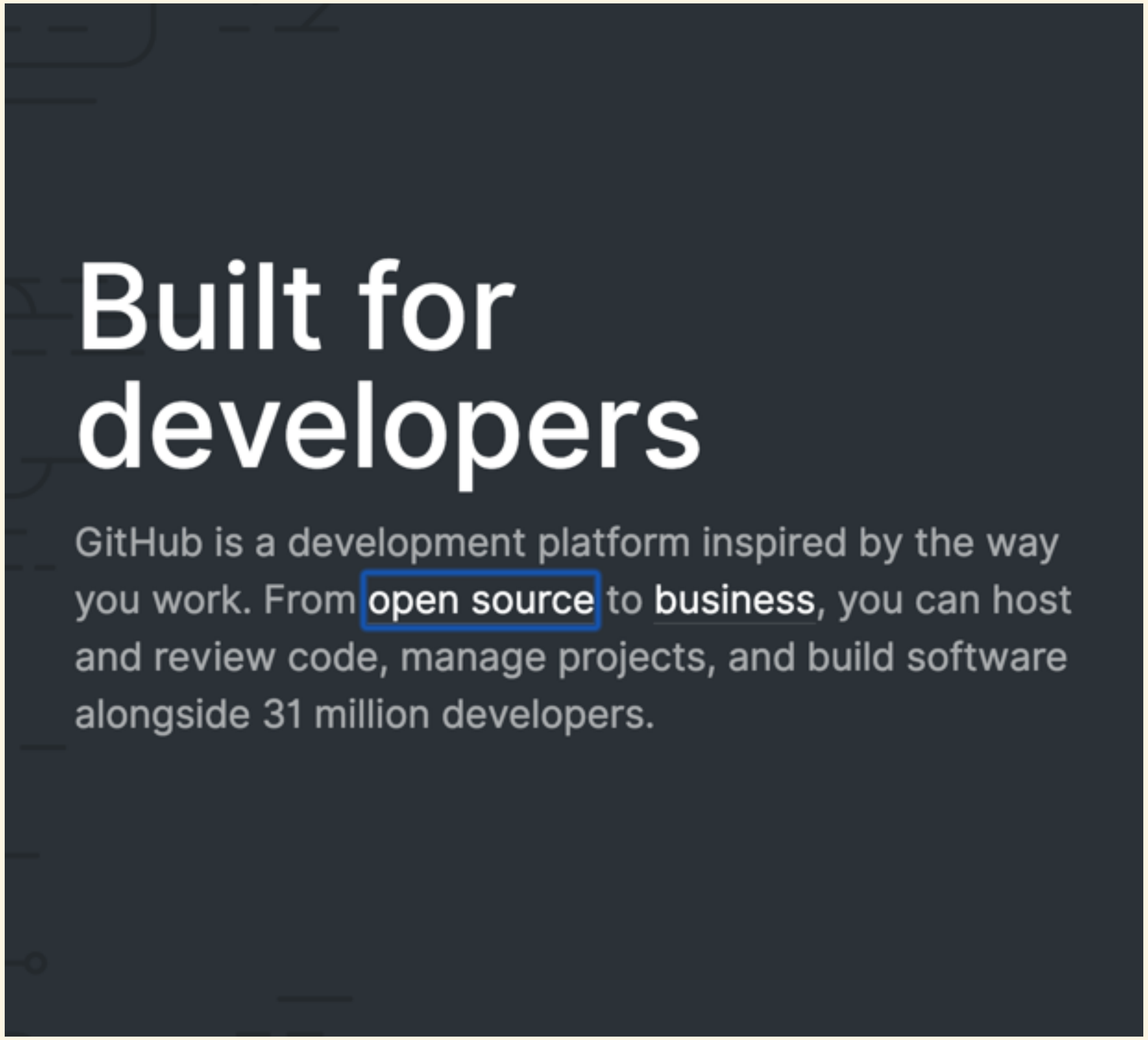
you@example.com

Password

Create a password

Make sure it's more than 15 characters OR at least 8 characters including a number and a lowercase letter. Read our documentation on safer password practices.

By clicking "Sign up for GitHub", you agree to our terms of service and privacy statement. We'll occasionally send you account related emails.



Built for developers

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 31 million developers.

DOM Order and Tab Order

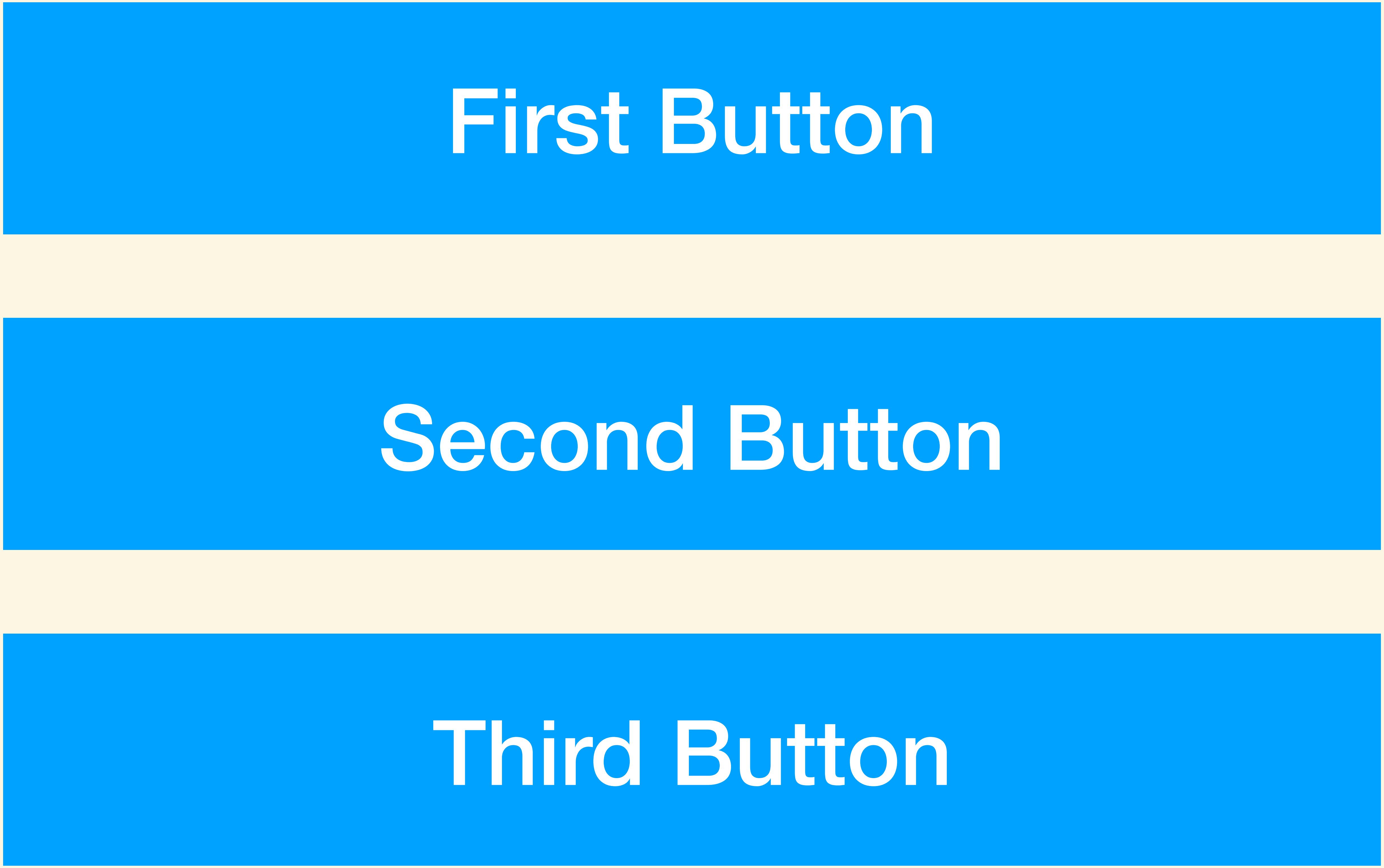
DOM Order **is the order in which elements are laid inside the HTML**

