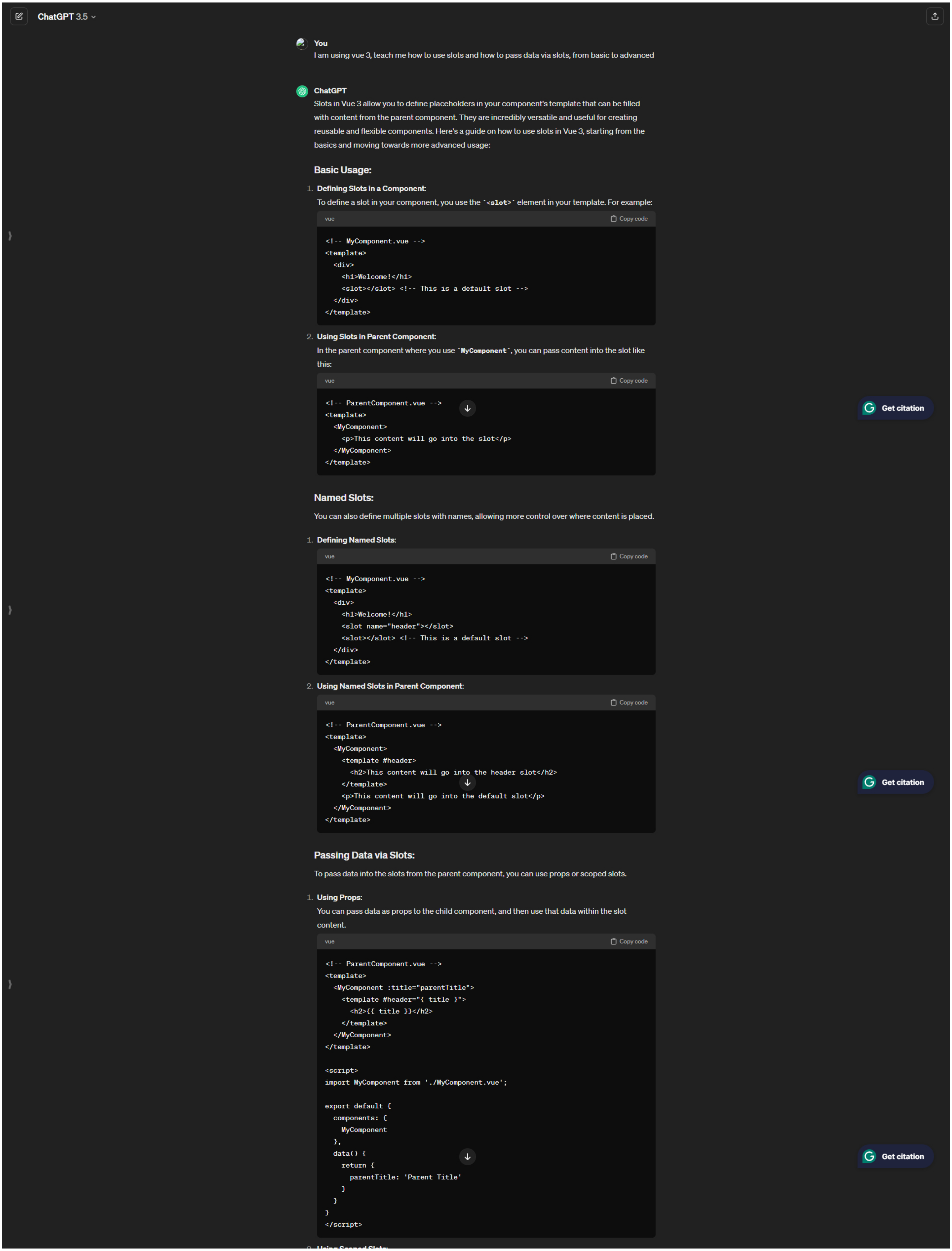


You

I am using vue 3, teach me how to use slots and how to pass data via slots, from basic to advanced

ChatGPT

Slots in Vue 3 allow you to define placeholders in your component's template that can be filled with content from the parent component. They are incredibly versatile and useful for creating reusable and flexible components. Here's a guide on how to use slots in Vue 3, starting from the basics and moving towards more advanced usage:

Basic Usage:

1. **Defining Slots in a Component:**
   To define a slot in your component, you use the `` `<slot>` `` element in your template. For example:

```vue
<!-- MyComponent.vue -->
<template>
  <div>
    <h1>Welcome!</h1>
    <slot></slot> <!-- This is a default slot -->
  </div>
</template>
```

2. **Using Slots in Parent Component:**
   In the parent component where you use `` `MyComponent` ``, you can pass content into the slot like this:

```vue
<!-- ParentComponent.vue -->
<template>
  <MyComponent>
    <p>This content will go into the slot</p>
  </MyComponent>
</template>
```

Named Slots:

You can also define multiple slots with names, allowing more control over where content is placed.

