

1. Vue 組件結構：自訂屬性，例如：在根組設定屬性 title，就可在子組件 帶入 title

組件的使用，自訂屬性

App.vue

```
<script setup>
  import Nav from './Nav.vue';
  import Main from './Main.vue';
</script>
<template>
  <Nav title="任意的標題"></Nav>
  <Main></Main>
</template>
<style scoped>
</style>
```

Nav.vue

```
<script setup>
  defineProps(['title']);
</script>
<template>
  <nav>
    {{ title }}
  </nav>
</template>
<style scoped>
  nav{font-weight:bold}
</style>
```

Main.vue

暫時忽略 Main 組件的細節...

2:01 / 18:21

2. 細節語法：設定組件的自訂屬性，就像是 HTML 標籤的屬性一樣：

設定組件的自訂屬性

就像是 HTML 標籤的屬性一樣

- 載入組件
import 組件名稱 from "組件"
- 在樣板中使用載入的組件
<組件名稱
屬性名稱="資料"屬
></組件名稱>
- 基本範例

```
<script setup>
  import NavComp from './Nav.vue';
</script>
<template>
  <NavComp title="任意的標題"></NavComp>
</template>
```

4:49 / 18:21

3. 設定標籤之後，在子組件內部就要定義接收並使用屬性：用defineProps 這個函式語法：



The video player displays a slide with the title "定義、接收並使用屬性" (Define, Receive, and Use Properties). Below the title, it says "defineProps 定義要接收的屬性名稱" (defineProps defines the property names to be received). To the right of the video player, there is a list of bullet points and a code example.

- 定義要接收的屬性名稱語法
`defineProps(["屬性名稱", "屬性名稱"])`
- 基本範例

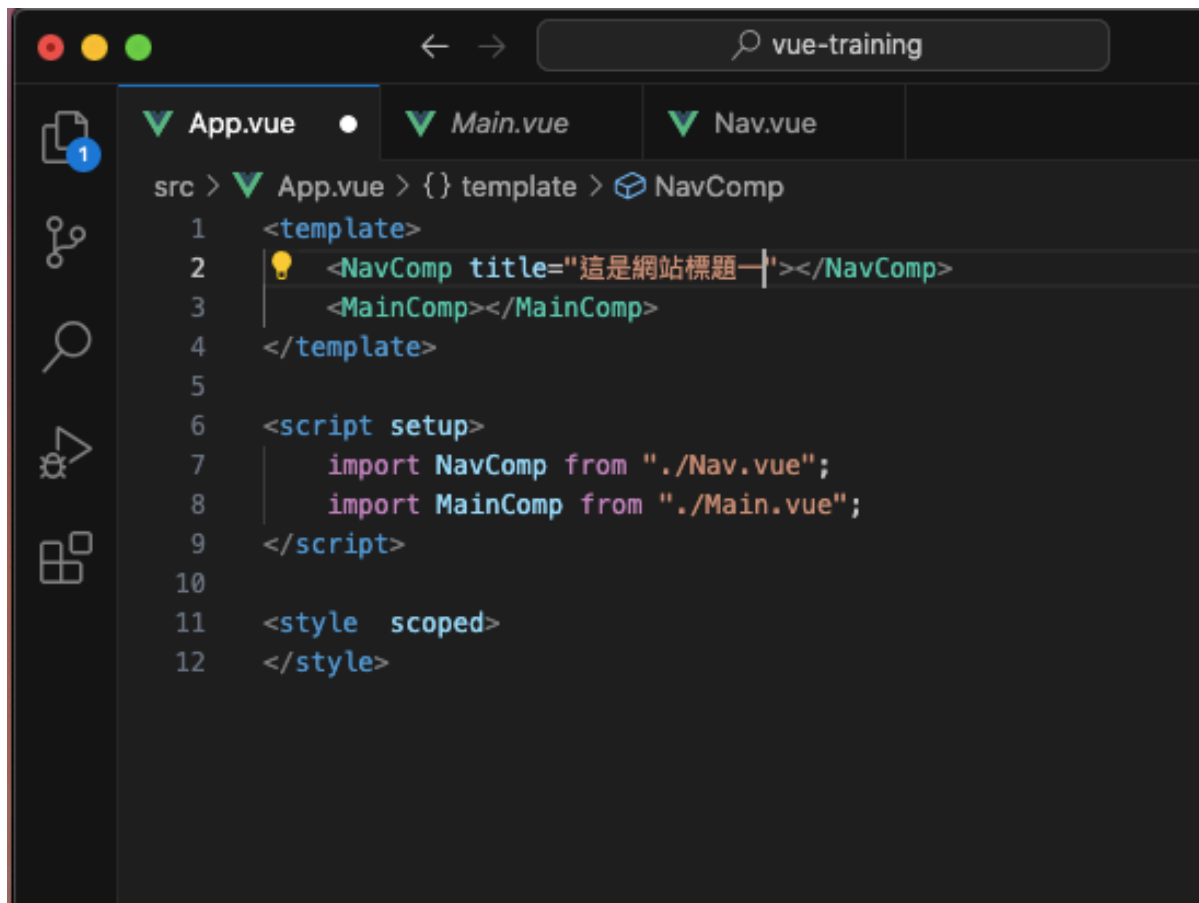
```
<script setup>
  import NavComp from './NavComp.vue'
</script>
<template>
  <NavComp title="標題" />
</template>
```

