

# Vue Router

<https://router.vuejs.org/>

## ***Встановлення модуля***

<b><u>Якщо ще не встановлено,</u> то встановлюємо модуль</b>	<b><code>npm install vue-router@4</code> або <code>yarn add vue-router@4</code></b>
--	---

<https://vue-router-ru.netlify.app/installation.html>

### Загальна схема

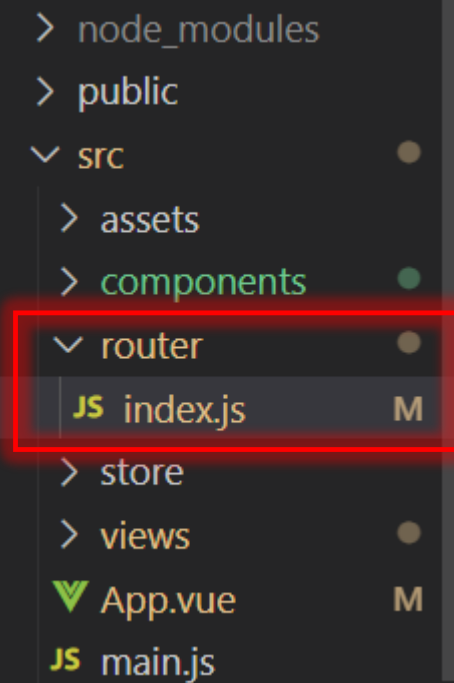
```
//0.Імпортуємо модуль та його компоненти  
import { createRouter, createWebHashHistory  
} from 'vue-router'
```

### Приклад

```
//0.Імпортуємо модуль та його компоненти  
import { createRouter, createWebHashHistory }  
from 'vue-router'
```

### Опис файлу налаштувань

0)імпорт необхідних функцій



## Загальна схема

```
//0.Імпортуємо модуль та його компоненти
import { createRouter, createWebHashHistory
} from 'vue-router'
```

```
// 1. Import route components.
. . . . .
імпортуємо компоненти
```

## Приклад

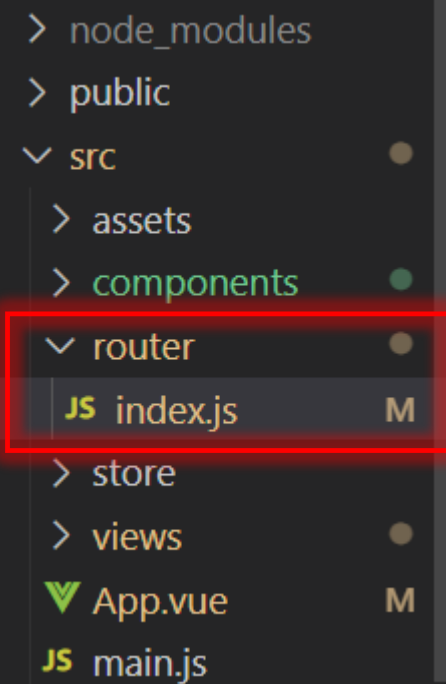
```
//0.Імпортуємо модуль та його компоненти
import { createRouter, createWebHashHistory }
from 'vue-router'
```

```
// 1. Import route components.
import Home from './pages/Home'
import Products from '@pages/Products'
```

Опис файлу налаштувань

0)імпорт необхідних функцій

1)імпорт компонентів, які  
прв'язуємо до маршрутів



## Загальна схема

## Приклад

### Опис файлу налаштувань

0) імпорт необхідних функцій

1) імпорт компонентів, які прив'язуємо до маршрутів

2) встановлюємо відповідність між маршрутами і компонентами

```
//0.Імпортуємо модуль та його компоненти
import { createRouter, createWebHashHistory } from 'vue-router'
```

```
// 1. Import route components.
```

```
.....
```

```
імпортуємо компоненти
```

```
.....
```

```
// 2. Задаємо роути
```

```
// Each route should map to a component.
```

```
const routes = [
```

```
{
  path: 'шлях',
  component: компонент-відповідний,
  name: ' ім'я_маршрута ',
},
```

```
{
  path: 'шлях / : параметр',
  component: компонент-відповідний,
  name: ' ім'я_маршрута ',
},
```

```
]
```

```
//0.Імпортуємо модуль та його компоненти
import { createRouter, createWebHashHistory } from 'vue-router'
```

```
// 1. Import route components.
```

```
import Home from './pages/Home'
```

```
import Products from '@pages/Products'
```

```
// 2. Задаємо роути
```

```
// Each route should map to a component.
```

```
const routes = [
```

```
{
  path: '/',
  component: Home,
  name: 'home',
},
```

```
// Приклад динамічного роута з параметром "id"
```

```
{
  path: '/products-list/:id?',
  component: Products,
  name: 'products'
},
```

```
]
```

> node\_modules

> public

▼ src

> assets

> components

▼ router

JS index.js

M

> store

> views

▼ App.vue

M

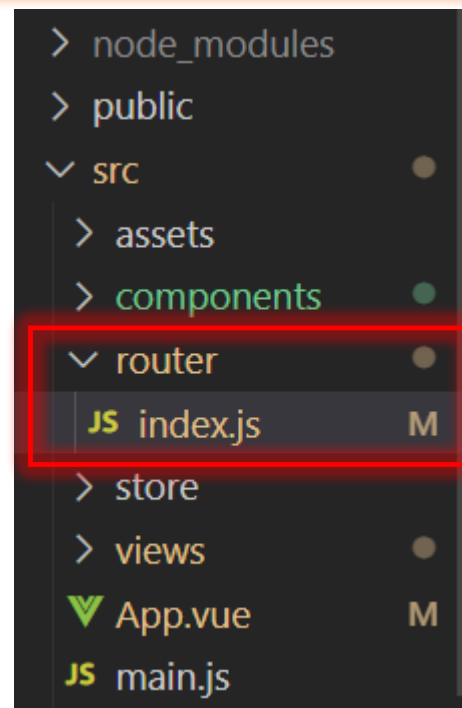
JS main.js

## Загальна схема

## Приклад

### Опис файлу налаштувань

- 0) імпорт необхідних функцій
- 1) імпорт компонентів, які прив'язуємо до маршрутів
- 2) встановлюємо відповідність між маршрутами і компонентами
- 3) створюємо об'єкт роутера



```
//0.Імпортуємо модуль та його компоненти
import { createRouter, createWebHashHistory
} from 'vue-router'
```

```
// 1. Import route components.
```

```
.....
```

```
імпортуємо компоненти
```

```
.....
```

```
// 2. Задаємо роути
```

```
// Each route should map to a component.
```

```
const routes = [
```

```
{
  path: 'шлях',
  component: компонент-відповідний,
  name: ' ім'я_маршрута ',
```

```
},
```

```
{
  path: 'шлях / : параметр',
  component: компонент-відповідний,
  name: ' ім'я_маршрута ',
```

```
},
```

```
]
```

```
// 3. Створюємо об'єкт роутера
```

```
const router = createRouter({
  history: createWebHashHistory(),
  routes,
```

```
})
```

```
export default router
```

```
//0.Імпортуємо модуль та його компоненти
```

```
import { createRouter, createWebHashHistory }
from 'vue-router'
```

```
// 1. Import route components.
```

```
import Home from './pages/Home'
```

```
import Products from '@pages/Products'
```

```
// 2. Задаємо роути
```

```
// Each route should map to a component.
```

```
const routes = [
```

```
{
  path: '/',
  component: Home,
  name: 'home',
```

```
},
```

```
// Приклад динамічного роута з параметером "id"
```

```
{
  path: '/products-list/:id?',
  component: Products,
  name: 'products'
```

```
},
```

```
]
```

```
// 3. Створюємо об'єкт роутера
```

```
const router = createRouter({
  history: createWebHashHistory(),
  routes, // short for `routes: routes`
})
```

```
export default router
```

# Підключення роутера

EXPLORER ...