1. **Defining Named Slots:**

```vue
<!-- MyComponent.vue -->
<template>
  <div>
    <h1>Welcome!</h1>
    <slot name="header"></slot>
    <slot></slot> <!-- This is a default slot -->
  </div>
</template>
```

2. **Using Named Slots in Parent Component:**

```vue
<!-- ParentComponent.vue -->
<template>
  <MyComponent>
    <template #header>
      <h2>This content will go into the header slot</h2>
    </template>
    <p>This content will go into the default slot</p>
  </MyComponent>
</template>
```

Passing Data via Slots:

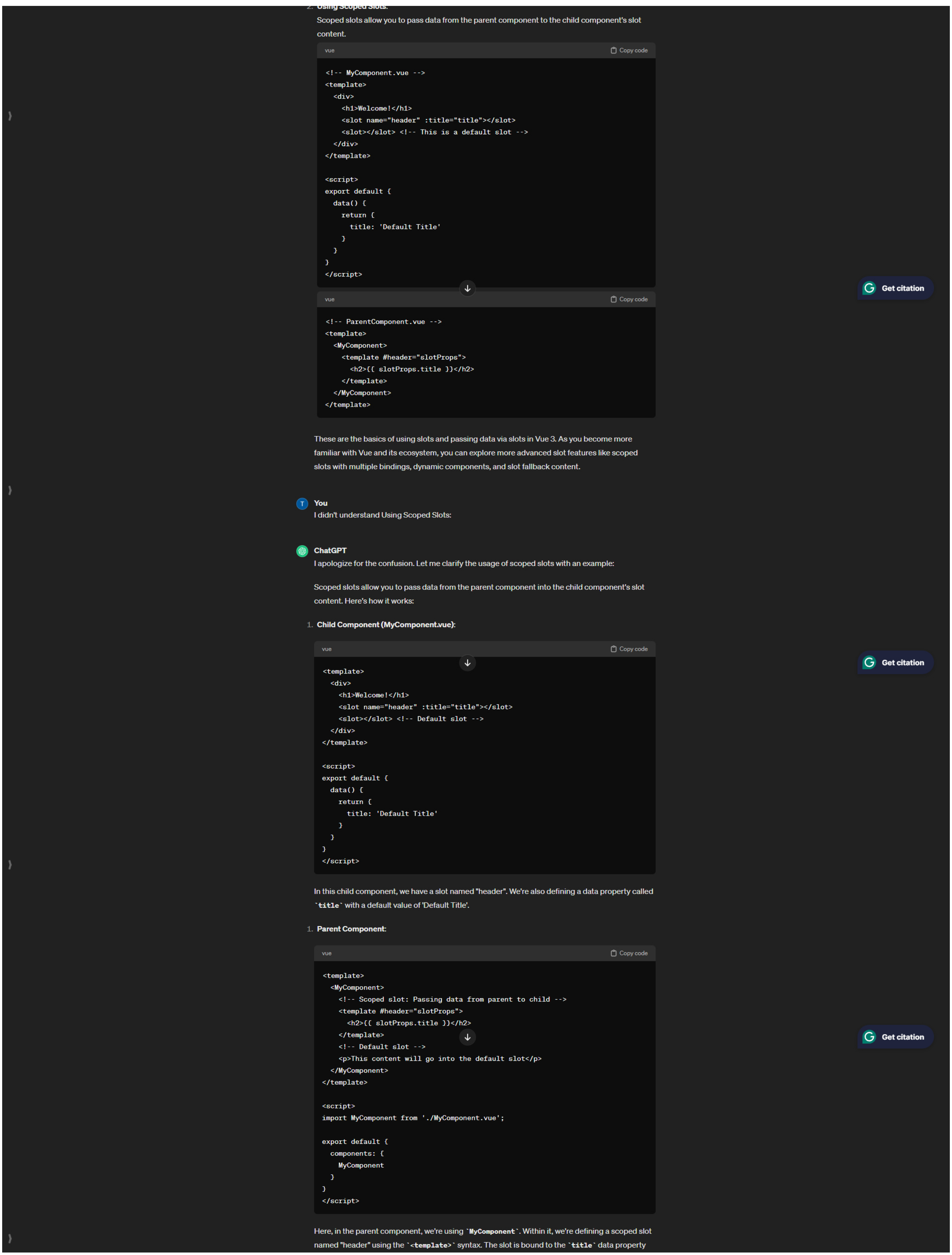
To pass data into the slots from the parent component, you can use props or scoped slots.

