

2025년 상반기 K-디지털 트레이닝

vue-router를 이용한 라우팅(심화1)

[KB] IT's Your Life

- ✓ **NotFound 페이지 컴포넌트를 정의하세요.**
 - 404 Not Found 출력
 - 잘못된 요청 경로 출력
- ✓ **잘못된 요청이 온 경우 NotFound 페이지로 이동하도록 라우터 모듈을 수정하세요.**

src/pages/NotFound.vue

```
<template>
  <div class="card card-body">
    <h2>404 Not Found</h2>
    <p>존재하지 않는 요청 경로 : {{$route.fullPath}}</p>
  </div>
</template>

<script>
export default {
  name: "NotFound",
  created() {
    console.log(this.$route.params)
  }
}
</script>
```

src/router/index.js

```
...
import NotFound from '@pages/NotFound.vue'
...
const router = createRouter({
  history: createWebHistory(),
  routes : [
    { path: '/', name:'home', component: Home },
    { path: '/about', name:'about', component: About },
    { path: '/members', name:'members', component: Members, },
    {
      path: '/members/:id', name:'members/id', component: MemberInfo,
      beforeEnter:membersIdGuard
    },
    {
      path: '/videos', name:'videos', component: Videos,
      children : [
        { path: ':id', name:'videos/id', component: VideoPlayer }
      ]
    },
    { path: '/*', name: 'NotFound', component: NotFound },
  ]
})
...
export default router;
```

2025년 상반기 K-디지털 트레이닝

vue-router를 이용한 라우팅(심화2)

[KB] IT's Your Life

- ✓ 다음 조건은 만족하는 라우트 수준의 가드를 추가하세요.
 - 이전 경로가 /members, members/:id인 경우만 members/:id 경로로 이동

src/router/index.js

```
import { createRouter, createWebHistory, isNavigationFailure } from 'vue-router'
...
const membersIdGuard = (to, from) => {
  // members/:id 경로는 반드시 이전 경로가
  // /members, /members:id 인 경우만 내비게이션 허용함
  if (from.name !== "members" && from.name !== "members/id") {
    return false;
  }
}
```

src/router/index.js

```
const router = createRouter({
  history: createWebHistory(),
  routes : [
    { path: '/', name:'home', component: Home },
    { path: '/about', name:'about', component: About },
    { path: '/members', name:'members', component: Members, },
    {
      path: '/members/:id', name:'members/id', component: MemberInfo,
      beforeEnter:membersIdGuard
    },
    {
      path: '/videos', name:'videos', component: Videos,
      children : [
        { path: ':id', name:'videos/id', component: VideoPlayer }
      ]
    },
  ],
})
```