> OPEN EDITORS

ROUTER-VUEX-TEMPLATE

- > .history
- > node\_modules
- > public
- src
  - > assets
  - > components
  - router
    - JS index.js
  - > store
  - > views
  - App.vue
  - JS main.js**

src > JS main.js

You, 7 hours ago | 1 author (You)

```
1 import { createApp } from "vue";
2 import App from "./App.vue";
3 import router from "./router";
4 import store from "./store";
5
6 createApp(App).use(store).use(router).mount("#app");
7
```

1.Імпортуємо модуль

2.Підключаємо як модуль

**Вказуємо позицію, де потрібно відображати компоненти, які відображать поточному маршруту. <router-view>**

//--- місце монтування компонента, який відповідає шляху

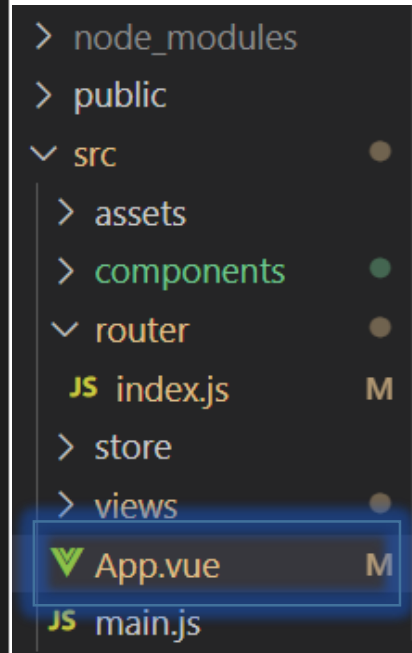
```
<router-view></router-view>
```

```
<template>
  <div>
```

```

    <!-- route outlet, component matched by the route
will render here -->
    <router-view></router-view>
</div>
</template>

```





localhost:8080

*Користувач переходить за URL*

Як це працює?

Як це працює?

localhost:8080

```
> router > JS index.js > routes
You, 1 second ago | 1 author (You)
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3
4 const routes = [
5   {
6     path: '/',
7     name: 'home',
8     component: HomeView,
9   },
10  {
11    You, 7 seconds ago • Uncommitted changes ...
12  },
13 ]
```

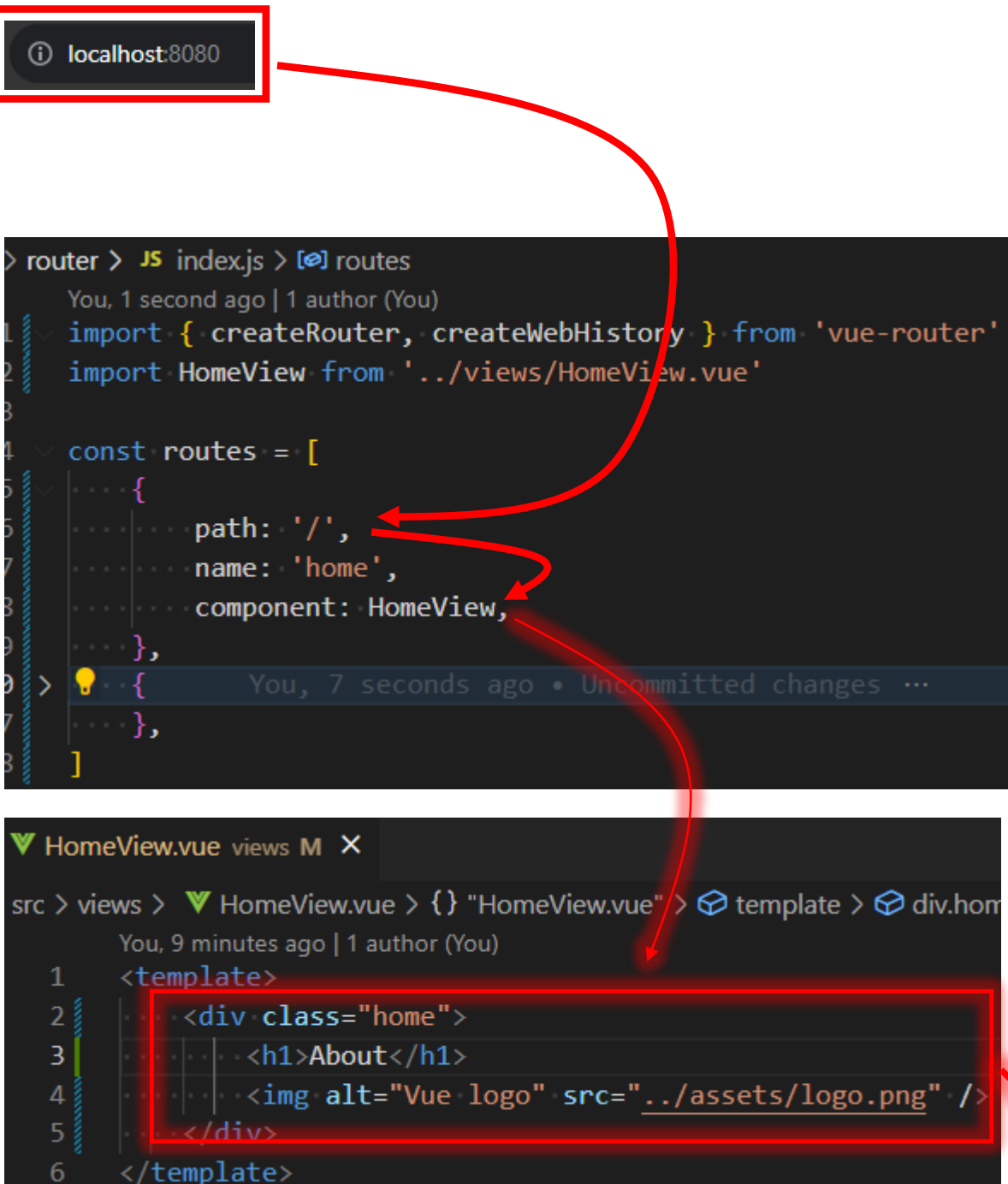
*Знаходимо  
відповідність  
маршруту*

Як це працює?

localhost:8080

```
> router > JS index.js > [e] routes
You, 1 second ago | 1 author (You)
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3
4 const routes = [
5   {
6     path: '/',
7     name: 'home',
8     component: HomeView,
9   },
10  {
11    You, 7 seconds ago • Uncommitted changes ...
12  },
13 ]
```

*Знаходимо  
компонент,  
який треба  
відобразити*



localhost:8080

```
> router > JS index.js > routes
You, 1 second ago | 1 author (You)
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3
4 const routes = [
5   {
6     path: '/',
7     name: 'home',
8     component: HomeView,
9   },
10 ]
```

Вміст компонента  
вставляється у позицію  
<router-view/>

```
▼ HomeView.vue views M X
src > views > ▼ HomeView.vue > {} "HomeView.vue" > template > div.home
You, 9 minutes ago | 1 author (You)
1 <template>
2   <div class="home">
3     <h1>About</h1>
4     
5   </div>
6 </template>
```

```
<template>
  <nav>
    <router-link to="/">
      Home
    </router-link> |
    <router-link to="/about">
      About
    </router-link>
  </nav>
  <router-view />
</template>
```

Як це працює?

localhost:8080

```
> router > JS index.js > routes
You, 1 second ago | 1 author (You)
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3
4 const routes = [
5   {
6     path: '/',
7     name: 'home',
8     component: HomeView,
9   },
10 ]
```

▼ HomeView.vue views M X

src > views > ▼ HomeView.vue > {} "HomeView.vue" > template > div.home

You, 9 minutes ago | 1 author (You)

```
1 <template>
2   <div class="home">
3     <h1>About</h1>
4     
5   </div>
6 </template>
```

```
<div id="app" data-v-app>
  <nav>
    <a href="/" class="router-link-active router-link-exact-active" aria-current="page">Home</a>
    " | "
    <a href="/about" class>About</a>
  </nav>
  <div class="home"> == $0
    <h1>About</h1>
    
  </div>
</div>
```

```
<template>
  <nav>
    <router-link to="/">
      Home
    </router-link> |
    <router-link to="/about">
      About
    </router-link>
  </nav>
  <router-view />
</template>
```

Як це працює?

