

Vue Slots

Slots

- A component can insert child nodes (specified in parents) into its template with `<slot>optional default content</slot>`
- If no child nodes are provided, default content is used instead
- Child nodes can be any HTML or Vue elements
- Example: a custom named `Select` can insert child `option` elements

```
<Select path="favorite.color">
  <option>red</option>
  <option>green</option>
  <option>blue</option>
</Select>
```

- Template for `select` can use `<slot></slot>` to indicate where all the child nodes should be inserted

Default Slot Example

```
<ShippingLabel>
  Mark Volkmann<br />
  123 Some Street<br />
  Somewhere, Some State 12345
</ShippingLabel>

<ShippingLabel /> no content
```

Ship To:
Mark Volkmann
123 Some Street
Somewhere, Some State 12345
Ship To:
unknown

ShippingLabel.vue

```
<template>
  <div>
    <label>Ship To:</label>
    <slot>unknown</slot>
  </div>
</template>

<script>
export default {
  name: 'ShippingLabel'
};
</script>

<style scoped>
label {
  display: block;
  font-weight: bold;
}
</style>
```

Named Slots

- A template can contain multiple named slots
- These add a **name** attribute to `<slot>`
- Each can optionally define default content
- Parent components specify content for each named slot with
`<template slot="slot-name">the content</template>`
- Alternatively, the `slot` attribute can be used on "normal" HTML elements
- Example: `<div slot="slot-name">the content</div>`
- Even when named slots are used, there can be an unnamed slot that is used for any content not marked with a specific slot name

Named Slots Example

```
<ShippingLabel>
  <div slot="address">
    123 Some Street<br />
    Somewhere, Some State 12345
  </div>
  <div slot="name">Mark Volkmann</div>
</ShippingLabel>
```

```
<ShippingLabel /> no content
```

Ship To:
Mark Volkmann
123 Some Street
Somewhere, Some State 12345
Ship To:
unknown

ShippingLabel.vue

```
<template>
  <div>
    <label>Ship To:</label>
    <slot name="name"><div>unknown</div></slot>
    <slot name="address" />
  </div>
</template>

<script>
export default {
  name: 'ShippingLabel'
};
</script>

<style scoped>
label {
  display: block;
  font-weight: bold;
}
</style>
```

Scoped Slots



- Allows parent components to use data from child elements in slot content
- **Rarely needed and somewhat complex to set up**

```
<Menu>
  <template v-slot:soupOfTheDay="slotProps">
    {{ slotProps.defaultSoup }}
  </template>
</Menu>
```

Menu instance in a parent component uses data from Menu component for content in the slot named "soupOfTheDay". The slot name after v-slot: can also be default to specify content for default slot.

Menu.vue

```
<template>
  <div>
    <label>Soup of the day: </label>
    <slot
      name="soupOfTheDay"
      :defaultSoup="popularSoup">
      unknown
    </slot>
  </div>
</template>

<script>
export default {
  name: 'Menu',
  data() {
    return {popularSoup: 'French Onion'};
  }
};
</script>
```