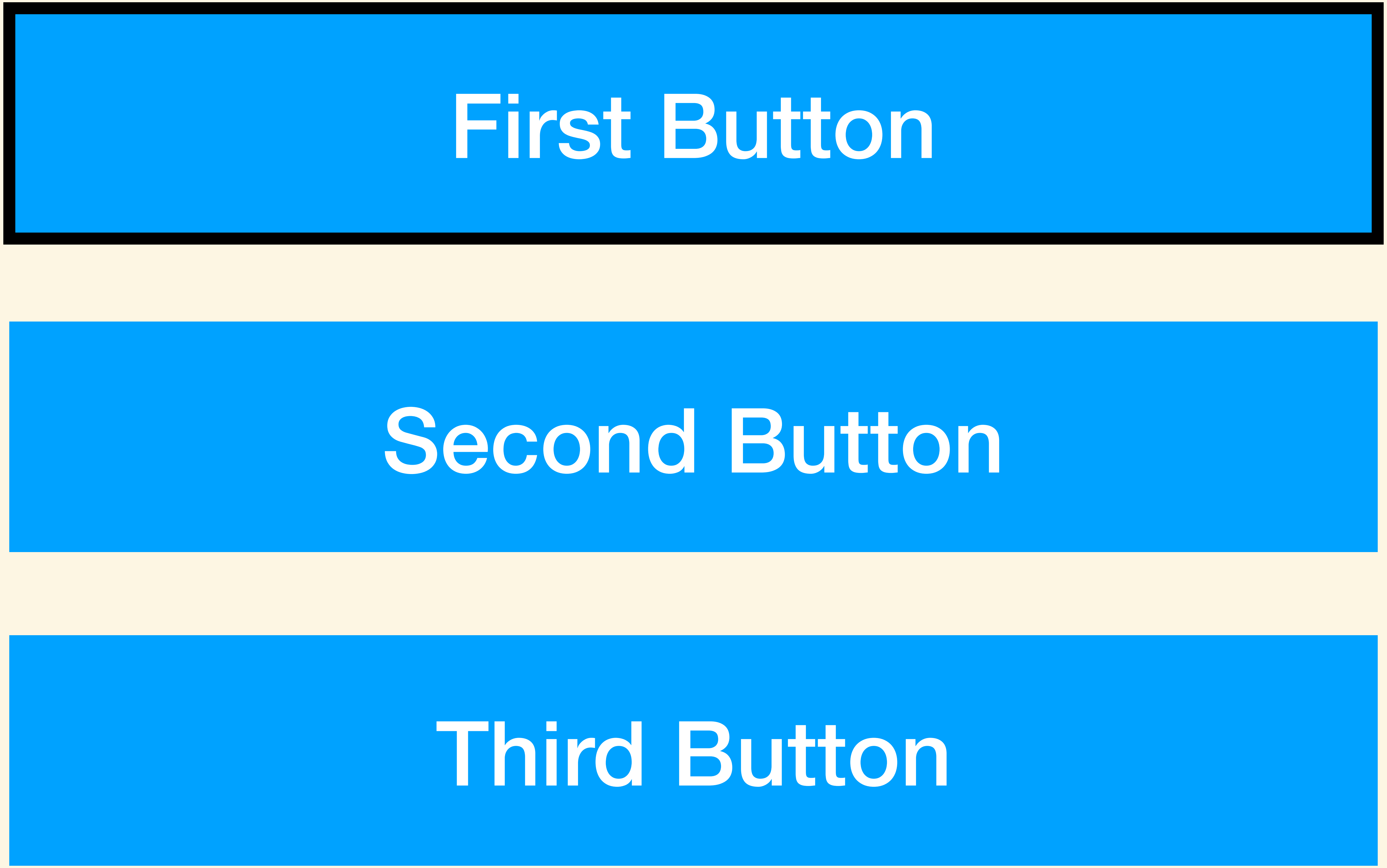
Tab Order **is the order in which elements are focused when tabbing through the page**

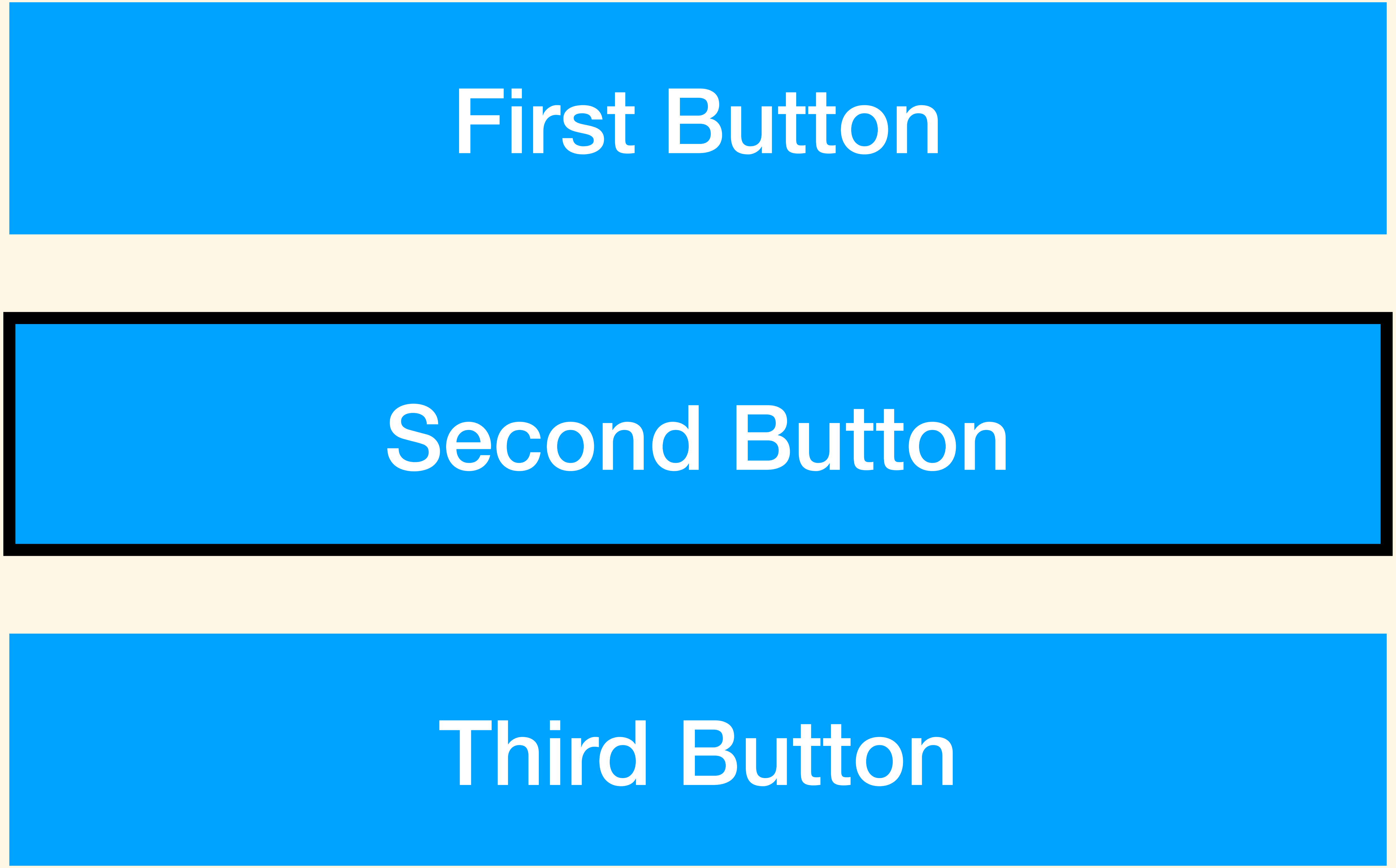
By default, *Tab Order* will match *DOM Order*