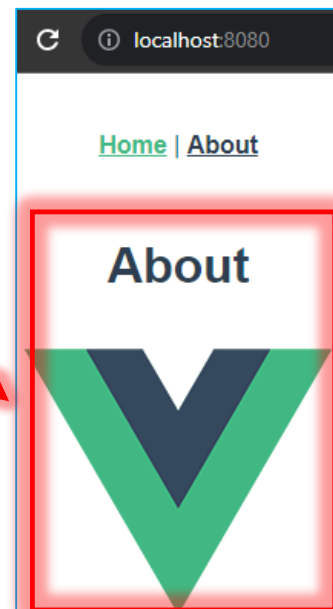
localhost:8080


```
> router > JS index.js > routes
You, 1 second ago | 1 author (You)
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3
4 const routes = [
5   {
6     path: '/',
7     name: 'home',
8     component: HomeView,
9   },
10 ]
```

```
▼ HomeView.vue views M X
src > views > ▼ HomeView.vue > {} "HomeView.vue" > template > div.home
You, 9 minutes ago | 1 author (You)
1 <template>
2   <div class="home">
3     <h1>About</h1>
4     
5   </div>
6 </template>
```

```
<div id="app" data-v-app>
  <nav>
    <a href="/" class="router-link-active router-link-exact-active" aria-current="page">Home</a>
    " | "
    <a href="/about" class>About</a>
  </nav>
  <div class="home"> == $0
    <h1>About</h1>
    
  </div>
</div>
```

```
<template>
  <nav>
    <router-link to="/">
      Home
    </router-link> |
    <router-link to="/about">
      About
    </router-link>
  </nav>
  <router-view />
</template>
```



 localhost:8080/about

*Користувач переходить за URL*

Як це працює?

Як це працює?

localhost8080/about

```
import { createRouter, createWebHistory } from 'vue-router'
import HomeView from '../views/HomeView.vue'
import AboutView from '../views/AboutView.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/about',
    name: 'about',
    component: AboutView,
  },
]
```

Знаходимо  
відповідність  
маршруту



Як це працює?

localhost:8080/about

```
import { createRouter, createWebHistory } from 'vue-router'
import HomeView from '../views/HomeView.vue'
import AboutView from '../views/AboutView.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/about',
    name: 'about',
    component: AboutView,
  },
]
```

*Знаходимо  
компонент,  
який треба  
відобразити*

localhost:8080/about

```
import { createRouter, createWebHistory } from 'vue-router'
import HomeView from '../views/HomeView.vue'
import AboutView from '../views/AboutView.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/about',
    name: 'about',
    component: AboutView,
  },
]
```

Вміст компонента  
вставляється у позицію  
<router-view/>

> views > AboutView.vue > template

```
1 <template>
2   <div class="about">
3     <h1>This is an about page</h1>
4   </div>
5 </template>
```

```
<template>
  <nav>
    <router-link to="/">
      Home
    </router-link> |
    <router-link to="/about">
      About
    </router-link>
  </nav>
  <router-view />
</template>
```

Як це працює?

localhost:8080/about

```
import { createRouter, createWebHistory } from 'vue-router'
import HomeView from '../views/HomeView.vue'
import AboutView from '../views/AboutView.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/about',
    name: 'about',
    component: AboutView,
  },
]
```

> views > AboutView.vue > template

```
1 <template>
2   <div class="about">
3     <h1>This is an about page</h1>
4   </div>
5 </template>
```

```
<div id="app" data-v-app>
  <nav>
    <a href="/" class> Home </a>
    |
    <a href="/about" class="router-link-active router-link-exact-active" aria-current="page"> About
  </nav>
  <div class="about"> == $0
    <h1>This is an about page</h1>
  </div>
</div>
```

```
<template>
  <nav>
    <router-link to="/">
      Home
    </router-link> |
    <router-link to="/about">
      About
    </router-link>
  </nav>
  <router-view />
</template>
```

Як це працює?

localhost:8080/about

```
import { createRouter, createWebHistory } from 'vue-router'
import HomeView from '../views/HomeView.vue'
import AboutView from '../views/AboutView.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/about',
    name: 'about',
    component: AboutView,
  },
]
```

```
<div id="app" data-v-app>
  <nav>
    <a href="/" class> Home </a>
    |
    <a href="/about" class="router-link-active router-link-exact-active" aria-current="page"> About
  </nav>
  <div class="about"> == $0
    <h1>This is an about page</h1>
  </div>
</div>
```

[Home](#) | [About](#)

This is an  
about page

> views > AboutView.vue > template

```
1 <template>
2   <div class="about">
3     <h1>This is an about page</h1>
4   </div>
5 </template>
```

```
<template>
  <nav>
    <router-link to="/">
      Home
    </router-link> |
    <router-link to="/about">
      About
    </router-link>
  </nav>
  <router-view />
</template>
```

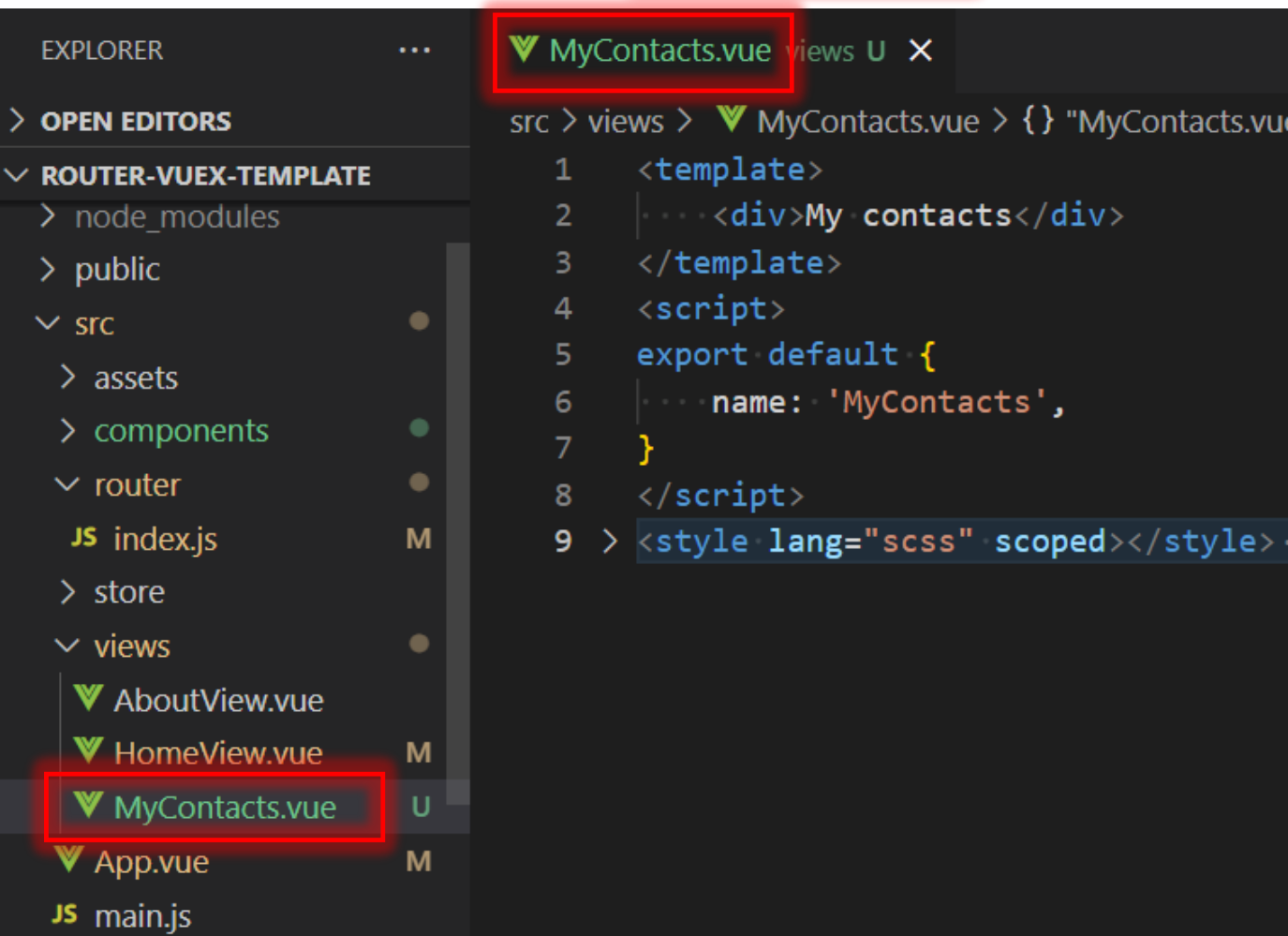
Додамо ще одну сторінку – ***Contacts***

