

2025년 상반기 K-디지털 트레이닝

vue-router를 이용한 라우팅 1

[KB] IT's Your Life



☑ SPA(단일 페이지 애플리케이션: Single Page Application)

- 하나의 페이지 안에서 데스크톱 애플리케이션과 같은 사용자 경험을 제공
- 여러 화면을 하나의 페이지 안에서 제공, 화면을 별도로 로딩하지 않음
- → 화면마다 고유의 식별자(URI)를 기반으로 화면을 렌더링해야 함
- 요청한 URI 경로에 따라 각각 다른 화면이 렌더링되도록 함
- → vue-router 라이브러리 이용

createRouter()

- o router 객체를 생성
 - URI와 이 경로를 처리할 컴포넌트 매핑

```
import { createRouter, createWebHistory } from 'vue-router'
...

const router = createRouter({
    history: createRouter(),
    routes: [
        { path: '/', components: Home },
        { path: '/about', components: About },
        { path: '/members', components: Members },
        { path: '/videos', components: Videos},
        ]
});
```

🧿 router 객체의 등록

```
const app = createApp(App)
app.use(router)
app.mount('#app')
```

<RouterView>

각 경로별 컴포넌트를 렌더링할 위치를 지정하는 컴포넌트

✓ <RouterLink to="경로">

○ 화면 전환 을 위한 링크 생성

<router-link to="[등록시킬 URI 경로]">[링크 텍스트]</router-link>

<RouterLink to="/">Home</RouterLink>

💟 실습 프로젝트 생성

```
npm init vue router-test
Vue.js – The Progressive JavaScript Framework
√ Add TypeScript? ... No / Yes
√ Add JSX Support? ... No / Yes

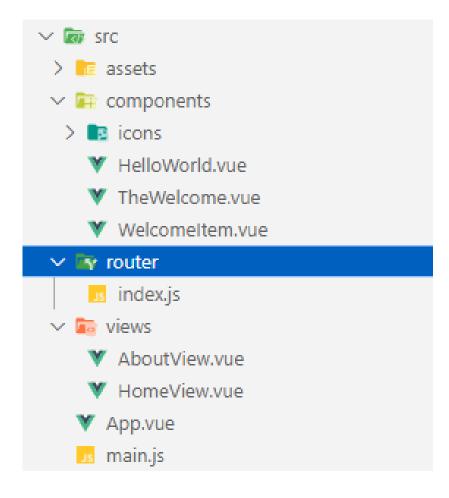
√ Add Vue Router for Single Page Application development? ... No / Yes

√ Add Pinia for state management? ... <u>No</u> / Yes
√ Add Vitest for Unit Testing? ... No / Yes
\sqrt{\text{Add an End-to-End Testing Solution?}} \gg \underline{\text{No}}
√ Add ESLint for code quality? ... No / Yes
 cd router-test
 npm install
```

package.json

```
"name": "router-test",
"version": "0.0.0",
"private": true,
"type": "module",
"scripts": {
 "dev": "vite",
 "build": "vite build",
  "preview": "vite preview"
"dependencies": {
  "vue": "^3.3.11",
 "vue-router": "^4.2.5"
"devDependencies": {
  "@vitejs/plugin-vue": "^4.5.2",
  "vite": "^5.0.10"
```

☑ 기본 골격



src/router/index.js

```
import { createRouter, createWebHistory } from 'vue-router'
import HomeView from '../views/HomeView.vue'
const router = createRouter({
  history: createWebHistory(import.meta.env.BASE_URL),
  routes: [
      path: '/',
      name: 'home',
      component: HomeView
    },
      path: '/about',
      name: 'about',
      // route level code-splitting
      // this generates a separate chunk (About.[hash].js) for this route
      // which is lazy-loaded when the route is visited.
      component: () => import('../views/AboutView.vue')
export default router
```

src/main.js

```
import './assets/main.css'
import { createApp } from 'vue'
import App from './App.vue'
import router from './router'

const app = createApp(App)

app.use(router)

app.mount('#app')
```

src/App.vue

```
<script setup>
import { RouterLink, RouterView } from 'vue-router'
import HelloWorld from './components/HelloWorld.vue'
</script>
<template>
  <header>
    <img alt="Vue logo" class="logo" src="@/assets/logo.svg" width="125" height="125" />
    <div class="wrapper">
      <HelloWorld msg="You did it!" />
      <nav>
        <RouterLink to="/">Home</RouterLink>
        <RouterLink to="/about</pre>/RouterLink>
      </nav>
    </div>
  </header>
  <RouterView />
</template>
<style scoped>
</style>
```



You did it!

