TELUS Component Library

Tags

Multi-Platform Component | [Figma UI KIT](#)

First tag ✛   Second tag ✛   Third tag ✛   Fourth tag ✛

Fifth tag ✛

```
<Tags
  items={[
    { label: 'First tag', id: 'first' },
    { label: 'Second tag', id: 'second' },
    { label: 'Third tag', id: 'third' },
    { label: 'Fourth tag', id: 'fourth' },
    { label: 'Fifth tag', id: 'fifth' }
  ]}
/>
```
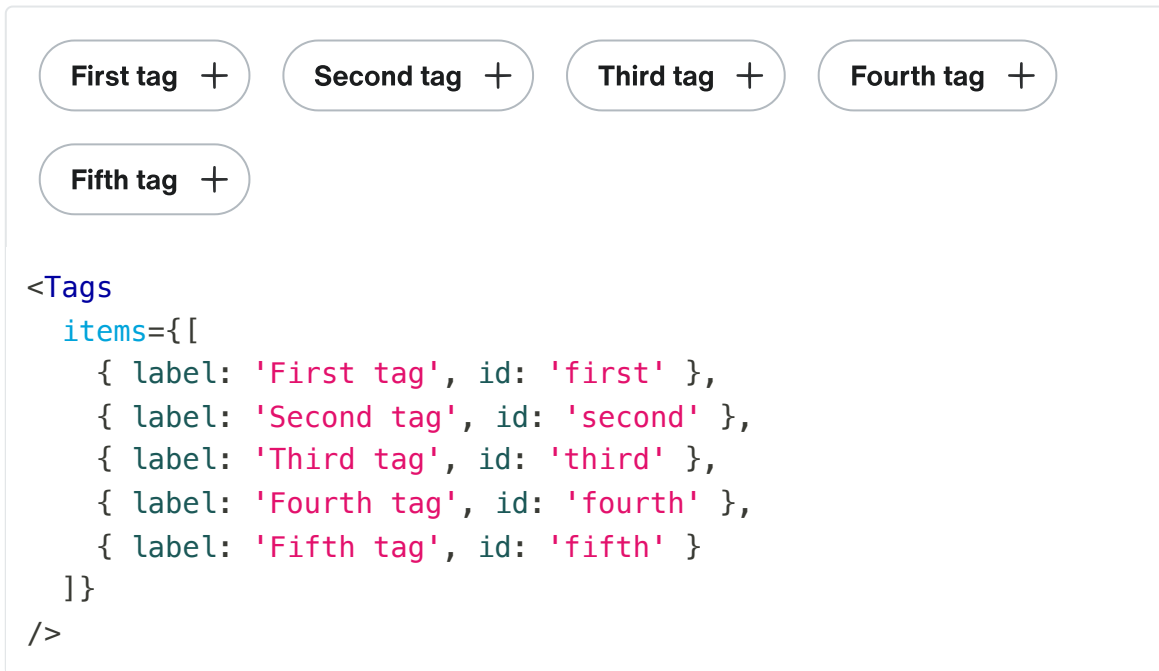
Introduction

Tags allows the selection of one or more items, typically used for filtering content.

Spacing between buttons increases on wider viewports (`lg` and `xl`).

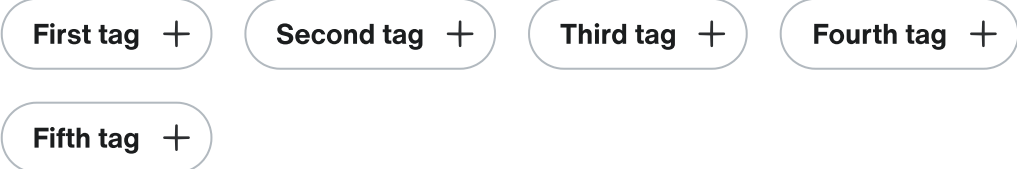[Follow the appropriate instructions](#) to add this component in to your app.

Guidance

Tags should have very simple, short labels, ideally one word or two.

`maxValues` prop

The prop `maxValues` may be passed a number to limit how many items are selelcted.

By default, any number of tags may be selected.

First tag +   Second tag +   Third tag +   Fourth tag +

Fifth tag +

```
<Tags
  items={[
    { label: 'First tag', id: 'first' },
    { label: 'Second tag', id: 'second' },
    { label: 'Third tag', id: 'third' },
    { label: 'Fourth tag', id: 'fourth' },
    { label: 'Fifth tag', id: 'fifth' }
  ]}
  maxValues={2}
/>
```

onChange prop

An onChange function may be provided, which is called when a Tags item is toggled.

First tag +   Second tag +   Third tag +   Fourth tag +

Fifth tag +

```
<Tags
  items={[
    { label: 'First tag', id: 'first' },
    { label: 'Second tag', id: 'second' },
    { label: 'Third tag', id: 'third' },
    { label: 'Fourth tag', id: 'fourth' },
    { label: 'Fifth tag', id: 'fifth' }
  ]}
  onChange={(state) => setTimeout(() => alert(`Tags has
selected: ${state}`), 100)}
/>
```

Controlled usage with `values` prop

Pass an array of item ids to the `values` prop as well as `onChange` to control the Tags from a parent.

This is usually the easiest way to control subsequent lists of content with Tags.

Accessibility

`Tags` buttons will be treated like checkboxes by accessibility tools; unless `maxValues` is set to `1`, in which case they will be treated like radio buttons.

Platform considerations

The component is available on both native platforms and web.

Props

Name	Type	Platform	Default	Description
items	arrayOf	standard	[]	The options a user may select
readOnly		standard	false	
inactive		standard	false	
tokens	tokens	standard		System tokens prop, see tokens for more details
variant	variant	standard		System variant prop, see variants for more details

Name	Type	Platform	Default	Description
maxValues	number	standard		If provided, sets a maximum number of items a user may select at once.
onChange	func	standard		If provided, this function is called when the current selection is changed and is passed an array of the `id`s of all currently selected `items`.
values	arrayOf	standard		If the selected item(s) in the button group are to be controlled externally by a parent component, pass an array of strings as well as an `onChange` handler. Passing an array for "values" makes the ButtonGroup a "controlled" component that expects its state to be handled via `onChange` and so doesn't handle it itself.
initialValues	arrayOf	standard		If `values` is not passed, making the ButtonGroup an

Name	Type	Platform	Default	Description
				"uncontrolled" component managing its own selected state, a default set of selections may be provided. Changing the `initialValues` does not change the user's selections.

Tokens

In exceptional circumstances, the following tokens can be passed to this component to override its default styles. **Do not do this unless absolutely necessary.** Read more about overriding styles.

▶ View Tokens

Variants

This component does not have any stylistic variants.

Feedback

- Spotted a problem with this component? Raise an issue on GitHub
- See any existing issues for this component
- Contact the team on slack in #ds-support