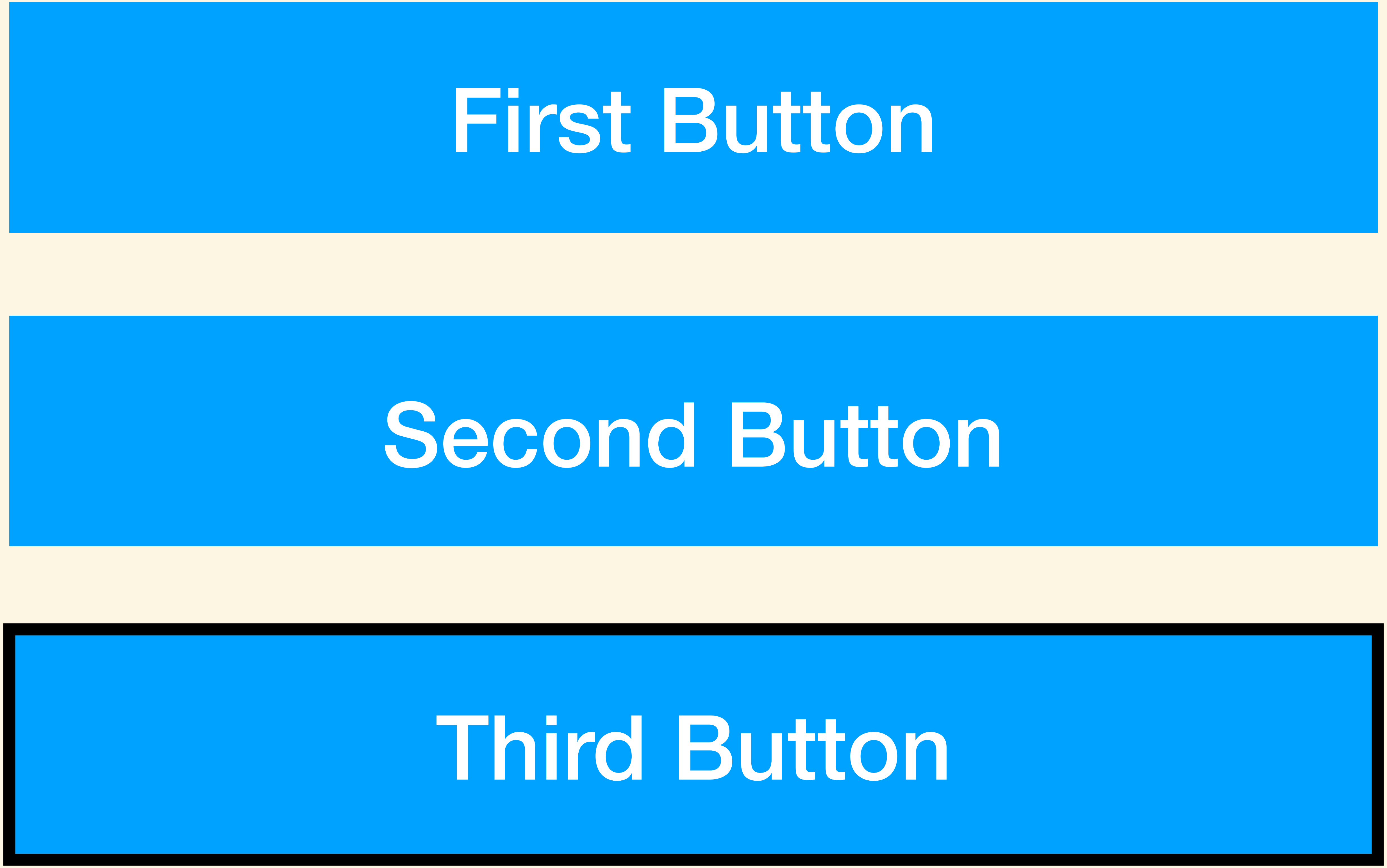
CSS can change the order in which elements appear, but that does not affect *DOM Order*

```html
<div class="buttons">
  <button>First Button</button>
  <button>Second Button</button>
  <button>Third Button</button>
</div>
```