An arrow points from the `title="標題"` attribute in the first code block to the `defineProps(["title"])` line in the second code block.

```
<script setup>
  defineProps(["title"])
</script>
<template>
  <nav>[[ title ]]</nav>
</template>
```

4. 寫程式碼：在子組件Nav.vue，呼叫函式 `defineProps(["title"])`，接收上層的屬性值：

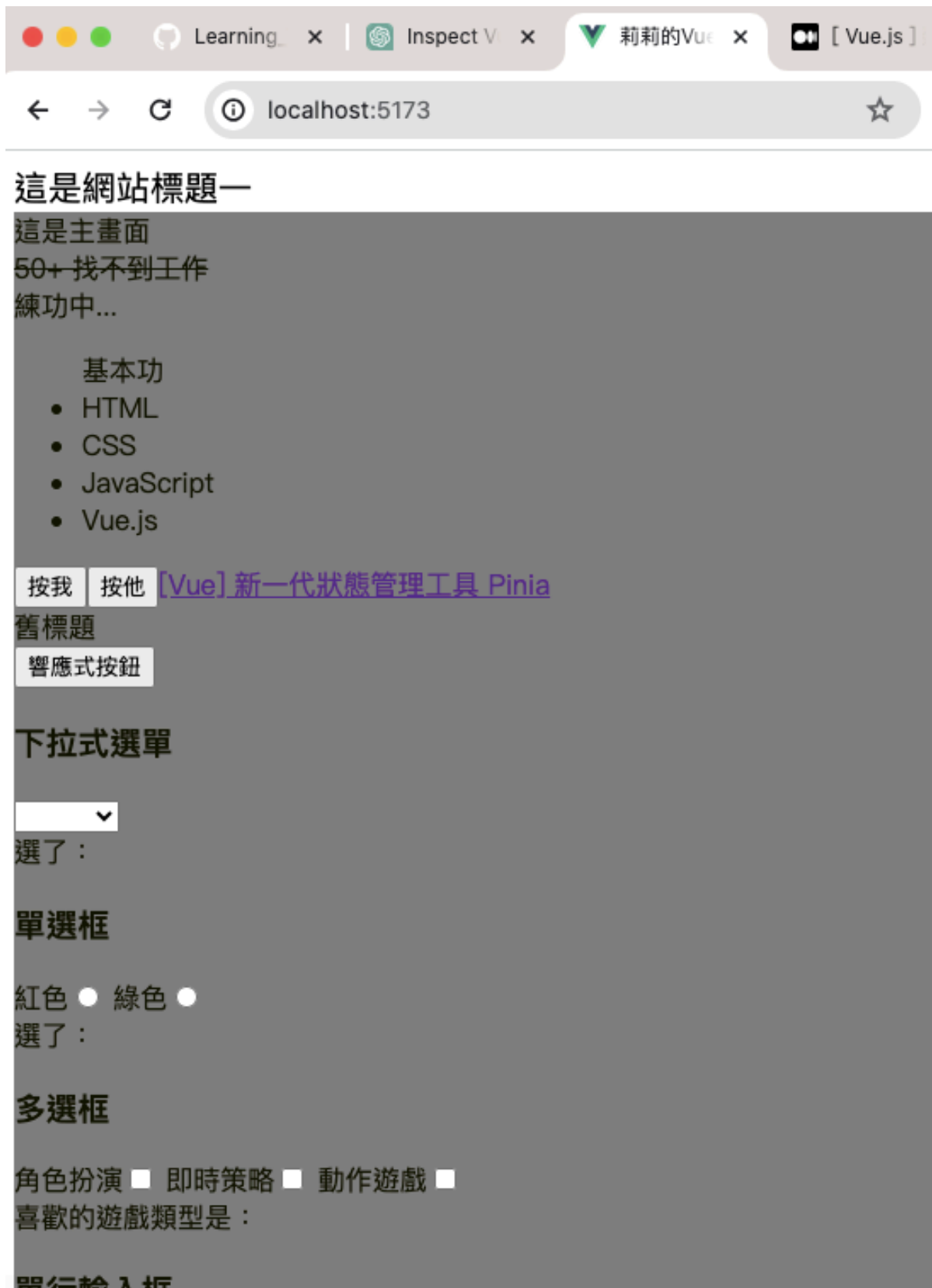
然後在 上層組件 App.vue , 的 NavComp 標題 , 自訂屬性 title=“這是網站標題一” :