Додамо ще одну сторінку – ***Contacts***

1) Створюємо компонент ***MyContacts.vue***

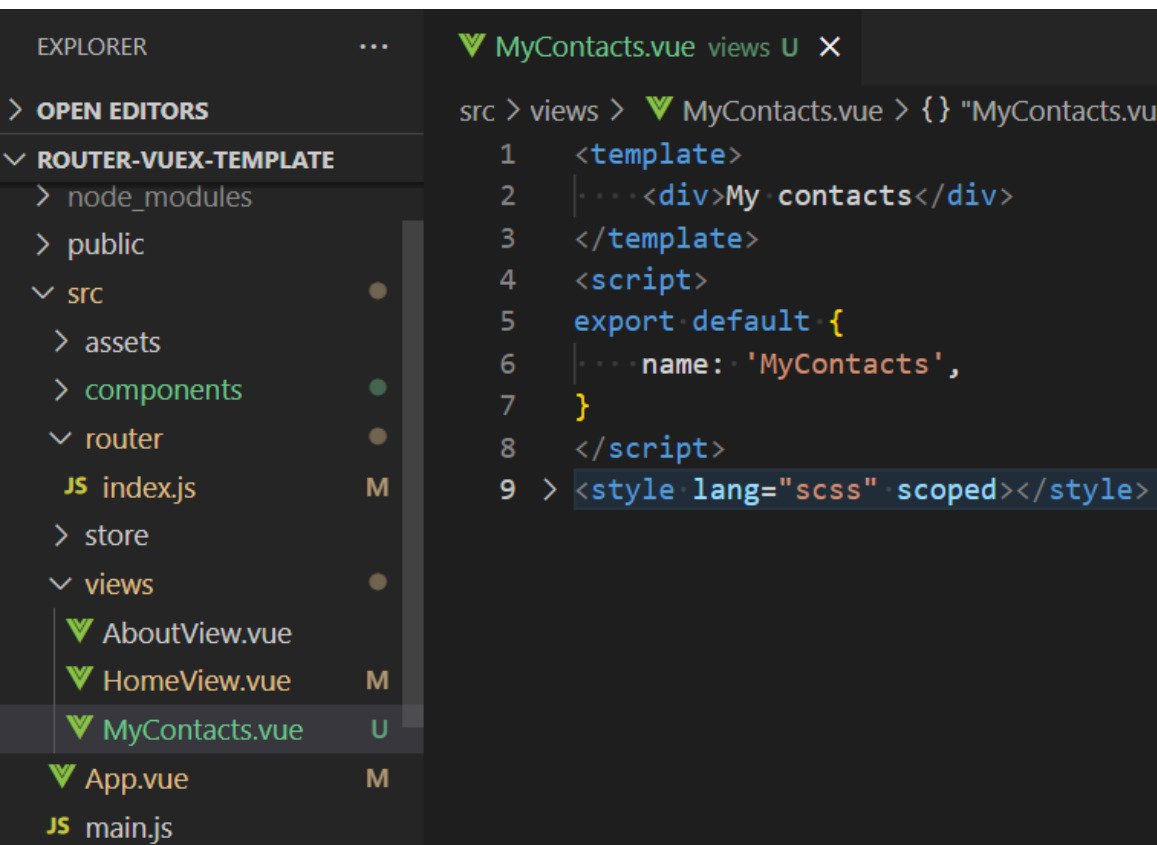
Додамо ще одну сторінку – **Contacts**

### 1) Створюємо компонент **MyContacts.vue**

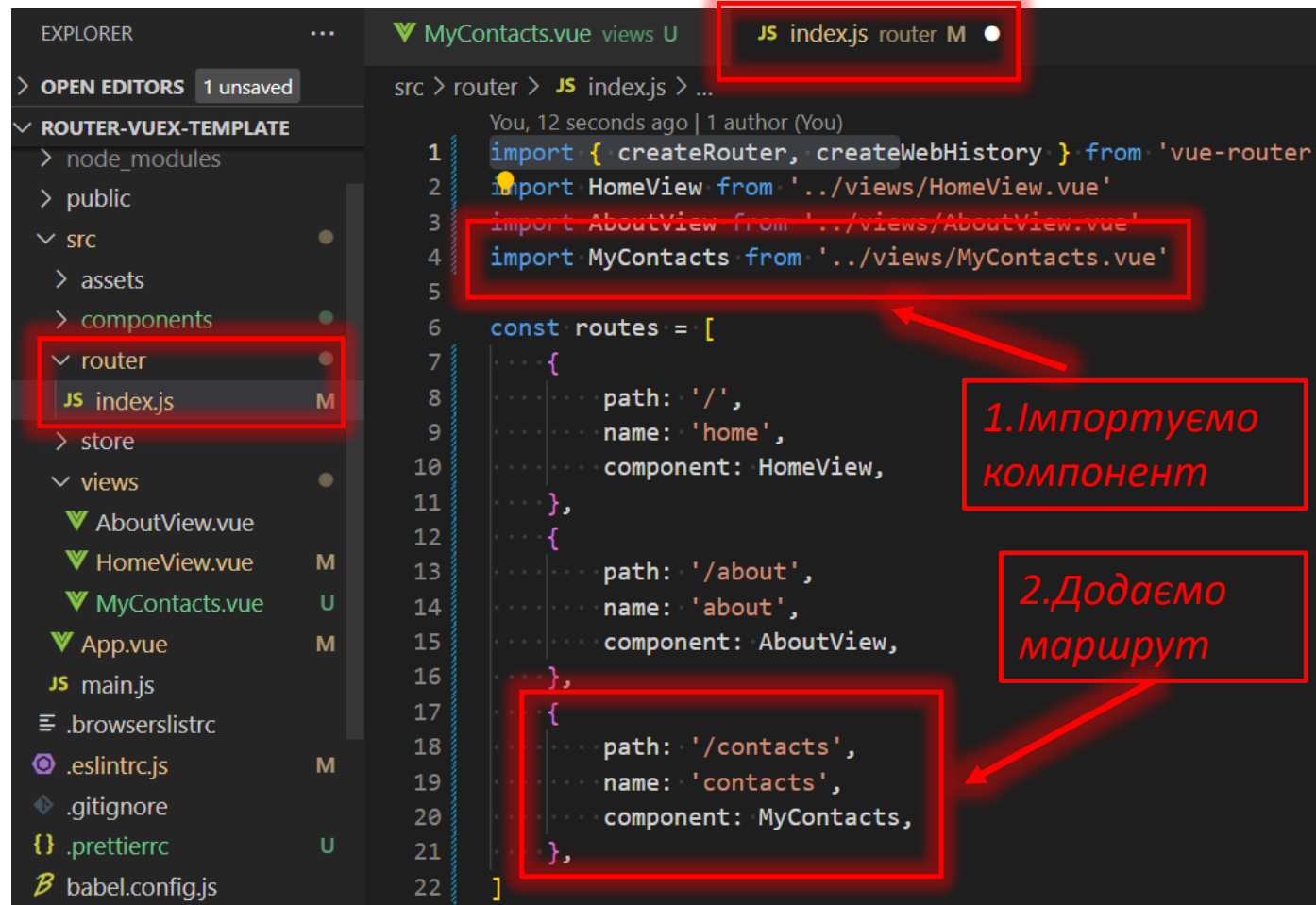


Додамо ще одну сторінку – **Contacts**

### 1) Створюємо компонент *MyContacts.vue*



### 2) Додаємо маршрут у файл налаштувань





## Створення посилань для переходу між маршрутами з використанням `<router-link>`

- ----- Можна вказувати шлях -----

```
<router-link to="шлях">  
  вміст посилання  
</router-link>
```

//---- місце монтування компонента, який відповідає шляху

```
<router-view></router-view>
```

```
<template>  
  <div>  
    <h1>Hello App!</h1>  
    <p>  
      <!-- ===ПРИКЛАД ВИКОРИСТАННЯ ПОСИЛАНЬ З ШЛЯХАМИ === -->  
      <router-link to="/">Go to Home</router-link>  
      <router-link to="/products-list">  
        Go to Products  
      </router-link>  
    </p>  
  </div>  
</template>
```

```
    <!-- route outlet, component matched by the route  
will render here -->  
    <router-view></router-view>  
  </div>  
</template>
```

# Створення посилань для переходу між маршрутами з використанням *<router-link>*

- - - - - Можна вказувати шлях - - - - -

```
<router-link to="шлях">  
  вміст посилання  
</router-link>
```

- - Можна вказувати ім'я маршруту і параметри за потреби - -

```
<router-link :to="{  
  name: 'ім'я_маршруту',  
  params: {параметр: значення}  
}">  
  вміст посилання  
</router-link>
```

//---- місце монтування компонента, який відповідає шляху

```
<router-view></router-view>
```

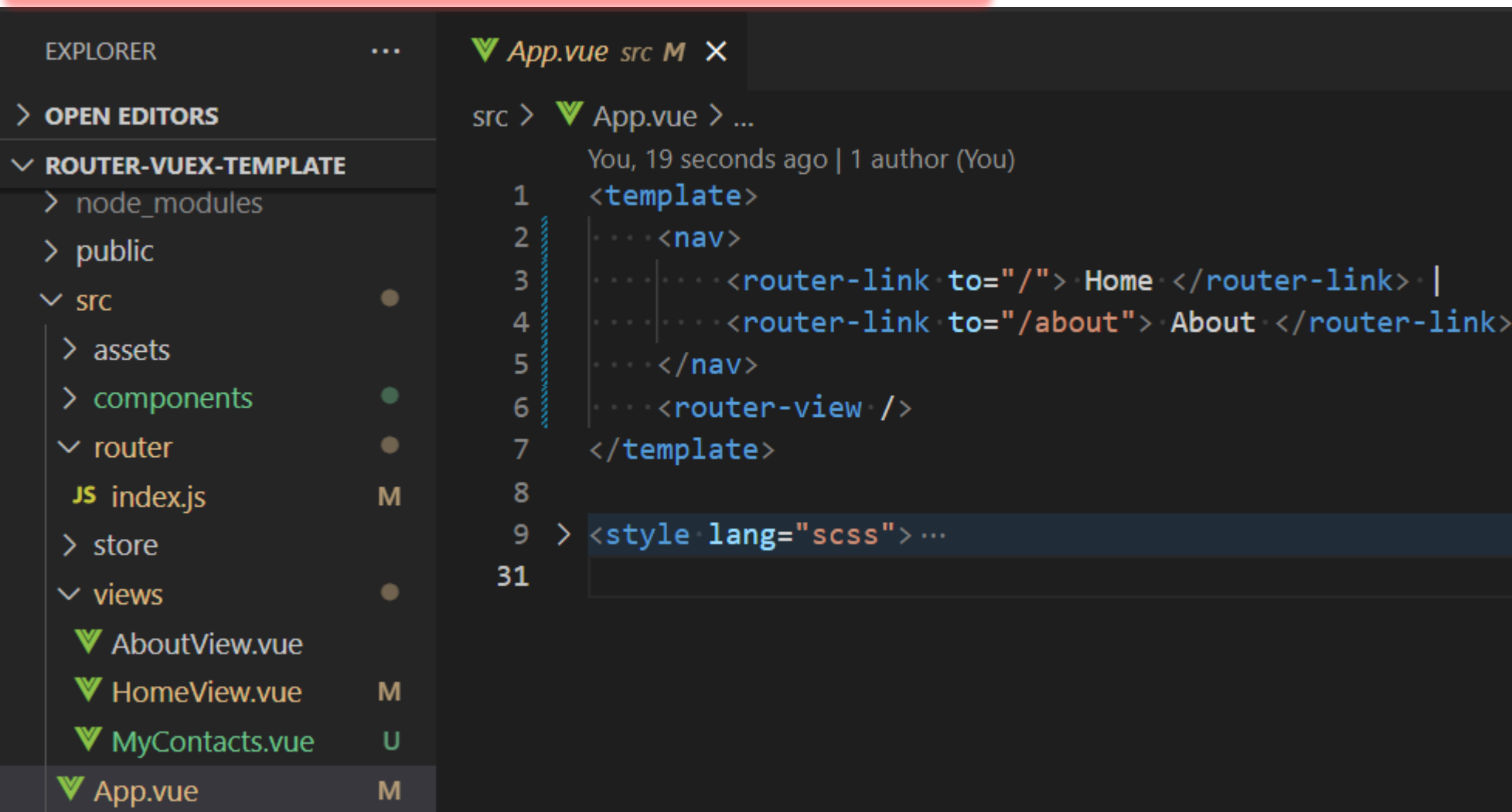
```
<template>  
  <div>  
    <h1>Hello App!</h1>  
    <p>  