1. **Using Props:**
   You can pass data as props to the child component, and then use that data within the slot content.

```vue
<!-- ParentComponent.vue -->
<template>
  <MyComponent :title="parentTitle">
    <template #header="{ title }">
      <h2>{{ title }}</h2>
    </template>
  </MyComponent>
</template>

<script>
import MyComponent from './MyComponent.vue';

export default {
  components: {
    MyComponent
  },
  data() {
    return {
      parentTitle: 'Parent Title'
    }
  }
}
</script>
```



2. **Using Scoped Slots:**

Scoped slots allow you to pass data from the parent component to the child component's slot content.

```vue
<!-- MyComponent.vue -->
<template>
  <div>
    <h1>Welcome!</h1>
    <slot name="header" :title="title"></slot>
    <slot></slot> <!-- This is a default slot -->
  </div>
</template>

<script>
export default {
  data() {
    return {
      title: 'Default Title'
    }
  }
}
</script>
```

```vue
<!-- ParentComponent.vue -->
<template>
  <MyComponent>
    <template #header="slotProps">
      <h2>{{ slotProps.title }}</h2>
    </template>
  </MyComponent>
</template>
```

These are the basics of using slots and passing data via slots in Vue 3. As you become more familiar with Vue and its ecosystem, you can explore more advanced slot features like scoped slots with multiple bindings, dynamic components, and slot fallback content.

You

I didn't understand Using Scoped Slots:

ChatGPT

I apologize for the confusion. Let me clarify the usage of scoped slots with an example:

Scoped slots allow you to pass data from the parent component into the child component's slot content. Here's how it works:

1. **Child Component (MyComponent.vue):**

```vue
<template>
  <div>
    <h1>Welcome!</h1>
    <slot name="header" :title="title"></slot>
    <slot></slot> <!-- Default slot -->
  </div>
</template>

<script>
export default {
  data() {
    return {
      title: 'Default Title'
    }
  }
}
</script>
```

In this child component, we have a slot named "header". We're also defining a data property called `title` with a default value of 'Default Title'.

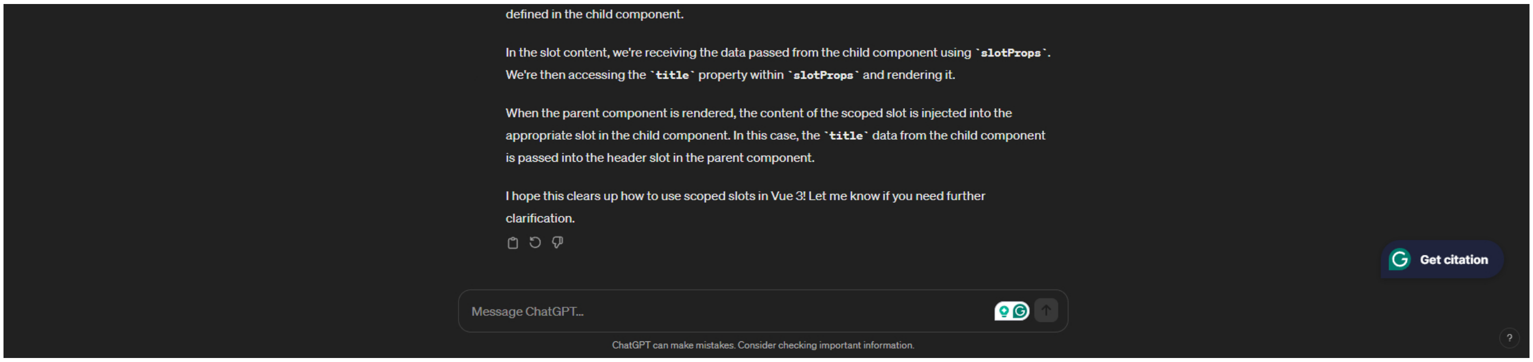
1. **Parent Component:**

```vue
<template>
  <MyComponent>
    <!-- Scoped slot: Passing data from parent to child -->
    <template #header="slotProps">
      <h2>{{ slotProps.title }}</h2>
    </template>
    <!-- Default slot -->
    <p>This content will go into the default slot</p>
  </MyComponent>
</template>

<script>
import MyComponent from './MyComponent.vue';

export default {
  components: {
    MyComponent
  }
}
</script>
```

Here, in the parent component, we're using `MyComponent`. Within it, we're defining a scoped slot named "header" using the `<template>` syntax. The slot is bound to the `title` data property



defined in the child component.

In the slot content, we're receiving the data passed from the child component using `slotProps`. We're then accessing the `title` property within `slotProps` and rendering it.

When the parent component is rendered, the content of the scoped slot is injected into the appropriate slot in the child component. In this case, the `title` data from the child component is passed into the header slot in the parent component.

I hope this clears up how to use scoped slots in Vue 3! Let me know if you need further clarification.

ChatGPT can make mistakes. Consider checking important information.