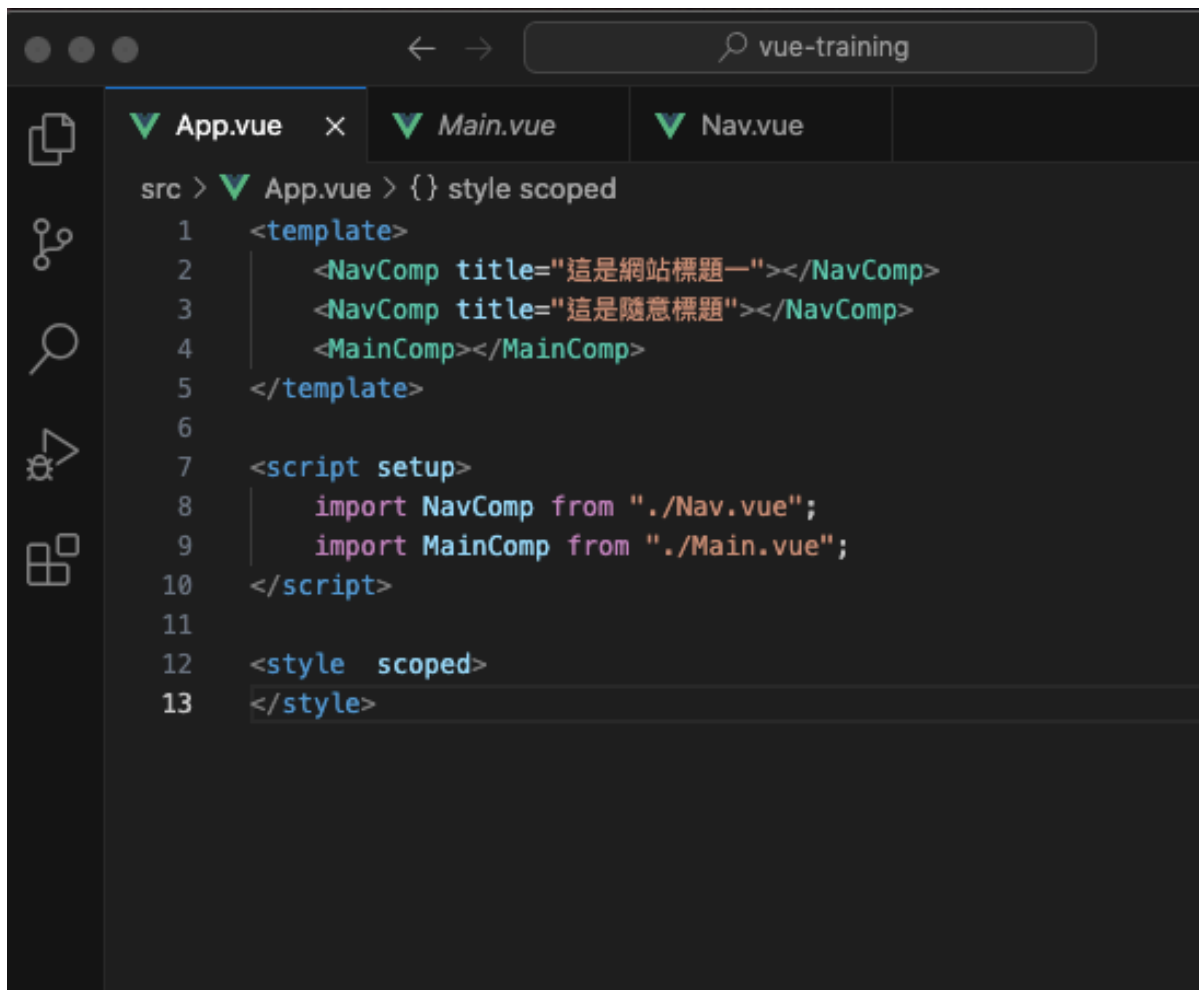
The screenshot shows a code editor window with a dark theme. At the top, there's a search bar containing 'vue-training'. Below it, a tab bar shows three files: 'App.vue' (selected), 'Main.vue', and 'Nav.vue'. The left sidebar contains icons for file explorer, source control, search, and a component inspector. The main editor area displays the code for 'App.vue' with the following structure:

```
src > App.vue > {} template > NavComp
1  <template>
2  |  <NavComp title="這是網站標題一"></NavComp>
3  |  <MainComp></MainComp>
4  </template>
5
6  <script setup>
7  |    import NavComp from "./Nav.vue";
8  |    import MainComp from "./Main.vue";
9  </script>
10
11 <style scoped>
12 </style>
```