First Button
Second Button
Third Button


First Button
Second Button
Third Button

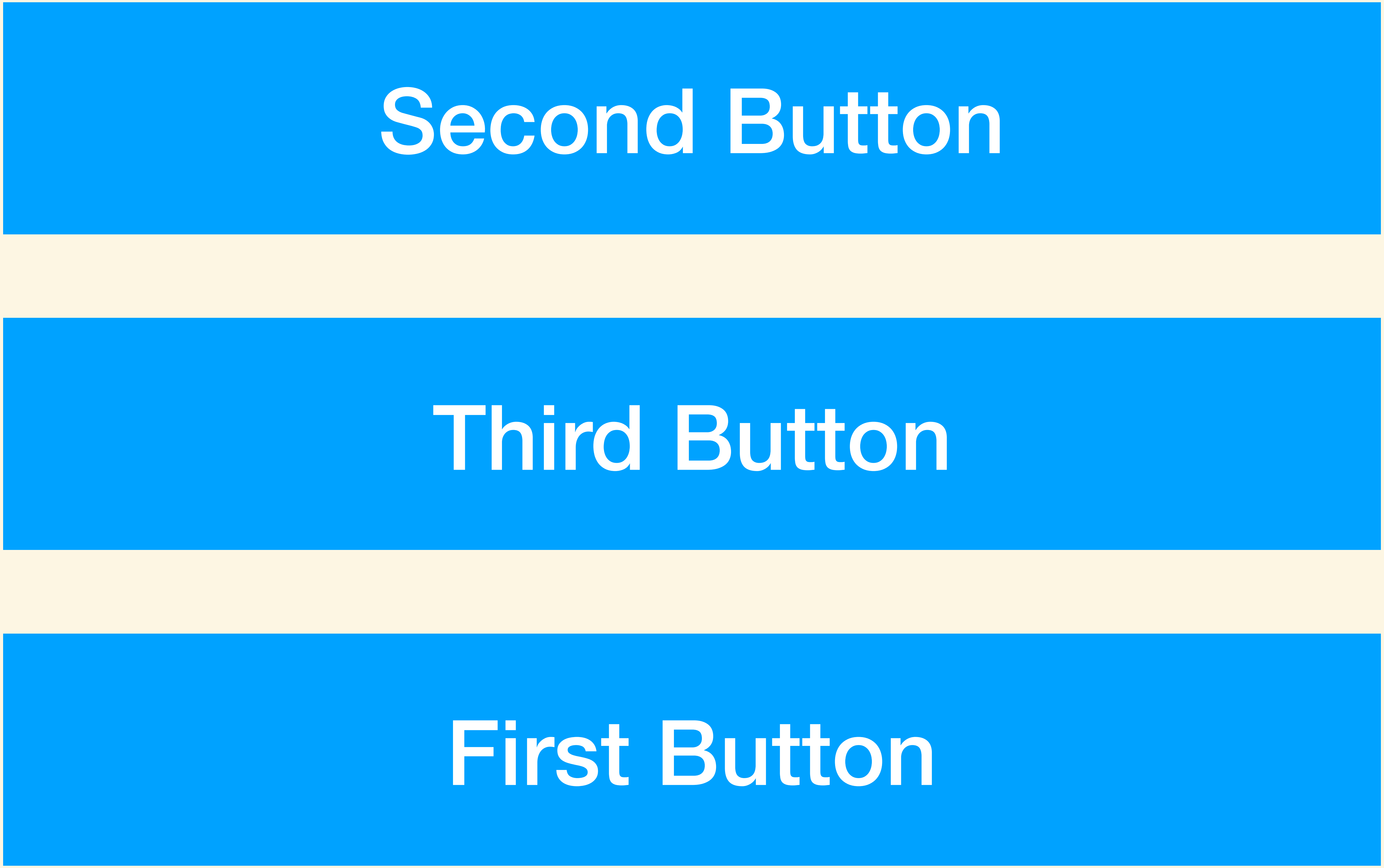

First Button
Second Button
Third Button

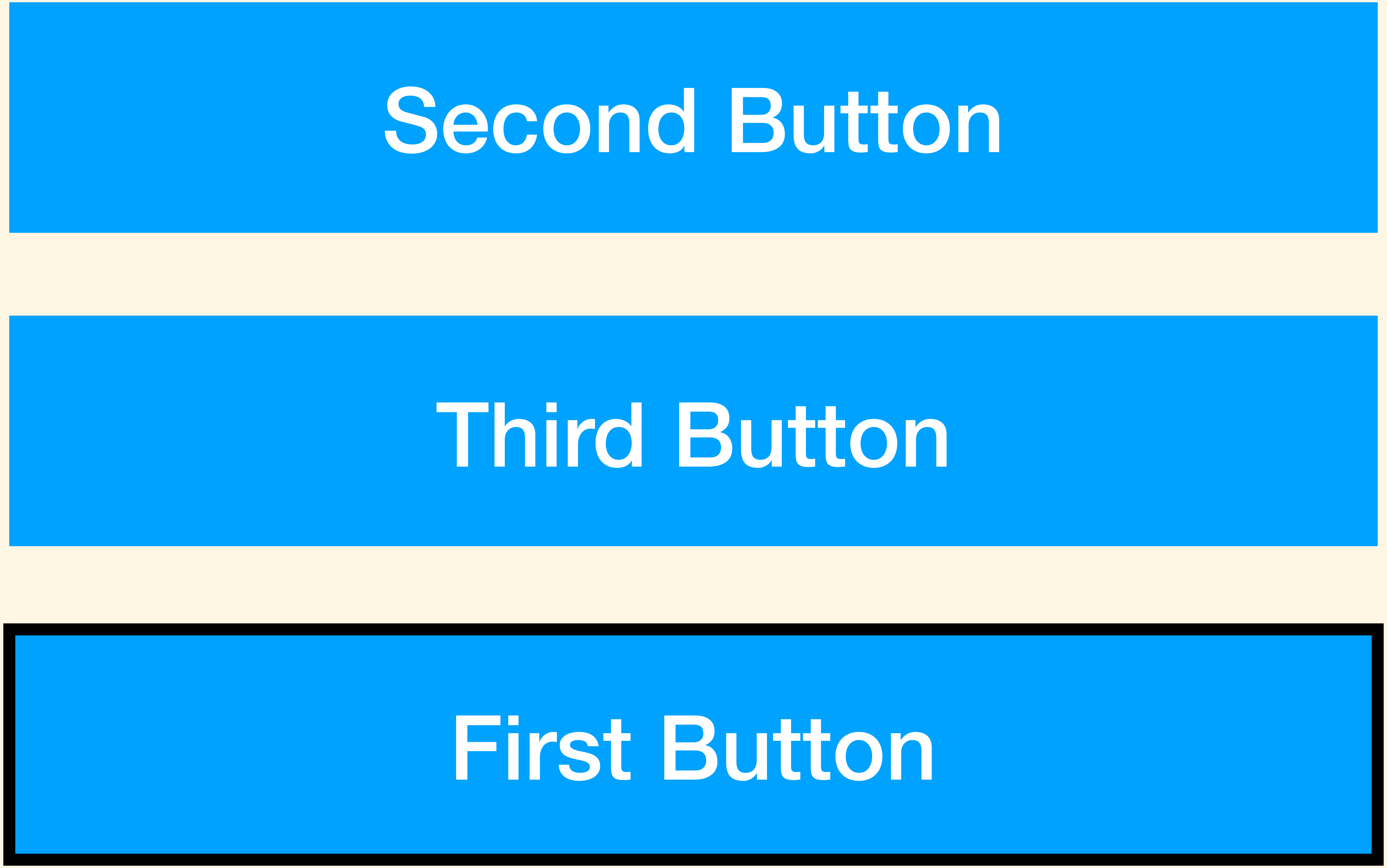

First Button
Second Button
Third Button

```css
.buttons {
  display: flex;
  flex-direction: column;
}

button:first-child {
  order: 2;
}

button:last-child {
  order: 1;
}
```

