TreeSelect

- Props
  - children
  - disabled
  - hideLabelFromVision
  - help
  - label
  - labelPosition
  - noOptionLabel
  - onChange
  - options
  - prefix
  - selectedId
  - size
  - suffix
  - tree
  - variant

See the WordPress Storybook for more detailed, interactive documentation.

Generates a hierarchical select input.

```
import { useState } from 'react';
import { TreeSelect } from '@wordpress/components';

const MyTreeSelect = () => {
    const [ page, setPage ] = useState( 'p21' );

    return (
        <TreeSelect
```

```
            label="Parent page"
            noOptionLabel="No parent page"
            onChange={ ( newPage ) => setPage( newPage ) }
            selectedId={ page }
            tree={ [
                {
                    name: 'Page 1',
                    id: 'p1',
                    children: [
                        { name: 'Descend 1 of page 1', id: 'p11' },
                        { name: 'Descend 2 of page 1', id: 'p12' },
                    ],
                },
                {
                    name: 'Page 2',
                    id: 'p2',
                    children: [
                        {
                            name: 'Descend 1 of page 2',
                            id: 'p21',
                            children: [
                                {
                                    name: 'Descend 1 of Descend 1 of page 2',
                                    id: 'p211',
                                },
                            ],
                        },
                    ],
                },
            ] }
        />
    );
}
```

Props

`children`

- Type: `ReactNode`
- Required: No

As an alternative to the `options` prop, `optgroup`s and `options` can be passed in as `children` for more customizability.

`disabled`

- Type: `boolean`
- Required: No
- Default: `false`

If true, the `input` will be disabled.

`hideLabelFromVision`

- Type: `boolean`
- Required: No
- Default: `false`

If true, the label will only be visible to screen readers.

`help`

- Type: `ReactNode`
- Required: No

Additional description for the control.

Only use for meaningful description or instructions for the control. An element containing the description will be programmatically associated to the BaseControl by the means of an `aria-describedby` attribute.

`label`

- Type: `ReactNode`
- Required: No

If this property is added, a label will be generated using label property as the content.

`labelPosition`

- Type: `"top" | "bottom" | "side" | "edge"`
- Required: No
- Default: `'top'`

The position of the label.

noOptionLabel

- Type: `string`
- Required: No

If this property is added, an option will be added with this label to represent empty selection.

onChange

- Type: `((value: string, extra?: { event?: ChangeEvent<HTMLSelectElement, Element>; }) => void) | undefined`
- Required: No

A function that receives the value of the new option that is being selected as input.

options

- Type: `readonly ({ label: string; value: string; } & Omit<OptionHTMLAttributes<HTMLOptionElement>, "label" | "value">)[]`
- Required: No

An array of option property objects to be rendered,
each with a `label` and `value` property, as well as any other
`<option>` attributes.

prefix

- Type: `ReactNode`
- Required: No

Renders an element on the left side of the input.

By default, the prefix is aligned with the edge of the input border, with no padding.
If you want to apply standard padding in accordance with the size variant, wrap the element in the provided `<InputControlPrefixWrapper>` component.

```
import {
  __experimentalInputControl as InputControl,
  __experimentalInputControlPrefixWrapper as InputControlPrefixWrapper,
} from '@wordpress/components';

<InputControl
  prefix={<InputControlPrefixWrapper>@</InputControlPrefixWrapper>}
/>
```

`selectedId`

- Type: `string`
- Required: No

The id of the currently selected node.

`size`

- Type: `"default" | "small" | "compact"`
- Required: No
- Default: `'default'`

Adjusts the size of the input.

`suffix`

- Type: `ReactNode`
- Required: No

Renders an element on the right side of the input.

By default, the suffix is aligned with the edge of the input border, with no padding.
If you want to apply standard padding in accordance with the size variant, wrap the element in the provided `<InputControlSuffixWrapper>` component.

```
import {
  __experimentalInputControl as InputControl,
  __experimentalInputControlSuffixWrapper as InputControlSuffixWrapper,
} from '@wordpress/components';

<InputControl
```

```
  suffix={<InputControlSuffixWrapper>%</InputControlSuffixWrapper>}
/>
```

tree

- Type: `Tree[]`
- Required: No

An array containing the tree objects with the possible nodes the user can select.

variant

- Type: `"default" | "minimal"`
- Required: No
- Default: `'default'`

The style variant of the control.

□□□ #2
□ □□□ □□□ 2026-10-31 18:15:21 UTC
□ □□□ □□□ 2026-10-31 18:15:22 UTC