看結果

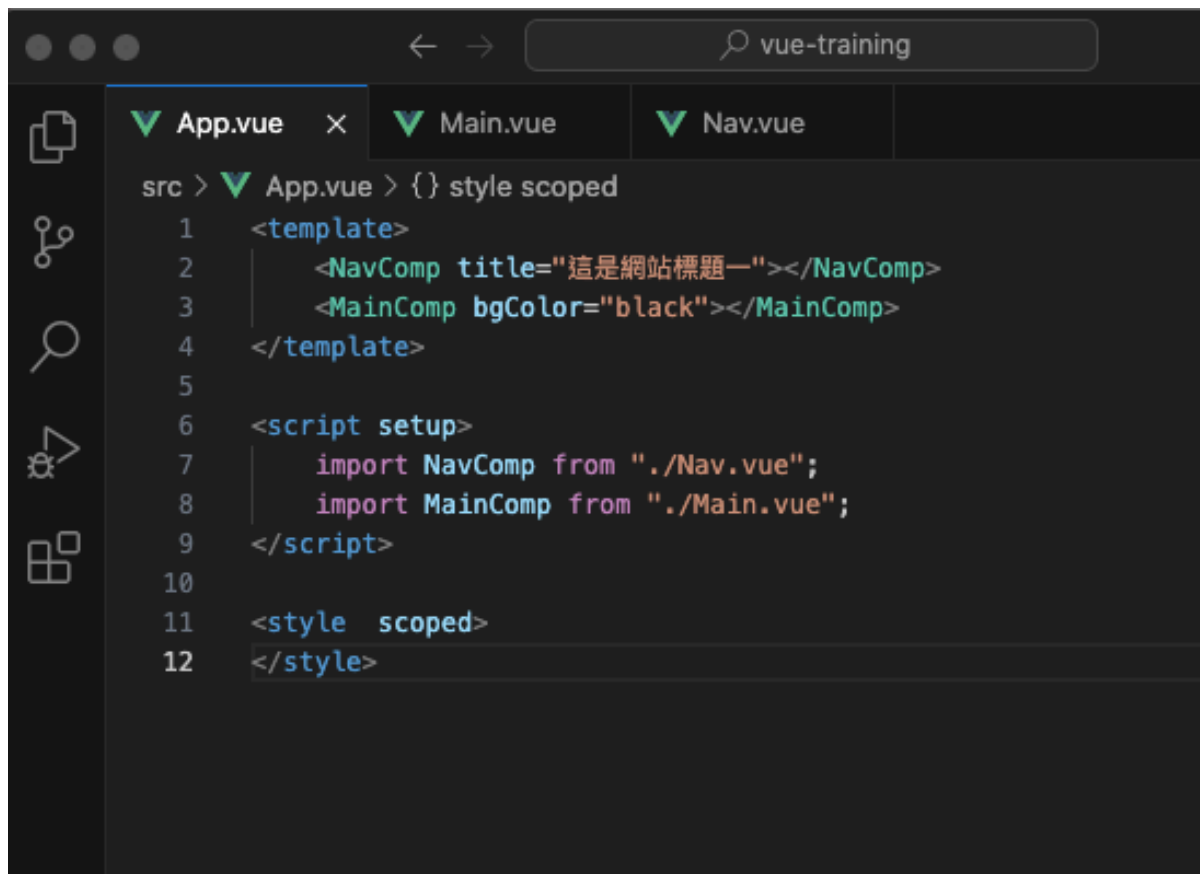


改變 NavComp 的標籤的值，可以很有彈性

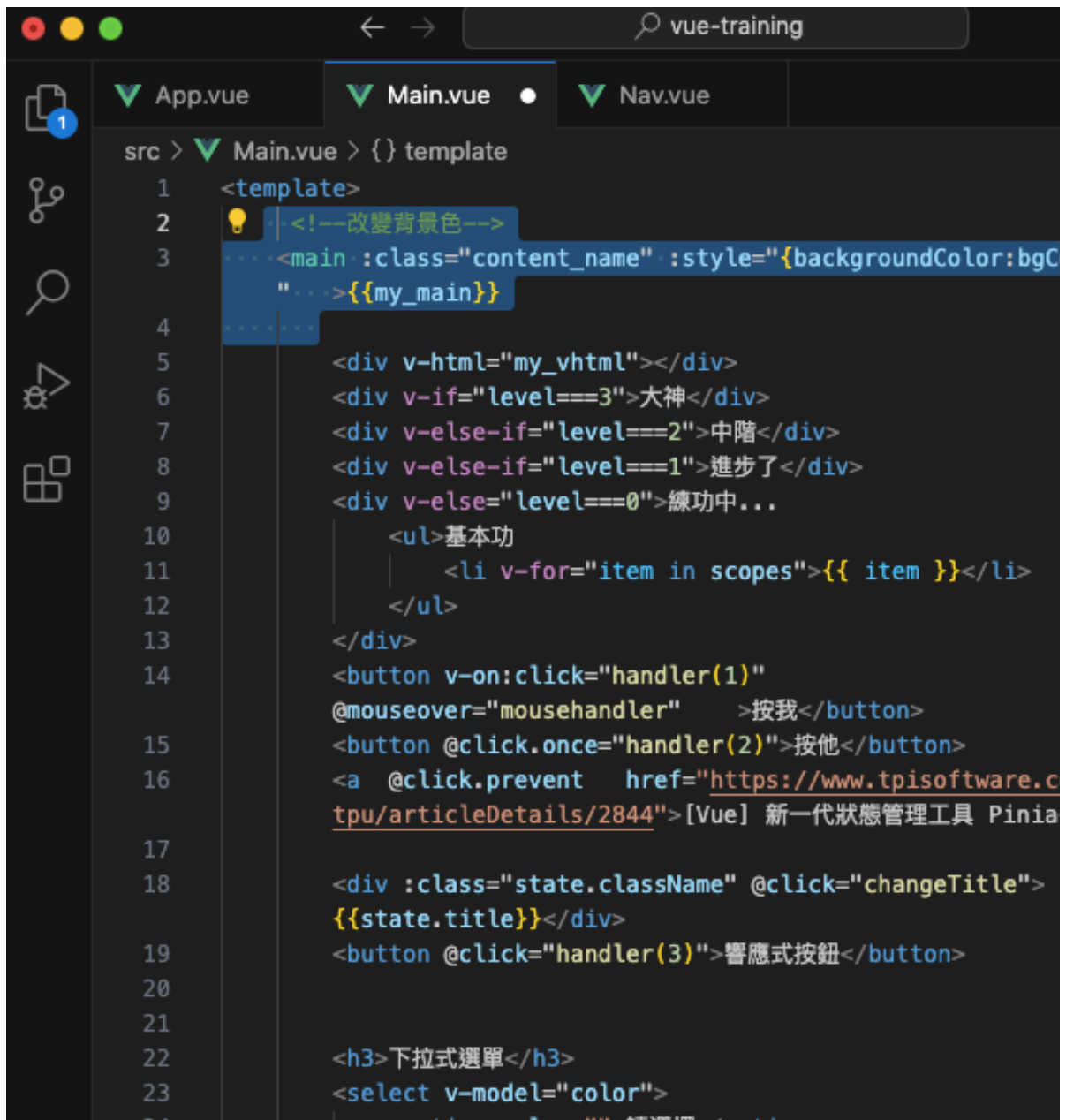


```
src > App.vue > {} style scoped
1  <template>
2    <NavComp title="這是網站標題一"></NavComp>
3    <NavComp title="這是隨意標題"></NavComp>
4    <MainComp></MainComp>
5  </template>
6
7  <script setup>
8    import NavComp from "./Nav.vue";
9    import MainComp from "./Main.vue";
10 </script>
11
12 <style scoped>
13 </style>
```