You've successfully created a project with Vite + Vue 3.

Home About

□ Documentation

Vue's official documentation provides you with all information you need to get started.



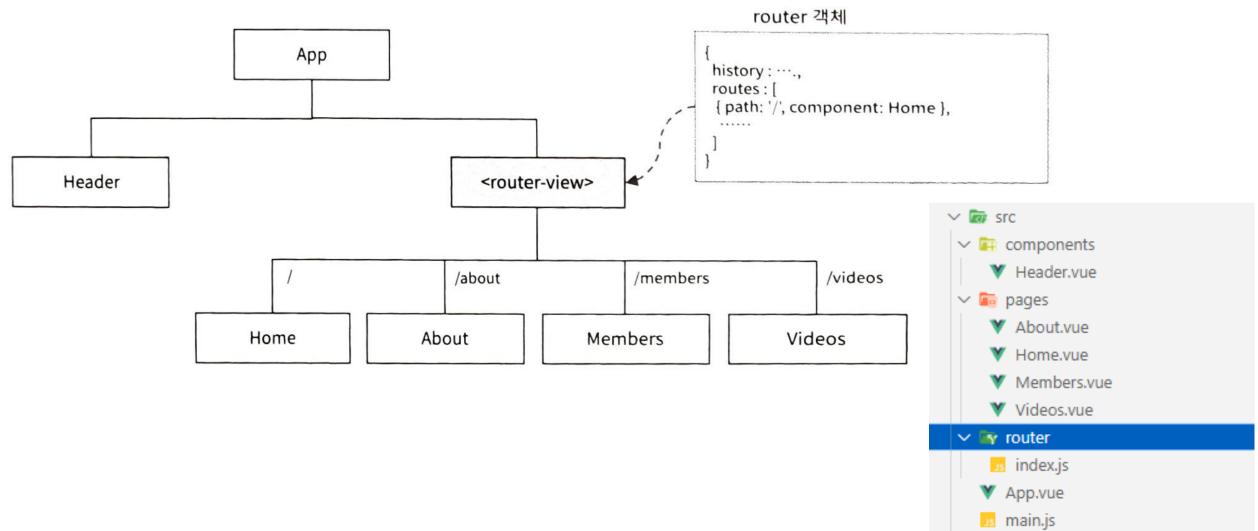
You did it!

You've successfully created a project with Vite + Vue 3.

Home About

This is an about page

☑ 작업할 프로젝트 컴포넌트 구조



☑ 부트스트랩5 설치

npm install bootstrap@5

🗸 page 컴포넌트 정의

- o src/pages
 - Home.vue
 - About.vue
 - Members.vue
 - Videos.vue

■ name과 타이틀을 각 페이지마다 지정

src/router/index.js

```
import { createRouter, createWebHistory } from 'vue-router'
import Home from '@/pages/Home.vue'
import About from '@/pages/About.vue'
import Members from '@/pages/Members.vue'
import Videos from '@/pages/Videos.vue'
const router = createRouter({
    history: createWebHistory(),
    routes : [
        { path: '/', component: Home },
        { path: '/about', component: About },
        { path: '/members', component: Members },
        { path: '/videos', component: Videos },
})
export default router;
```

src/main.js

```
import { createApp } from 'vue'
import 'bootstrap/dist/css/bootstrap.css'
import App from './App.vue'
import router from './router'

const app = createApp(App)
app.use(router)
app.mount('#app')
```

src/components/Header.vue

```
<template>
 <nav class="navbar navbar-expand-md bg-dark navbar-dark mt-2">
   <span class="navbar-brand">이날치(LeeNalChi)</span>
   <button class="navbar-toggler" type="button" @click="changeIsNavShow">
    <span class="navbar-toggler-icon"></span>
   </button>
   <div :class="navClass">
    <router-link class="nav-link" to="/">喜</router-link>
      <router-link class="nav-link" to="/about"> 소개</router-link>
      <router-link class="nav-link" to="/members">멤버</router-link>
      <router-link class="nav-link" to="/videos">영상</router-link>
      </div>
 </nav>
</template>
```