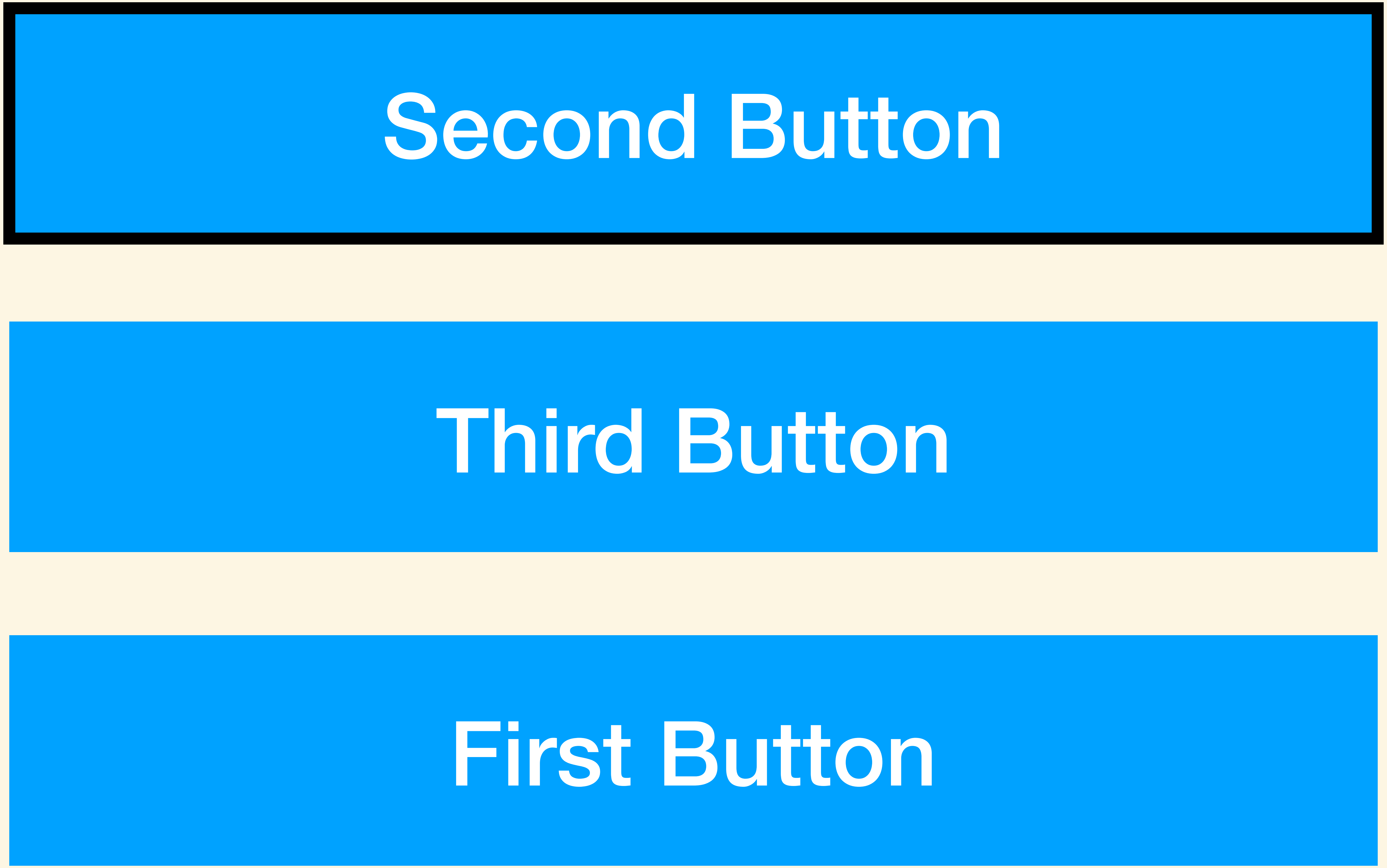

Second Button
Third Button
First Button



Second Button

Third Button

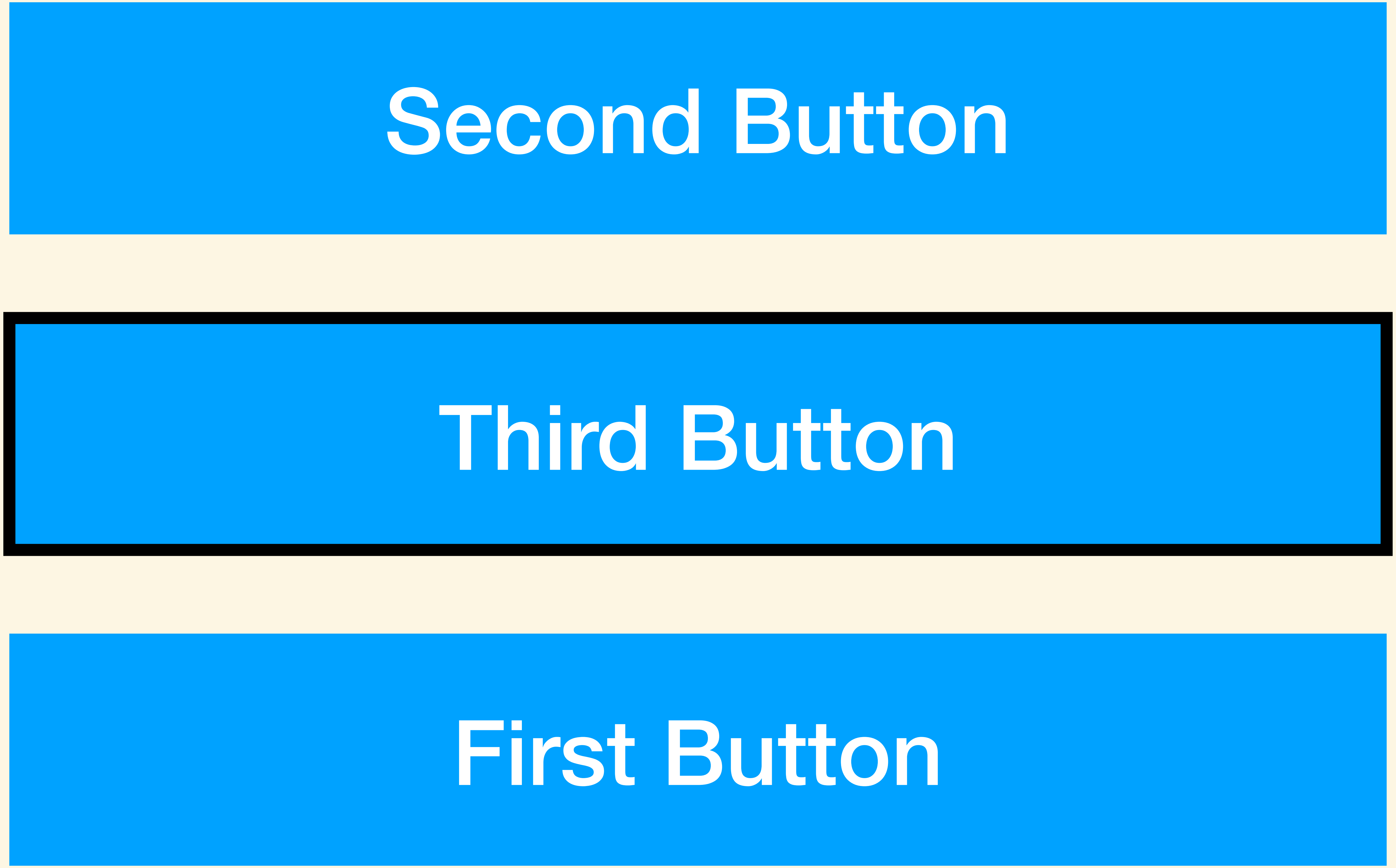
First Button



Second Button

Third Button

First Button


Second Button
Third Button
First Button

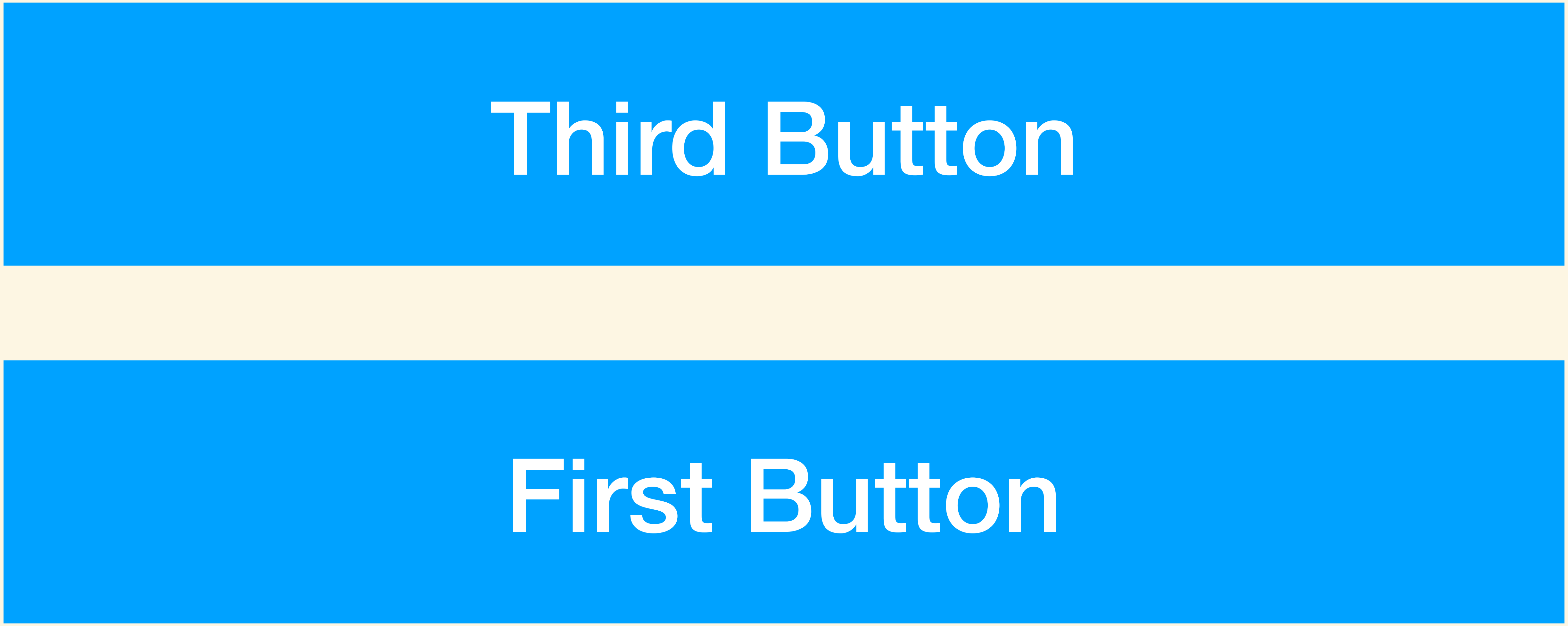
Off Screen / Invisible Content

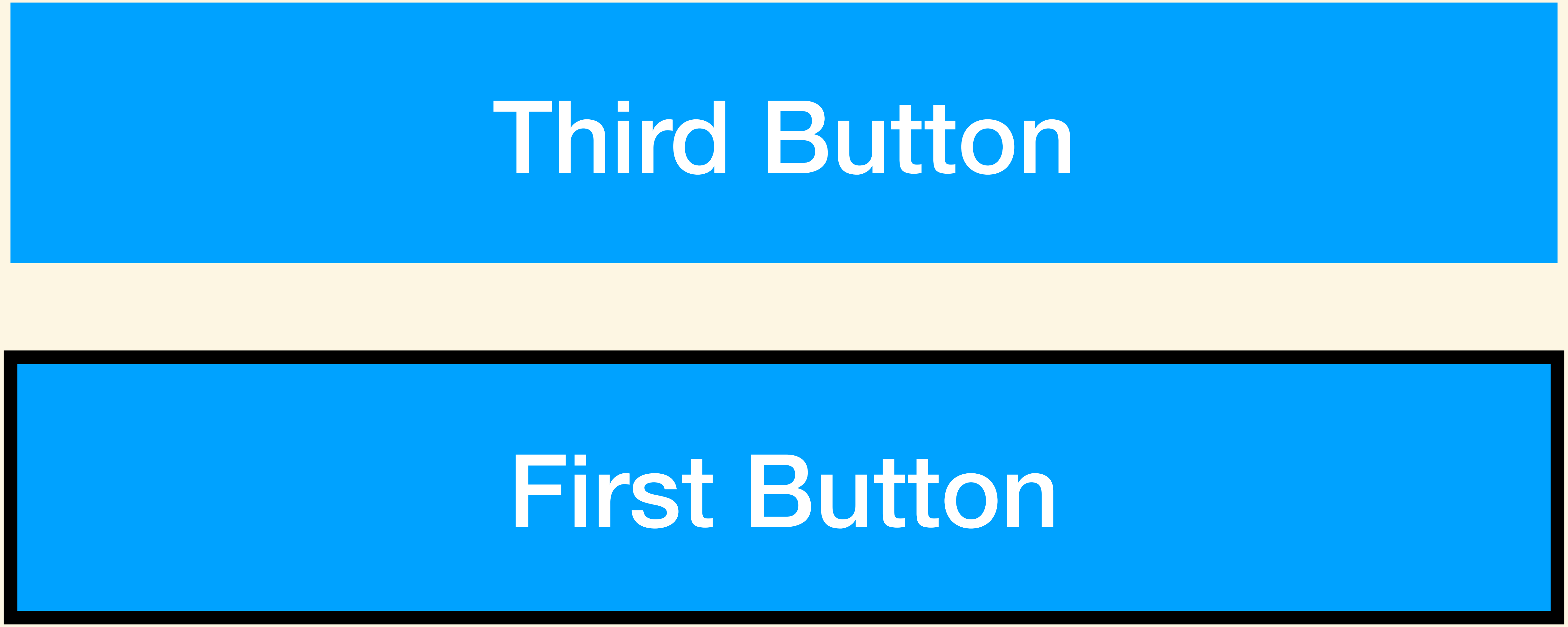
```
.buttons {
  display: flex;
  flex-direction: column;
}

button:first-child {
  order: 2;
}

button:last-child {
  order: 1;
}

button:nth-child(2) {
  opacity: 0;
}
```