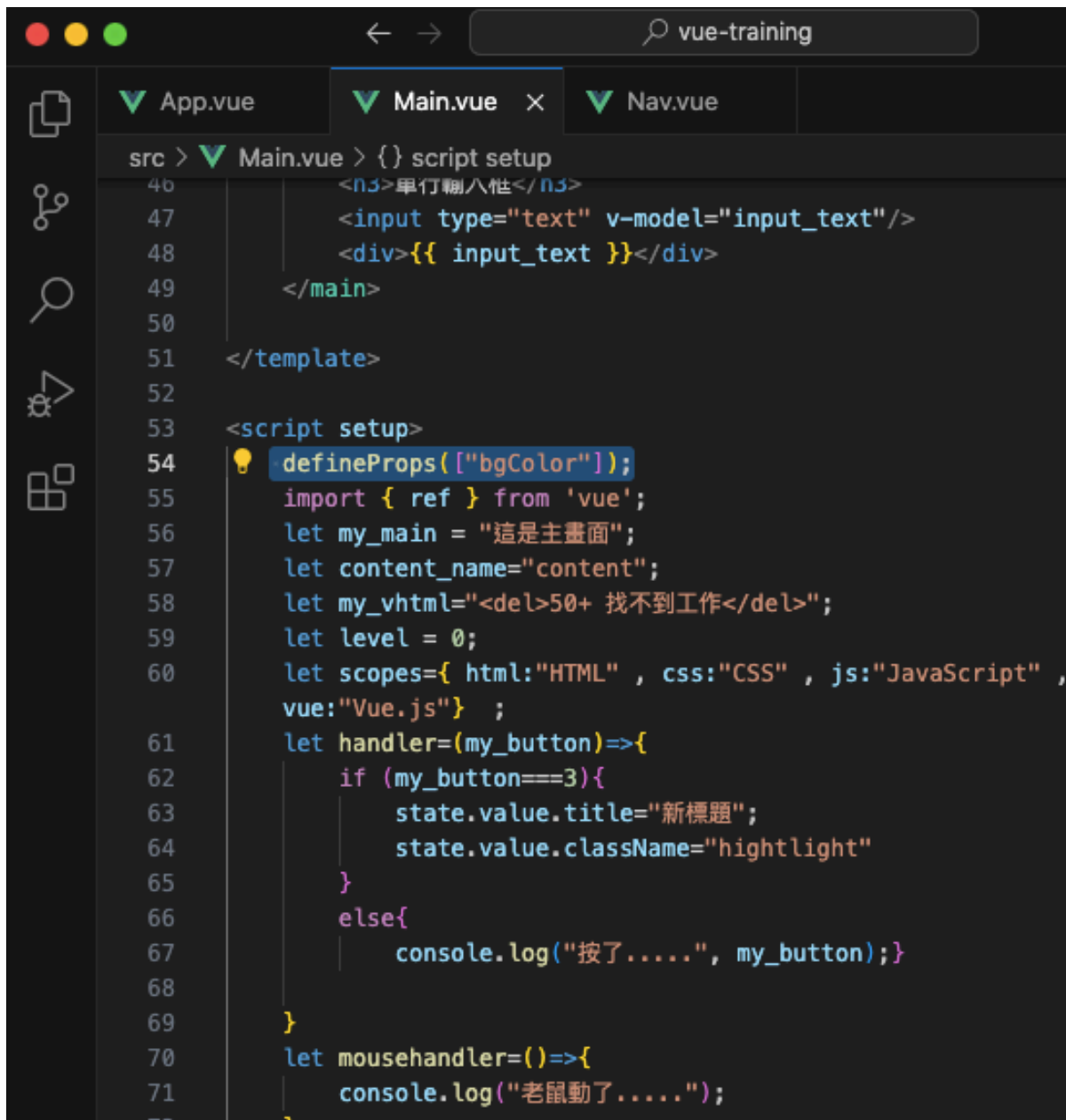
也可以改變 CSS 屬性值：例如 設定main的 背景色為 黑色:



```
src > ▼ App.vue > {} style scoped
1  <template>
2    <NavComp title="這是網站標題一"></NavComp>
3    <MainComp bgColor="black"></MainComp>
4  </template>
5
6  <script setup>
7    import NavComp from "./Nav.vue";
8    import MainComp from "./Main.vue";
9  </script>
10
11 <style scoped>
12 </style>
```