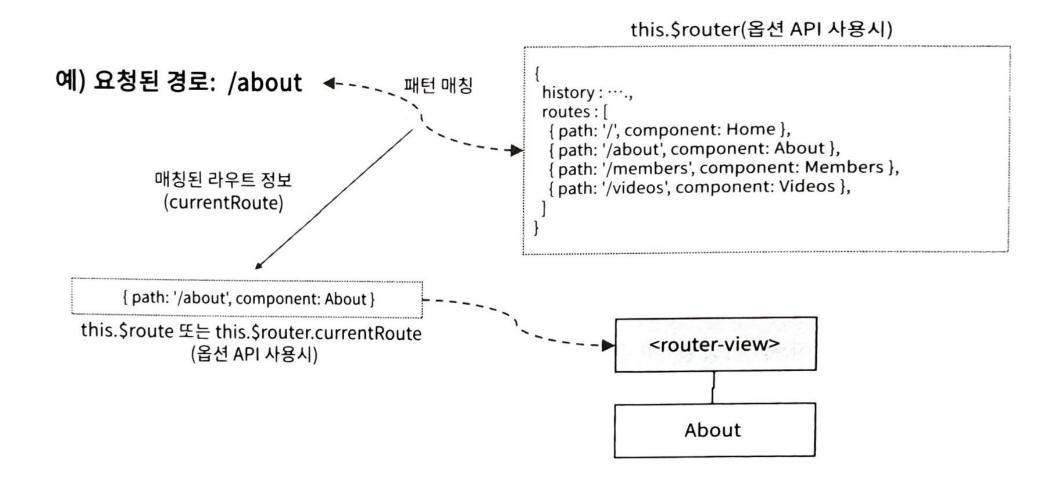
src/components/Header.vue

```
이날치(LeeNalChi) 홈 소개 멤버 영상
<script>
import { reactive, computed } from 'vue';
                                                                                     Home
export default {
  setup() {
                                                              이날치(LeeNalChi)
    const state = reactive({ isNavShow: false });
                                                                         Home
    const navClass = computed(() =>
      state.isNavShow
        ? 'collapse navbar-collapse show'
        : 'collapse navbar-collapse'
    );
    const changeIsNavShow = () => { state.isNavShow = !state.isNavShow; };
    return { state, changeIsNavShow, navClass };
</script>
```

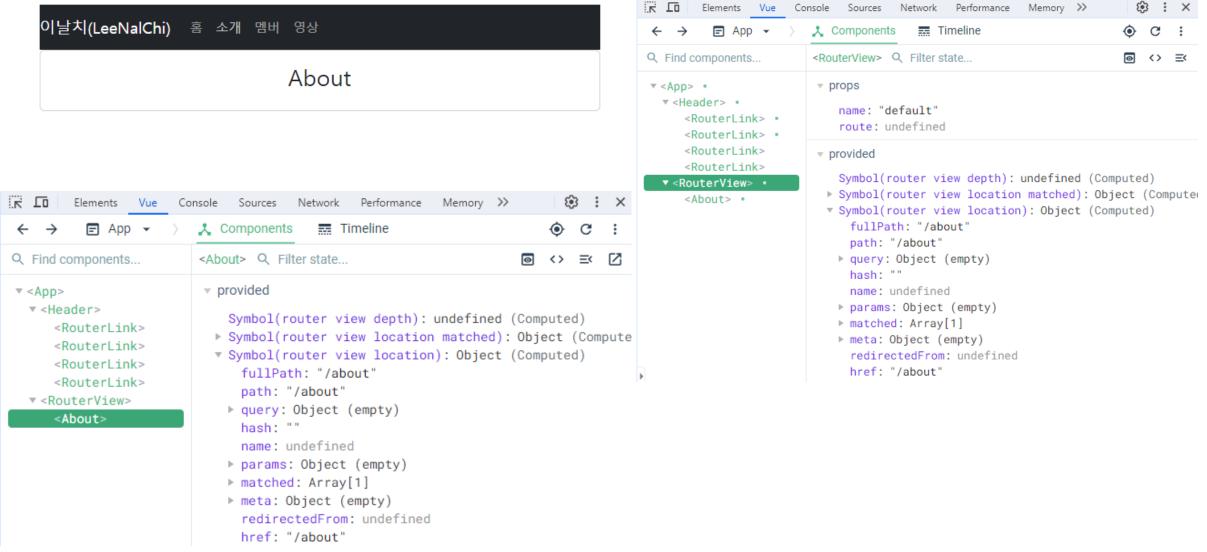
src/App.vue

```
<template>
  <div class="container">
    <Header />
    <router-view></router-view>
  </div>
</template>
<script>
import Header from '@/components/Header.vue'
export default {
  name : "App",
  components : { Header },
</script>
<style>
.container {
  text-align: center;
  margin-top:10px;
</style>
```