Third Button
First Button

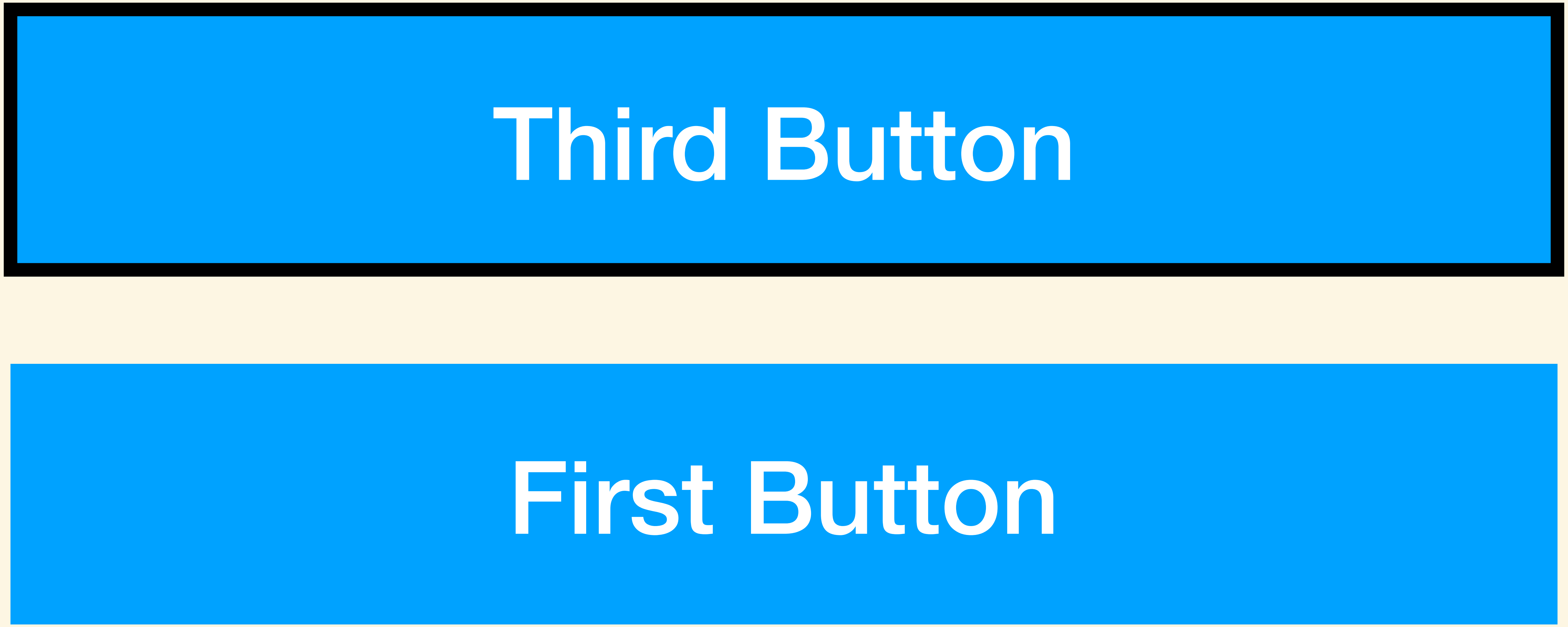


Third Button

First Button

Third Button

First Button



Third Button

First Button

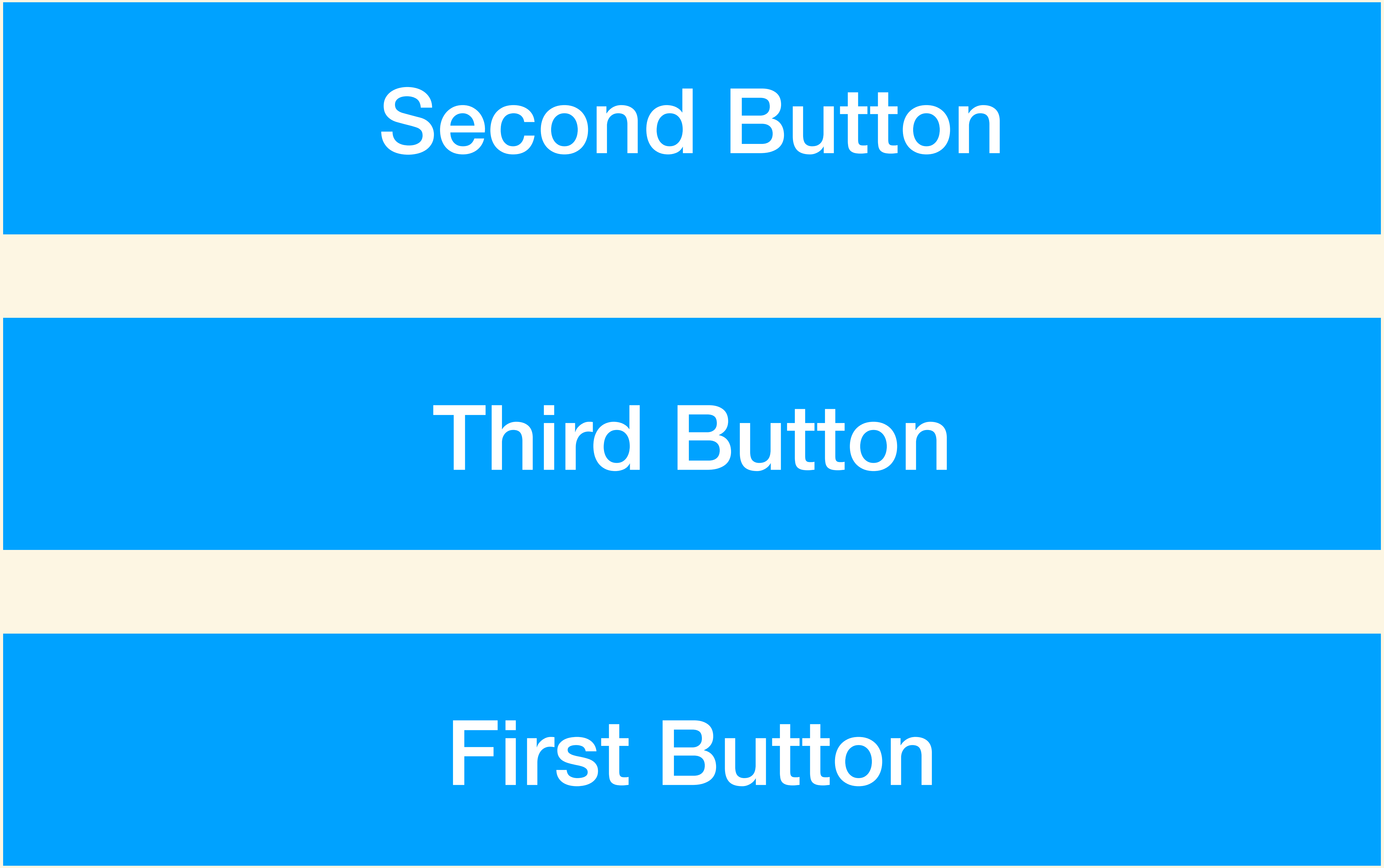
Use `display: none` or `visibility: hidden` to remove an element from Tab Order