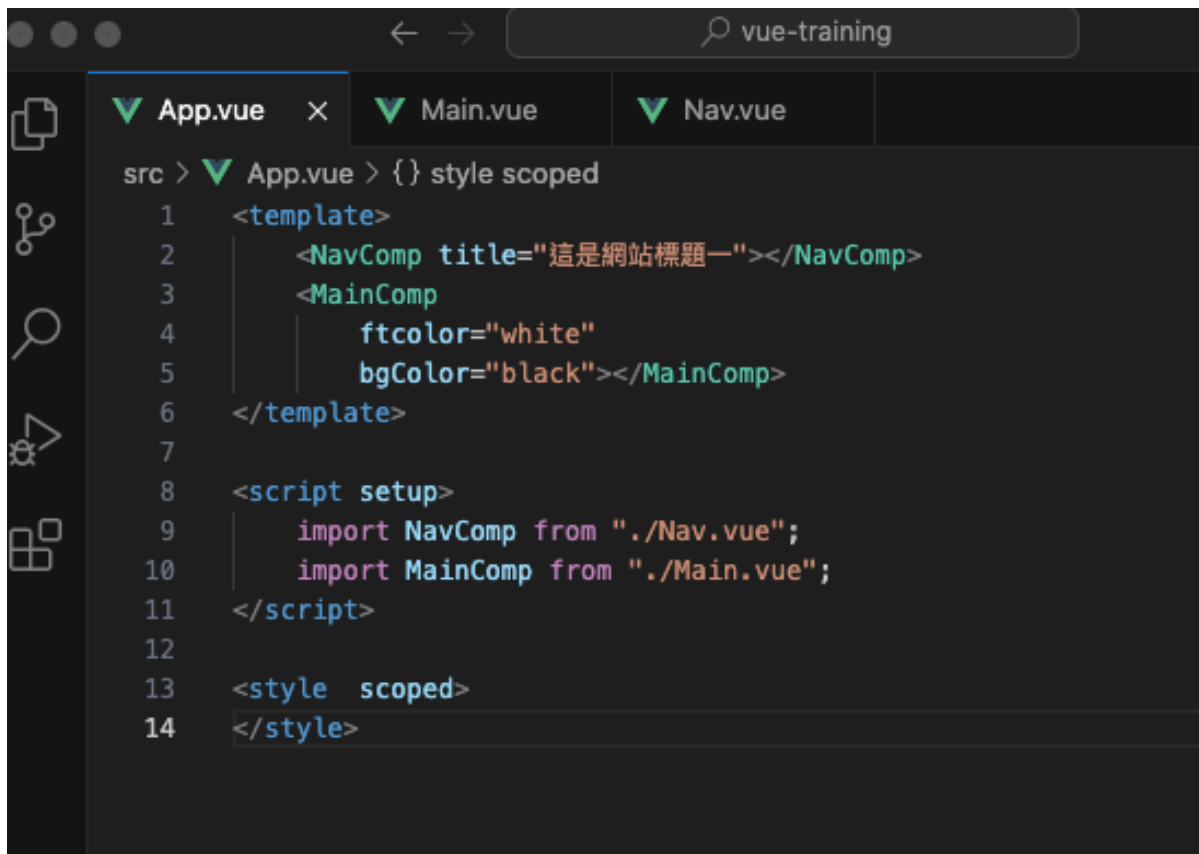


```
src > App.vue Main.vue Nav.vue
1 <template>
2   <!--改變背景色-->
3   <main :class="content_name" :style="{backgroundColor:bgC
4   " :>{{my_main}}
5
6   <div v-html="my_vhtml"></div>
7   <div v-if="level===3">大神</div>
8   <div v-else-if="level===2">中階</div>
9   <div v-else-if="level===1">進步了</div>
10  <div v-else="level===0">練功中...
11    <ul>基本功
12      <li v-for="item in scopes">{{ item }}</li>
13    </ul>
14  </div>
15  <button v-on:click="handler(1)"
16    @mouseover="mousehandler">按我</button>
17  <button @click.once="handler(2)">按他</button>
18  <a @click.prevent href="https://www.tpisoftware.c
19    tpu/articleDetails/2844">[Vue] 新一代狀態管理工具 Pinia
20
21  <div :class="state.className" @click="changeTitle">
22    {{state.title}}</div>
23  <button @click="handler(3)">響應式按鈕</button>
24
25  <h3>下拉式選單</h3>
26  <select v-model="color">
27    <option value="">請選擇</option>
```