      <!-- ===ПРИКЛАД ВИКОРИСТАННЯ ПОСИЛАНЬ З ШЛЯХАМИ === -->  
      <router-link to="/">Go to Home</router-link>  
      <router-link to="/products-list">  
        Go to Products  
      </router-link>  
      <router-link to="/about">Go to About</router-link>  
  
      <!-- ===ПРИКЛАД ВИКОРИСТАННЯ ПОСИЛАНЬ З ІМЕНАМИ  
           МАРШРУТІВ== -->  
      <router-link :to="{ name: 'home' }">  
        Go to Home  
      </router-link>  
      <router-link :to="{ name: 'products' }">  
        Go to Products  
      </router-link>  
      <router-link :to="{ name: 'about' }">  
        Go to About  
      </router-link>  
    </p>  
    <!-- route outlet, component matched by the route  
    will render here -->  
    <router-view></router-view>  
  </div>  
</template>
```

Додамо ще одну сторінку – **Contacts**

1) Створюємо компонент **MyContacts.vue**

2) У файлі налаштувань додати взаємозв'язок між маршрутом і компонентом

3) Додати посилання на сторінку контактів



Додамо ще одну сторінку – **Contacts**

- 1) Створюємо компонент **MyContacts.vue**
- 2) У файлі налаштувань додати взаємозв'язок між маршрутом і компонентом
- 3) Додати посилання на сторінку контактів

EXPLORER

OPEN EDITORS

ROUTER-VUEX-TEMPLATE

- node\_modules
- public
- src
  - assets
  - components
  - router
    - index.js
  - store
  - views
    - AboutView.vue
    - HomeView.vue
    - MyContacts.vue

App.vue src M X

- App.vue
  - template
    - nav
      - router-link

1<template>

2<nav>

3<router-link to="/"> Home </router-link> |

4<router-link to="/about"> About </router-link>

5<router-link to="/contacts"> My Contacts </router-link>

6</nav>

7<router-view />

8</template>

9

10<style lang="scss"> ...