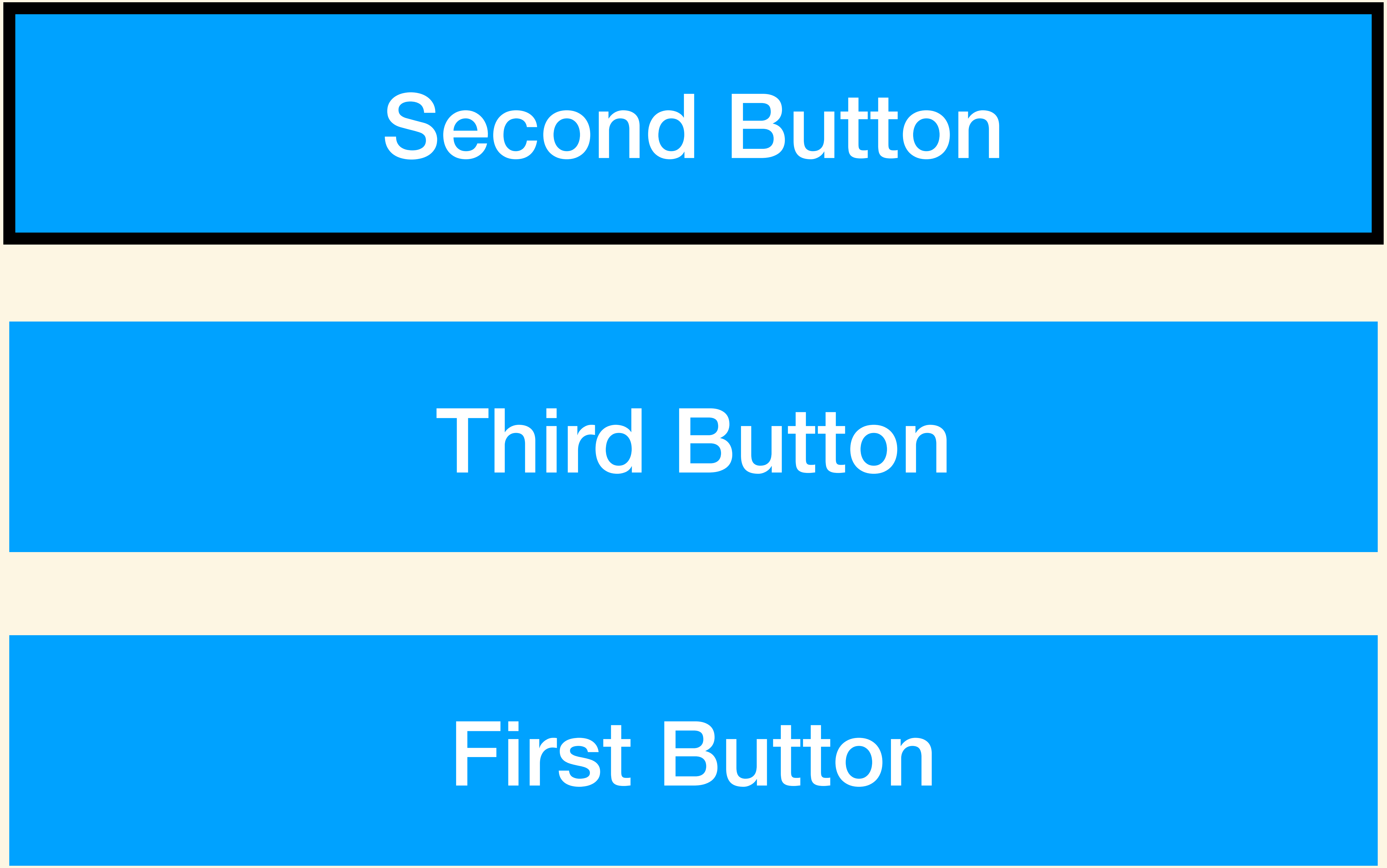
Tabindex

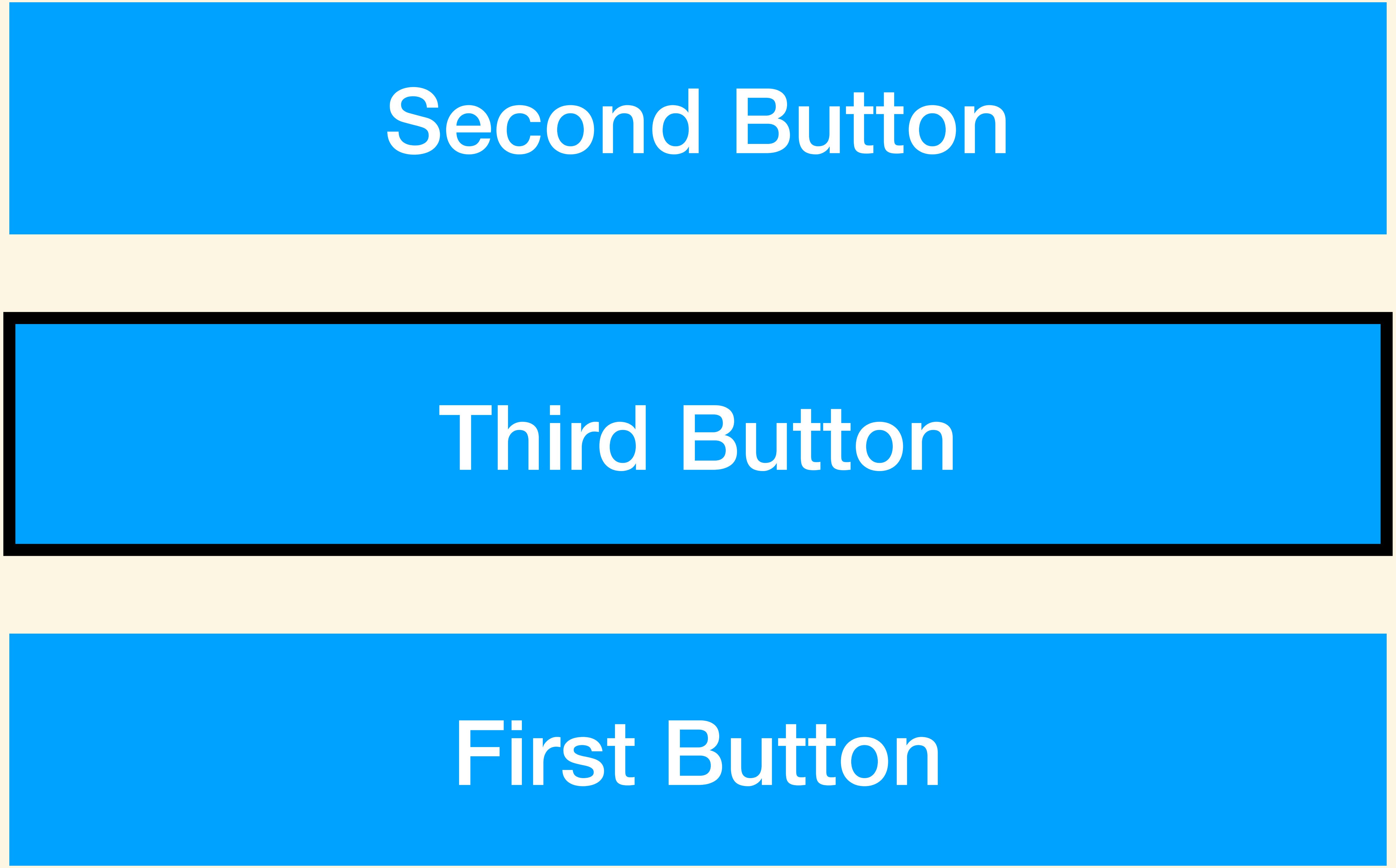
Tabindex **can explicitly set an element's tab position**

Tabindex can be applied to any HTML element

```
<div class="buttons">
  <button tabindex="3">First Button</button>
  <button tabindex="1">Second Button</button>
  <button tabindex="2">Third Button</button>
</div>
```