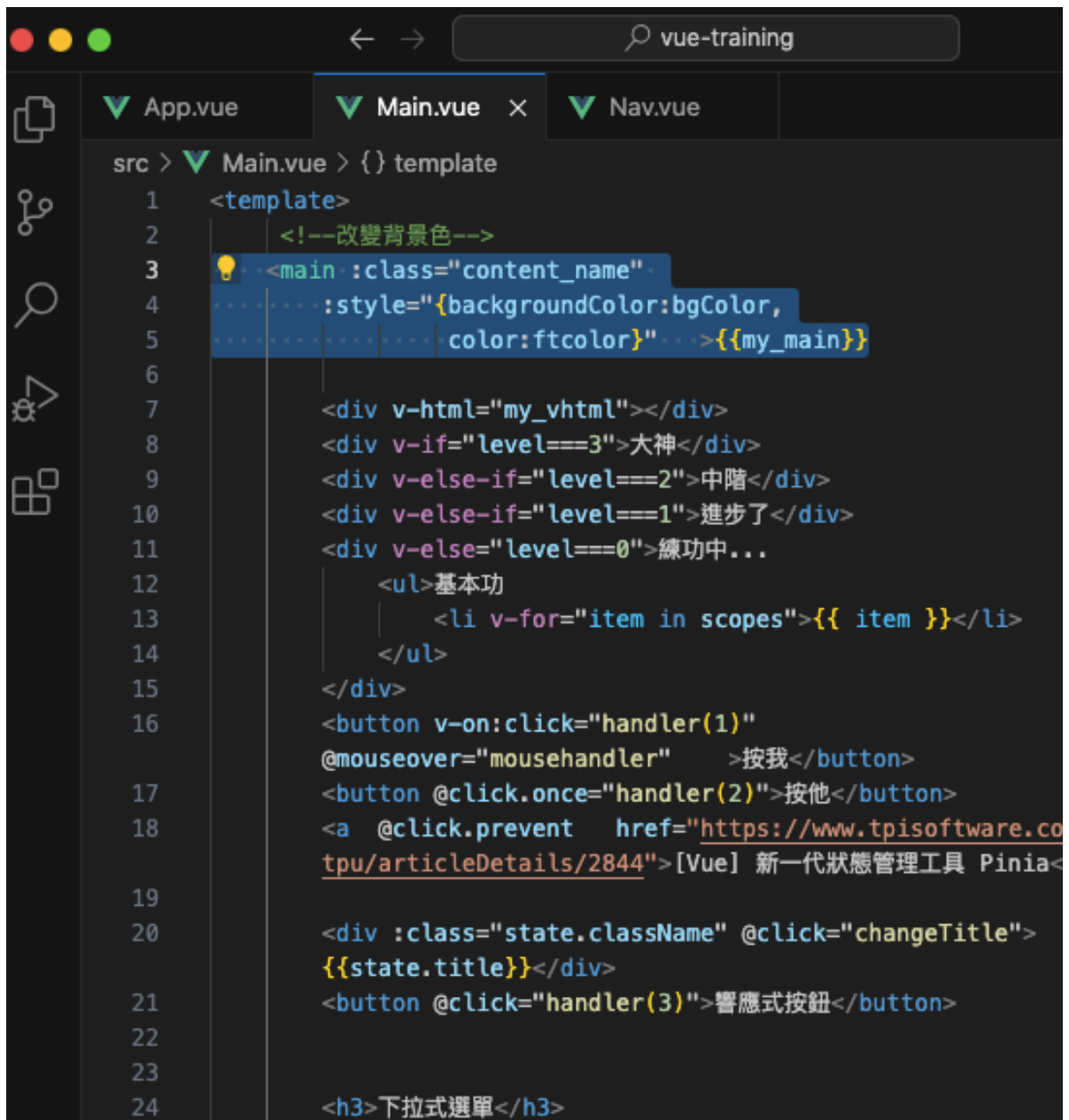
```
src > App.vue Main.vue X Nav.vue
src > Main.vue > {} script setup
46      <n3>單行輸入框</n3>
47      <input type="text" v-model="input_text"/>
48      <div>{{ input_text }}</div>
49    </main>
50
51  </template>
52
53  <script setup>
54    defineProps(["bgColor"]);
55    import { ref } from 'vue';
56    let my_main = "這是主畫面";
57    let content_name="content";
58    let my_vhtml="<del>50+ 找不到工作</del>";
59    let level = 0;
60    let scopes={ html:"HTML" , css:"CSS" , js:"JavaScript" ,
vue:"Vue.js"} ;
61    let handler=(my_button)=>{
62      if (my_button===3){
63        state.value.title="新標題";
64        state.value.className="hightlight"
65      }
66      else{
67        console.log("按了.....", my_button);}
68    }
69    let mousehandler=()=>{
70      console.log("老鼠動了.....");
71    }
```