🥝 <router-view>의 컴포넌트 마운트 방법



vue-router 기본 적용 방법 예제 실행



💟 컴포넌트에서 router 객체 접근하기

- o this.\$router
 - src/router/index.js에서 작성한 router 객체
 - 전체 라우트 정보 파악
 - push(), replace(), go() 등의 메서드 이용가능

o Router 관련 정보 접근

구분	Options API	Composition API
라우터 객체	this.\$router	<pre>const router = useRouter()</pre>
매칭된 라우트 (currentRoute)	this.\$route - 또는-	<pre>const currentRoute = useRoute()</pre>
	this.\$router.currentRoute	

3 router 객체와 currentRoute 객체

☑ 컴포넌트에서 router 객체 접근하기

o currentRoute 객체의 주요 속성

구분 설명 fullPath 전체 요청 경로, 쿼리문자열까지 포함(ex: /about?a=1&b=2) matched vue—router 객체의 routes 배열의 라우트 중 매칭된 라우트 params URI 경로에 동적으로 전달된 파라미터 정보, 자세한 내용은 다음 절(동적 라우트)에서 다룸 path 요청 URI 경로 query 쿼리 문자열 정보 ?a=1&b=2로 요청되었다면 this.\$route.query는 { a:1, b:2 }와 같은 객체에 redirectedFrom 다른 경로에서 리디렉트된 경우 리디렉트시킨 URI 경로 정보를 포함			
matched vue—router 객체의 routes 배열의 라우트 중 매칭된 라우트 params URI 경로에 동적으로 전달된 파라미터 정보, 자세한 내용은 다음 절(동적 라우트)에서 다룸 path 요청 URI 경로 query 쿼리 문자열 정보 ?a=1&b=2로 요청되었다면 this.\$route.query는 { a:1, b:2 }와 같은 객체	구분	설명	
params URI 경로에 동적으로 전달된 파라미터 정보, 자세한 내용은 다음 절(동적 라우트)에서 다룸 path 요청 URI 경로 query 쿼리 문자열 정보 ?a=1&b=2로 요청되었다면 this.\$route.query는 { a:1, b:2 }와 같은 객체(fullPath	전체 요청 경로, 쿼리문자열까지 포함(ex: /about?a=1&b=2)	
path 요청 URI 경로 query 쿼리 문자열 정보 ?a=1&b=2로 요청되었다면 this.\$route.query는 { a:1, b:2 }와 같은 객체	matched	vue-router 객체의 routes 배열의 라우트 중 매칭된 라우트	
query 쿼리 문자열 정보 ?a=1&b=2로 요청되었다면 this.\$route.query는 { a:1, b:2 }와 같은 객체	params	URI 경로에 동적으로 전달된 파라미터 정보, 자세한 내용은 다음 절(동적 라우트)에서 다룸	
	path	요청 URI 경로	
redirectedFrom 다른 경로에서 리디렉트된 경우 리디렉트시킨 URI 경로 정보를 포함	query	쿼리 문자열 정보 ?a=1&b=2로 요청되었다면 this,\$route.query는 { a:1, b:2 }와 같은 객체(
	redirectedFrom	다른 경로에서 리디렉트된 경우 리디렉트시킨 URI 경로 정보를 포함	

▶ [[Prototype]]: Object

3 router 객체와 currentRoute 객체

src/pages/About.vue

```
<template>
    <div class="card card-body">
         <h2>About</h2>
        요청 경로 : {{$route.fullPath}}
    </div>
</template>
<script>
export default {
  name : "About",
  created() {
    console.log(this.$route)
                                                                                                                       Elements
                                                                                                                                Vue
                                                                                                                                    Console
                       이날치(LeeNalChi) 홈 소개 멤버 영상
</script>
                                                                                                                      top ▼ ◎ Filter
                                                                                                               No Issues
                                                                                                                 ▼ Object 1
                                                           About
                                                                                                                     fullPath: "/about"
                                                                                                                     hash: ""
                                                        요청 경로:/about
                                                                                                                     href: "/about"
                                                                                                                   ▶ matched: [{...}]
                                                                                                                   ▶ meta: {}
                                                                                                                     name: undefined
                                                                                                                   ▶ params: {}
                                                                                                                     path: "/about"
                                                                                                                   ▶ query: {}
                                                                                                                     redirectedFrom: undefined
```

3 router 객체와 currentRoute 객체

src/pages/Members.vue

```
<template>
   <div class="card card-body">
       <h2>Members</h2>
       요청 경로 : {{currentRoute.fullPath}}
   </div>
</template>
<script>
import { useRoute } from 'vue-router'
export default {
 name : "Members",
 setup() {
   const currentRoute = useRoute();
   return { currentRoute };
</script>
```