Second Button
Third Button
First Button

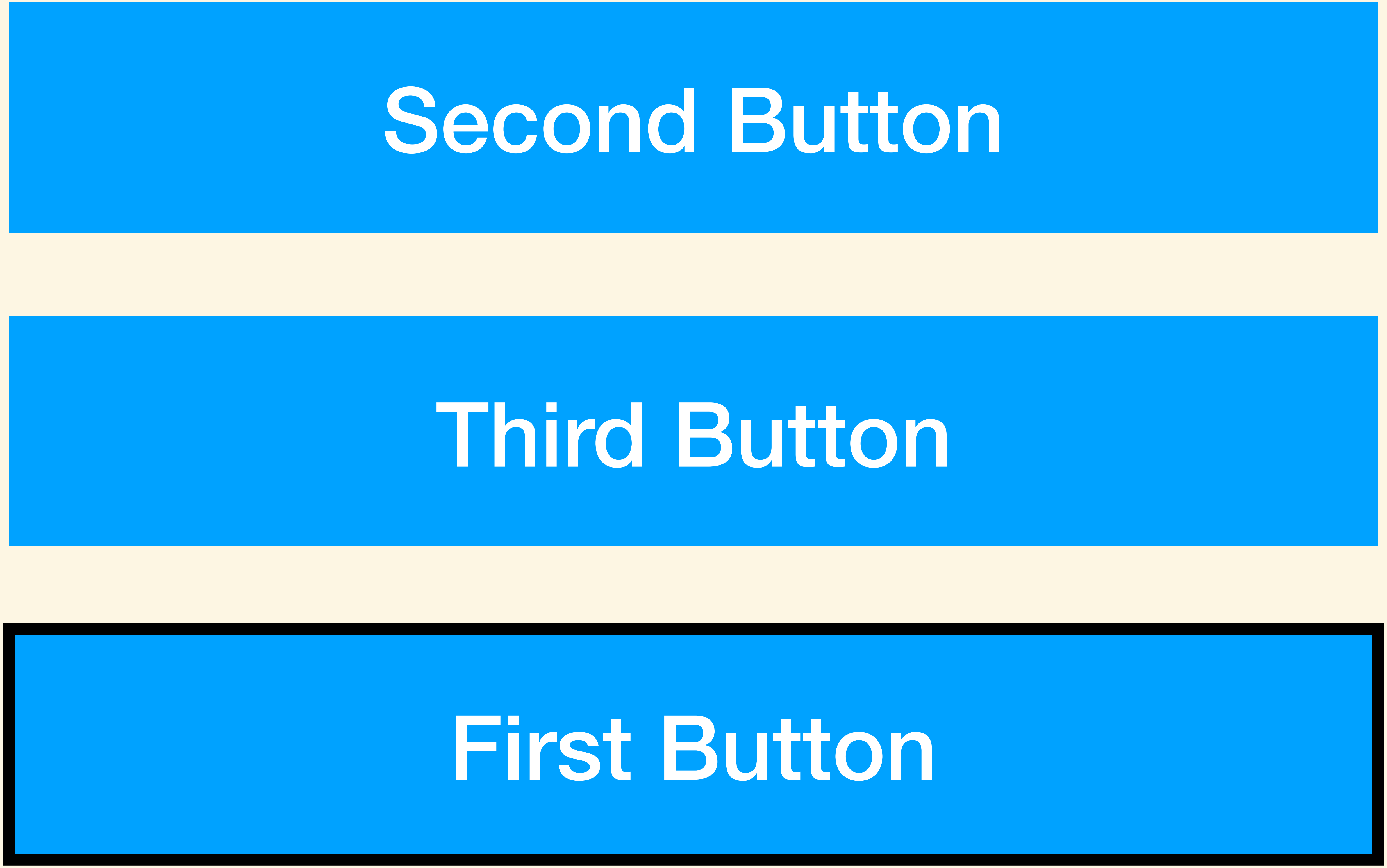

Second Button
Third Button
First Button



Second Button

Third Button

First Button



Second Button

Third Button

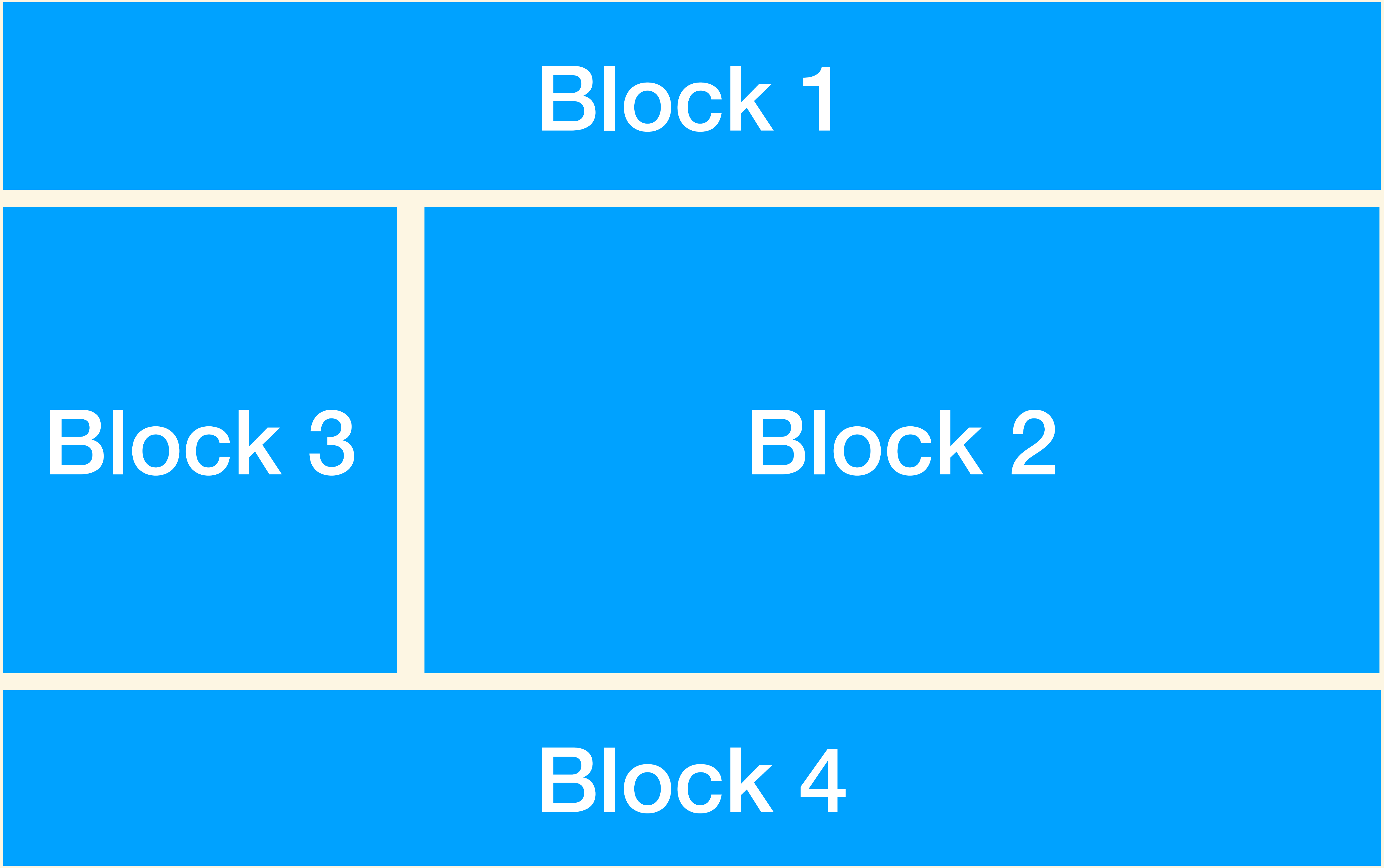
First Button

Semantic HTML

Semantic HTML **is the use of HTML that reinforces the meaning or purpose of an element**

Assistive technologies use the HTML to get content as well as meaning or purpose

```
<div class="block1">Block 1</div>
<div class="block2">Block 2</div>
<div class="block3">Block 3</div>
<div class="block4">Block 4</div>
```



Block 1

Block 3

Block 2

Block 4

```html
<header class="block1">Block 1</header>
<article class="block2">Block 2</article>
<aside class="block3">Block 3</aside>
<footer class="block4">Block 4</footer>
```

Order and Hierarchy

It is important that the HTML is in a logical order with proper hierarchy

Using heading tags, <h1>, <h2>, etc., helps provides a natural structure of the content.

Text Alternative

Text Alternative **is a description of any visual or audio element that provides content in plain text**

All images, including image buttons and image maps, should have an appropriate text alternative

Decorative images or images already described in text should be given null or empty alt text

Form inputs should have text labels

Transcripts or captions should be provided for video and audio content

Styling

CSS also plays a big part in accessibility, so there are several key guidelines to keep in mind

Make sure the page is responsive and zoomable

Make sure the lines of text are not too long and that there is enough space between lines

Make sure there is enough contrast between the text, background, and images

Make sure that the focus state of an element is visible, which can be set using the `:focus` pseudo-class

ARIA

Accessible Rich Internet Application

ARIA **is used provide semantics to custom HTML structures or to change/add meaning to an element**

```html
<li tabindex="0" class="checkbox" checked>
  Learn Web Accessibility
</li>
```

Learn Web Accessibility

```html
<li tabindex="0" class="checkbox" role="checkbox"
checked aria-checked="true">
  Learn Web Accessibility
</li>
```

For next class...

- Flexbox
- **Exercise:** Flexbox Froggy