32

Home | About My Contacts

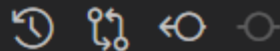
My contacts

# Передача параметрів

## 1. Опис параметрів у маршрутах

```
const routes = [
  // Опис параметра у шляху : параметр
  {
    path: 'шлях / : параметр',
    component: компонент-для-відображення,
    name: ' ім'я_маршрута ',
  },
  // Опис необов'язкового параметра у
  // шляху : параметр?
  {
    path: 'шлях / : параметр?',
    component: компонент-для-відображення,
    name: ' ім'я_маршрута ',
  },
]
```

JS index.js router M



src > router > JS index.js > routes

You, 22 seconds ago | 1 author (You)

```
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3 import AboutView from '../views/AboutView.vue'
4 import MyContacts from '../views/MyContacts.vue'
5 import MyPredictor from '../views/MyPredictor.vue'
6
7 const routes = [
8   {
9     path: '/',
10    name: 'home',
11    component: HomeView,
12  },
13  {
14    path: '/predictor/:person',
15    name: 'predictor',
16    component: MyPredictor,
17  },
18 ]
```

localhost:8080/predictor/man

[Home](#) | [About My Contacts](#)

# Передача параметрів

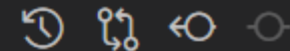
## 1. Опис параметрів у маршрутах

```
const routes = [
  // Опис параметра у шляху : параметр
  {
    path: 'шлях / : параметр',
    component: компонент-для-відображення,
    name: ' ім'я_маршрута ',
  },
  // Опис необов'язкового параметра у
  // шляху : параметр?
  {
    path: 'шлях / : параметр?',
    component: компонент-для-відображення,
    name: ' ім'я_маршрута ',
  },
]
```

## 2. Отримуємо параметри маршрута у компонентах

```
this.$route.params. параметр
```

JS index.js router M



src > router > JS index.js > routes

You, 22 seconds ago | 1 author (You)

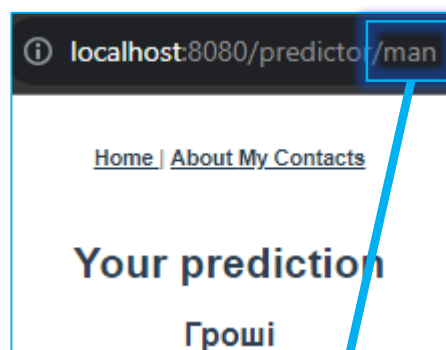
```
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3 import AboutView from '../views/AboutView.vue'
4 import MyContacts from '../views/MyContacts.vue'
5 import MyPredictor from '../views/MyPredictor.vue'
6
7 const routes = [
8   {
9     path: '/',
10    name: 'home',
11    component: HomeView,
12  },
13  {
14    path: '/predictor/:person',
15    name: 'predictor',
16    component: MyPredictor,
17  },
18 ]
```

localhost:8080/predictor/man

```
this.$route.params. person
```

# Передача параметрів

```
const routes = [
  // Опис параметра у шляху : параметр
  {
    path: 'шлях / : параметр',
    component: компонент-для-відображення,
    name: 'ім'я_маршрута',
  },
  // Опис необов'язкового параметра у
  // шляху : параметр?
  {
    path: 'шлях / : параметр?',
    component: компонент-для-відображення,
    name: 'ім'я_маршрута',
  },
]
```



JS index.js router M

src > router > JS index.js > routes

```
You, 22 seconds ago | 1 author (You)
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3 import AboutView from '../views/AboutView.vue'
4 import MyContacts from '../views/MyContacts.vue'
5 import MyPredictor from '../views/MyPredictor.vue'
6
7 const routes = [
8   {
9     path: '/',
10    name: 'home',
11    component: HomeView,
12  },
13   {
14    path: '/predictor/:person',
15    name: 'predictor',
16    component: MyPredictor,
17  },
18 ]
```

MyPredictor.vue views U X

src > views > MyPredictor.vue > {} "MyPredictor.vue"

```
1 <template>
2   <div>
3     <h1>Your prediction</h1>
4     <h2>{{ currentPrediction }}</h2>
5   </div>
6 </template>
7
8 <script>
9   const predictions = {
10     boy: ['Іграшка', 'Цукерка', 'Торт'],
11     man: ['Гроші', 'Авто', 'Шовдарь'],
12   }
13   export default {
14     name: 'MyPredictor',
15     computed: {
16       currentPerson() {
17         return this.$route.params.person
18       },
19       currentPrediction() {
20         const randIndex = Math.floor(Math.random() * predictions[
21           this.currentPerson
22         ].length)
23         return predictions[this.currentPerson][randIndex]
24       },
25     },
26   }
27 </script>
```

//Зчитування параметра у компоненті

**this.\$route.params.** параметр

**Створимо сторінку, при переході на яку здійснюється перевірка прикладу таблиці множення (вказуємо два числа, а якщо **друге** не вказано, то вибираємо випадково)**

**`/test/ перше_число / друге_число`**

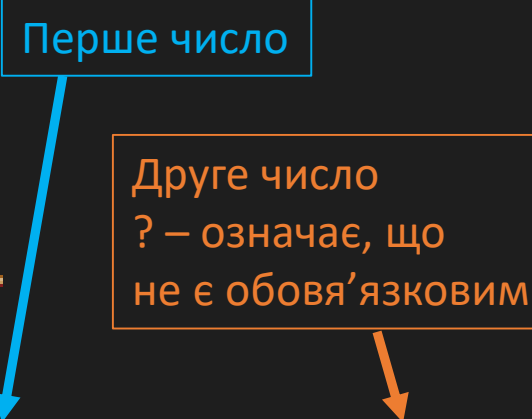


**Створимо сторінку, при переході на яку здійснюється перевірка прикладу таблиці множення (вказуємо два числа, а якщо друге не вказано, то вибираємо випадково)**

`/test/ перше_число / друге_число`

```
import MyMultTest from '../views/MyMultTest.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/test/:first_num/:second_num?',
    name: 'test',
    component: MyMultTest,
  },
]
```



**Створимо сторінку, при переході на яку здійснюється перевірка прикладу таблиці множення (вказуємо два числа, а якщо друге не вказано, то вибираємо випадково)**

`/test/ перше_число / друге_число`

```
import MyMultTest from '../views/MyMultTest.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/test/:first_num/:second_num?',
    name: 'test',
    component: MyMultTest,
  },
]
```

Перше число

Друге число  
? – означає, що  
не є обов'язковим

localhost:8080/test / 7 / 3

[Home](#) | [About My Contacts](#)

7\*3 =

Відповідь:

Створимо сторінку, при переході на яку здійснюється перевірка прикладу таблиці множення (вказуємо два числа, а якщо друге не вказано, то вибираємо випадково)

`/test/ перше_число / друге_число`

```
import MyMultTest from '../views/MyMultTest.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/test/:first_num/:second_num?',
    name: 'test',
    component: MyMultTest,
  },
]
```

Перше число

Друге число  
? – означає, що  
не є обов'язковим

localhost:8080/test / 7 / 3

[Home](#) | [About My Contacts](#)

7\*3 =

Відповідь:

Перевірити

# Перевірка прикладу таблиці множення

`/test/` *перше\_число* / *друге\_число*

```
import MyMultTest from '../views/MyMultTest.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/test/:first_num/:second_num?',
    name: 'test',
    component: MyMultTest,
  },
]
```