也可以多加一個 color 的 CSS 屬性：

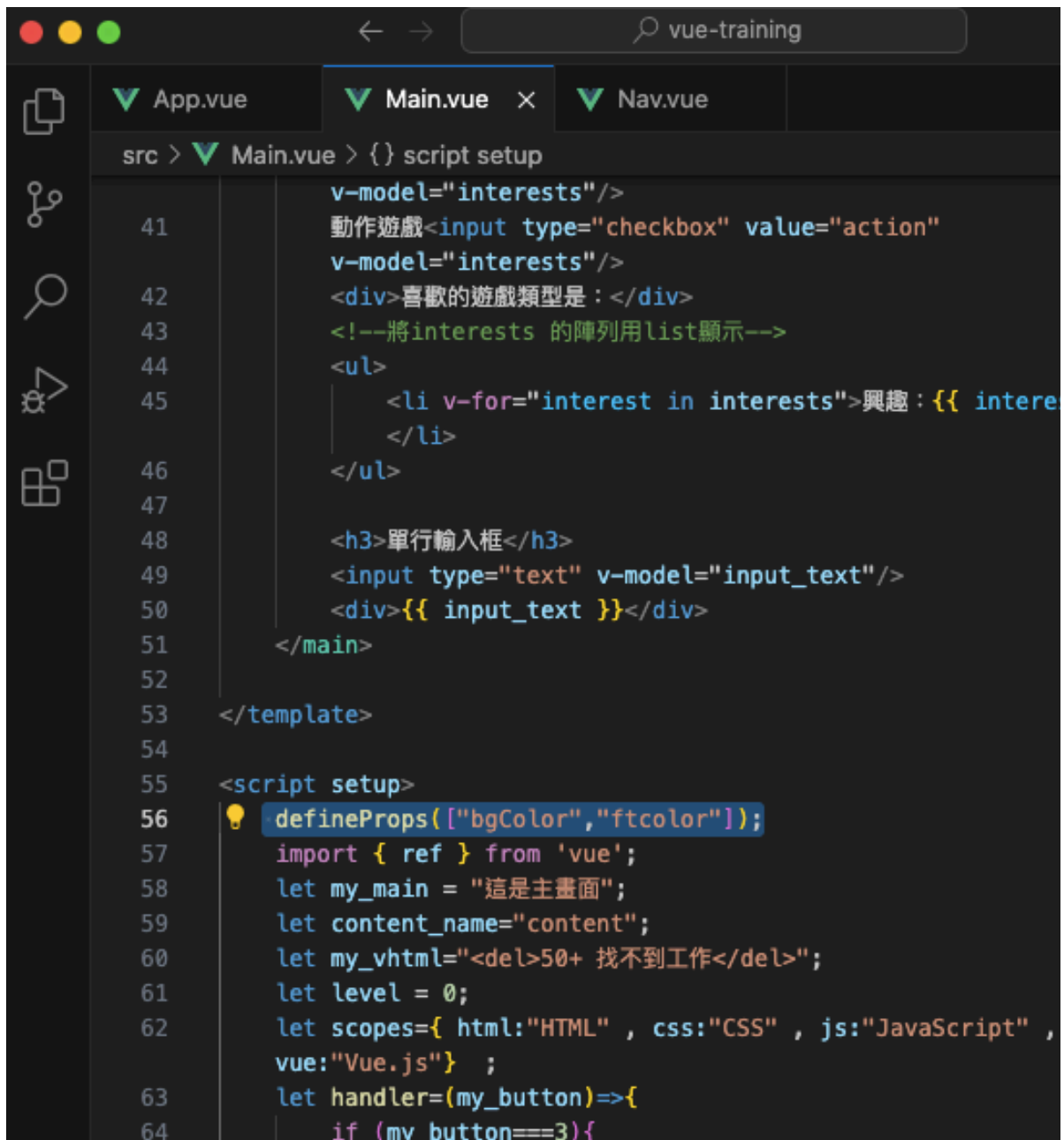


The image shows a code editor window with a dark theme. At the top, there is a search bar containing the text "vue-training". Below the search bar, there are three tabs: "App.vue" (selected), "Main.vue", and "Nav.vue". The "App.vue" tab is active, and its content is displayed in the main editor area. The code is a Vue component with a template, script setup, and style scoped sections. The template section contains two child components: "NavComp" and "MainComp". The "NavComp" component has a title attribute set to "這是網站標題一". The "MainComp" component has two attributes: "ftcolor" set to "white" and "bgColor" set to "black". The script setup section imports "NavComp" from "./Nav.vue" and "MainComp" from "./Main.vue". The style scoped section is empty.

```
src > ▼ App.vue > {} style scoped
1  <template>
2    <NavComp title="這是網站標題一"></NavComp>
3    <MainComp
4      ftcolor="white"
5      bgColor="black"></MainComp>
6  </template>
7
8  <script setup>
9    import NavComp from "./Nav.vue";
10   import MainComp from "./Main.vue";
11 </script>
12
13 <style scoped>
14 </style>
```



```
src > App.vue Main.vue x Nav.vue
src > Main.vue > {} template
1 <template>
2   <!--改變背景色-->
3   <main :class="content_name"
4     :style="{backgroundColor:bgColor,
5     color:ftcolor}">{{my_main}}
6
7   <div v-html="my_vhtml"></div>
8   <div v-if="level===3">大神</div>
9   <div v-else-if="level===2">中階</div>
10  <div v-else-if="level===1">進步了</div>
11  <div v-else="level===0">練功中...
12    <ul>基本功
13      <li v-for="item in scopes">{{ item }}</li>
14    </ul>
15  </div>
16  <button v-on:click="handler(1)"
17    @mouseover="mousehandler" >按我</button>
18  <button @click.once="handler(2)">按他</button>
19  <a @click.prevent href="https://www.tpisoftware.com/tpu/articleDetails/2844">[Vue] 新一代狀態管理工具 Pinia<
20
21  <div :class="state.className" @click="changeTitle">
22    {{state.title}}</div>
23
24  <button @click="handler(3)">響應式按鈕</button>
25
26  <h3>下拉式選單</h3>
```



```
vue-training

App.vue Main.vue x Nav.vue
src > Main.vue > {} script setup

41 v-model="interests"/>
    動作遊戲
```

看結果：

這是網站標題一

這是主畫面

50+ 找不到工作

練功中...

基本功

- HTML
- CSS
- JavaScript
- Vue.js

按我 按他 [\[Vue\] 新一代狀態管理工具 Pinia](#)

舊標題

響應式按鈕

下拉式選單

選了：

單選框

紅色 ☐ 綠色 ☐

選了：

多選框

角色扮演 ☐ 即時策略 ☐ 動作遊戲 ☐

喜歡的遊戲類型是：

單行輸入框