Перше число

Друге число  
? – означає, що  
не є обов'язковим

```
export default {
  name: 'MyPredictor',
  data() {
    return {
      userAnswer: null,
      result: null,
    }
  },
  computed: {
    firstNum() { return parseInt(this.$route.params.first_num) },
    secondNum() {
      return this.$route.params.second_num ||
        1 + Math.floor(Math.random() * 10)
    },
  },
  methods: {
    check() {
      if (this.firstNum * this.secondNum === this.userAnswer) {
        this.result = 'Ok'
      } else {
        this.result = 'Error'
      }
    },
  },
}
</script>
```

# Перевірка прикладу таблиці множення

`/test/` *перше\_число* / *друге\_число*

```
import MyMultTest from '../views/MyMultTest.vue'

const routes = [
  {
    path: '/',
    name: 'home',
    component: HomeView,
  },
  {
    path: '/test/:first_num/:second_num?',
    name: 'test',
    component: MyMultTest,
  },
]
```

Перше число

Друге число  
? – означає, що  
не є обов'язковим

MyMultTest.vue views U

src > views > MyMultTest.vue > {} "MyMultTest.vue" > template > div > label > input

```
1 <template>
2   <div>
3     <h1>{{ firstNum }}*{{ secondNum }}=</h1>
4     <label>
5       Відповідь:<input v-model="userAnswer" type="number"/>
6     </label>
7     <button @click="check">Перевірити</button>
8     <h2 v-if="result">{{ result }}</h2>
9   </div>
10 </template>
11 <script>
12 export default {
13   name: 'MyPredictor',
14   data() {
15     return {
16       userAnswer: null,
17       result: null,
18     },
19     computed: {
20       firstNum() { return parseInt(this.$route.params.first_num); },
21       secondNum() {
22         return this.$route.params.second_num ||
23           1 + Math.floor(Math.random() * 10)
24       },
25     },
26     methods: {
27       check() {
28         if (this.firstNum * this.secondNum === this.userAnswer) {
29           this.result = 'Ok'
30         } else {
31           this.result = 'Error'
32         }
33       }
34     }
35   }
36 </script>
```

# Програмний перехід між маршрутами

	Загальна форма	Приклад
Вказуємо шляхи	<pre>this.\$router.push(   {     path: 'шлях',   } )</pre>	<pre>this.\$router.push(   {     path: '/products/provider',   } )</pre>
Задання параметрів запиту	<pre>this.\$router.push(   {     path: 'шлях',     query: {       параметр: 'значення',       . . . . .     }   } )</pre>	<pre>this.\$router.push(   {     path: '/register',     query: {       plan: 'private'     }   } )</pre> Result: <code>/register?plan=private</code>
Вказуємо імена маршрутів та можливо параметри маршруту	<pre>this.\$router.push(   {     name: 'назва_маршруту',     params: {       параметр: 'значення',       . . . . .     }   } )</pre>	<pre>this.\$router.push(   {     name: 'user',     params: {       username: 'Ivan',       age: 21     }   } )</pre> Route: <code> '/user/:username/:age'</code> Result: <code> '/user/Ivan/21'</code>

# Програмний перехід між маршрутами

	Загальна форма
Вказуємо шляхи	<pre>this.\$router.push(   {     path: 'шлях',   } )</pre>

Вказуємо імена маршрутів та можливо параметри маршруту	<pre>this.\$router.push(   {     name: 'назва_маршруту',     params: {       параметр: 'значення',       . . . . .     }   } )</pre>
--	--

```
<template>  
  <div class="home">  
    <h2 @click="onContactsClick">Контакти</h2>  
    <h2 @click="onTestClick">Тест : 8 * 3</h2>  
    <h2 @click="onPredictClick">Передбачення для хлопця</h2>  
  </div>  
</template>  
  
<script>  
export default {  
  name: 'HomeView',  
  methods: {  
    onContactsClick() {  
      this.$router.push({  
        path: '/contacts',  
      })  
    },  
    onTestClick() {  
      // this.$router.push({ path: '/test/8/3' })  
      const num1 = 8,  
            num2 = 3  
      this.$router.push({  
        path: `/test/${num1}/${num2}`,  
      })  
    },  
    onPredictClick() {  
      this.$router.push({  
        name: 'predictor',  
        params: {  
          person: 'boy',  
        },  
      })  
    },  
  },  
}
```

Приклад. Розробити додаток з такими сторінками і маршрутами

Маршрут: /

Home Users Contacts

Вітаємо на сайті

Маршрут: /users

Home Users Contacts

Іванов І.І.  
Петров П.П.  
Сидоров С.С.  
Григоренко Г.П.  
Артем С.О.

Маршрут: /users/userId

Home Users Contacts

Користувач: Іванов І.І.  
Стаж: 21  
Посада: директор

Маршрут: /contacts

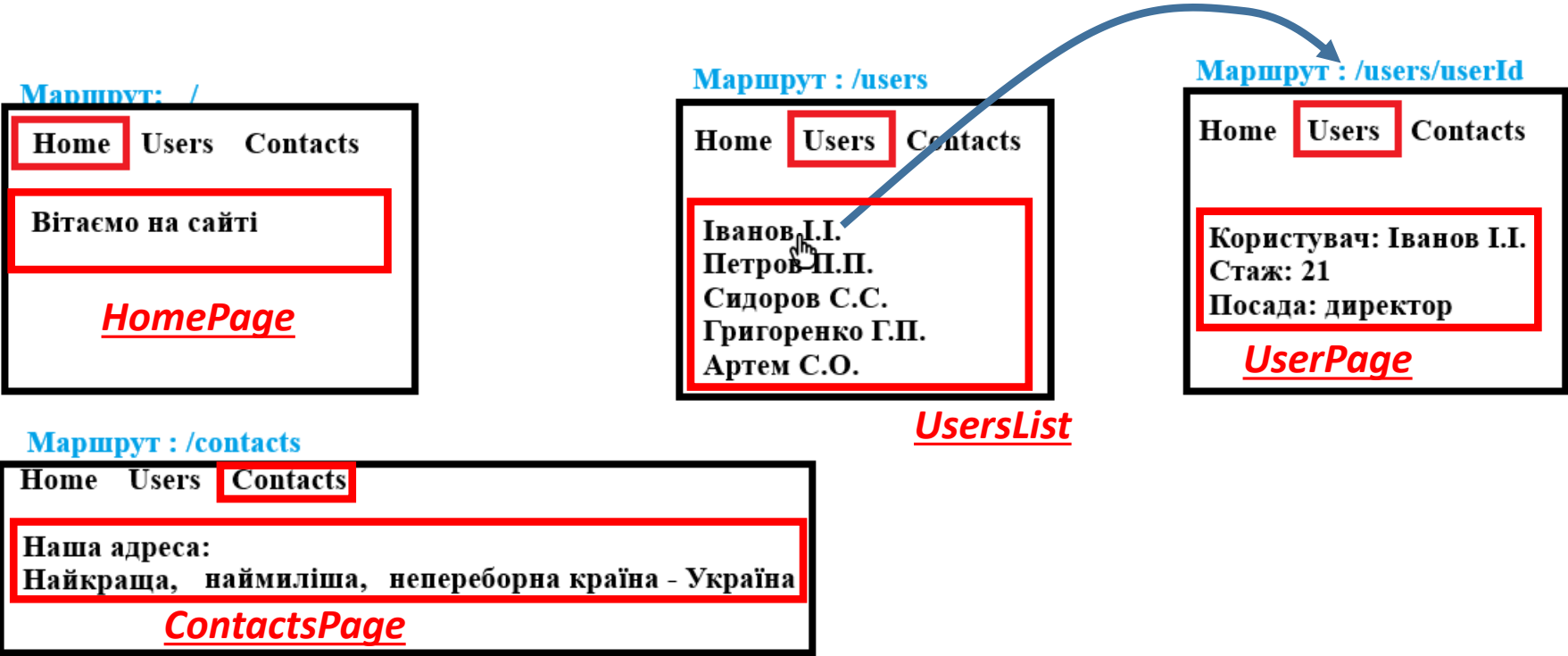
Home Users Contacts

Наша адреса:  
Найкраща, наймиліша, непереборна країна - Україна



Приклад. Розробити додаток з такими сторінками і маршрутами

**Розподіл на компоненти**



Приклад. Розробити додаток з такими сторінками і маршрутами

**Визначаємось з взаємодією з store**

Маршрут : /users



**UsersList**

**Store**

**State:**

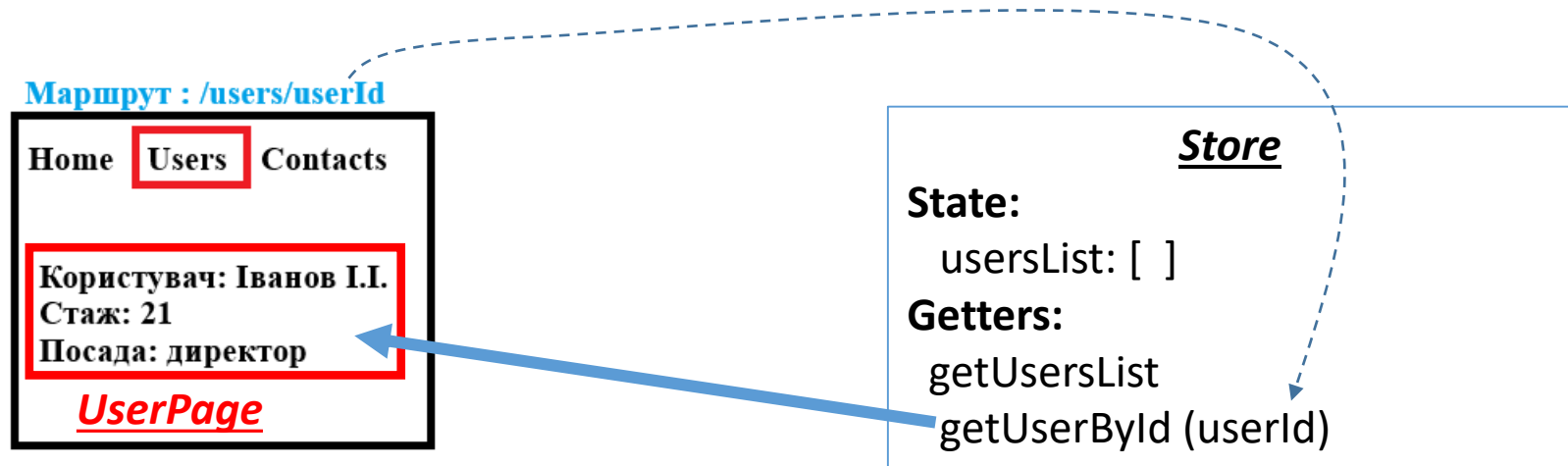
usersList: [ ]

**Getters:**

getUsersList

Приклад. Розробити додаток з такими сторінками і маршрутами

**Визначаємось з взаємодією з store**



Приклад. Довідник податківця. База даних підприємств: назва підприємства, рівень оподаткування, рік заснування, власник. Організувати вибір за довільним запитом.

[Home](#) | [About](#) | [Інформація про контакти](#)

## Filters

Title  Rate  Year  Owner

## Companies list

Comp1	5	2000	Ivanov I.I.	Delete	Edit
Comp2	21	2021	Petrov I.I.	Delete	Edit
Comp3	15	2003	Sara I.I.	Delete	Edit

Create a company

[Home](#) | [About](#) | [Інформація про контакти](#)

## This is an about page

[Home](#) | [About](#) | [Інформація про контакти](#)

My Contacts

← → ↻ ⓘ localhost:8080

[Home](#) | [About](#) | [Інформація про контакти](#)

## Filters

Title  Rate  Year  Owner

---

## Companies list

Comp1	5	2000	Ivanov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp2	21	2021	Petrov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp3	15	2003	Sara I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>

### Додавання компанії

← → ↻ ⓘ localhost:8080/company

[Home](#) | [About](#) | [Інформація про контакти](#)

Title

Rate

Year

Owner

← → ↻ ⓘ localhost:8080

[Home](#) | [About](#) | [Інформація про контакти](#)

## Filters

Title  Rate  Year  Owner

---

## Companies list

Comp1	5	2000	Ivanov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp2	21	2021	Petrov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp3	15	2003	Sara I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>

### Редагування інформації про компанію

← → ↻ ⓘ localhost:8080/company/2

[Home](#) | [About](#) | [Інформація про контакти](#)

Title

Rate

Year

Owner

**Розподіл на компоненти**

[Home](#) | [About](#) | [Інформація про контакти](#)

**CompaniesManager**

**CompaniesFilters**

**Filters**

Title  Rate  Year  Owner

**Companies list**

Comp1	5	2000	Ivanov I.I.	Delete	Edit
Comp2	21	2021	Petrov I.I.	Delete	Edit
Comp3	15	2003	Sara I.I.	Delete	Edit

Create a company

**CompaniesList**

**CompanyItem**

← → ↻ ⓘ localhost:8080

[Home](#) | [About](#) | [Інформація про контакти](#)

## Filters

Title  Rate  Year  Owner

## Companies list

Comp1	5	2000	Ivanov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp2	21	2021	Petrov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp3	15	2003	Sara I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>

### Додавання компанії

← → ↻ ⓘ localhost:8080/company

[Home](#) | [About](#) | [Інформація про контакти](#)

Title

Rate

Year

Owner

### CompanyEditor

### Редагування інформації про компанію

← → ↻ ⓘ localhost:8080

[Home](#) | [About](#) | [Інформація про контакти](#)

## Filters

Title  Rate  Year  Owner

## Companies list

Comp1	5	2000	Ivanov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp2	21	2021	Petrov I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>
Comp3	15	2003	Sara I.I.	<input type="button" value="Delete"/>	<input type="button" value="Edit"/>

← → ↻ ⓘ localhost:8080/company/2

[Home](#) | [About](#) | [Інформація про контакти](#)

Title

Rate

Year

Owner

## Розподіл на компоненти

[Home](#) | [About](#) | [Інформація про контакти](#)

This is an about page

*AboutPage*

[Home](#) | [About](#) | [Інформація про контакти](#)

My Contacts

*ContactsPage*



### Companies list

Comp1	5	2000	Ivanov I.I.	Delete	Edit
Comp2	21	2021	Petrov I.I.	Delete	Edit
Comp3	15	2003	Sara I.I.	Delete	Edit

Create a company

Store

State:

companiesList : [ ]

**Getters:**

getCompaniesList

**Actions:**

removeCompany(companyId)



[Home](#) | [About](#) | [Інформація про контакти](#)

## Filters

Title  Rate  Year  Owner

### Store

#### State:

```
companiesList : [ ]  
filterObj: { }
```

#### **Getters:**

```
getCompaniesList  
getFilterObj
```

#### **Actions:**

```
removeCompany(companyId)  
updateFilter(filterData)
```



## Додавання компанії

← → ↻ ⓘ localhost:8080/company

[Home](#) | [About](#) | [Інформація про контакти](#)

Title

Rate

Year

Owner

Create

## CompanyEditor

## Редагування інформації про компанію

← → ↻ ⓘ localhost:8080/company/2

[Home](#) | [About](#) | [Інформація про контакти](#)

Title

Rate

Year

Owner

Save

## Store

### State:

```
companiesList : [ ]  
filterObj: { }
```

### **Getters:**

```
getCompaniesList  
getFilterObj  
getCompanyById(companyId)
```

### **Actions:**

```
removeCompany(companyId)  
updateFilter(filterData)  
addCompany(company)  
updateCompany(company)
```