

# Роутинг та навігація в React

**№ уроку:** 7 **Курс:** React Базовий

**Засоби навчання:** Visual Studio Code, пакет-менеджер node.

## Огляд, ціль та призначення уроку

Навчитись встановлювати роутинг в реакт-застосунках. Ознайомитися з бібліотекою React Router актуальної версії. Оволодіти технологією роботи з методами бібліотеки React Router.

## Вивчивши матеріал даного заняття, студент зможе:

- Встановлювати маршрутизацію в React застосунках.
- Обробляти помилки неіснуючих адрес.
- Працювати з динамічним роутингом.
- Використовувати найактуальніші методи реакт-роутингу (loader і т. д.).

## Зміст уроку

1. Визначення роутингу.
2. Механізм роутингу в React.
3. Метод `createBrowserRouter`.
4. Обробка неіснуючих адрес (сторінка з помилкою).
5. Елементи `Link` та `NavLink`.
6. Вкладені маршрути та `loader`.

## Резюме

- Роутинг або маршрутизація - це процес організації маршрутів між посиланнями. У вебзастосунках роутингом називають зміну URL-адреси в момент переходу по різних частинам інтернет-ресурсу.
- За замовчуванням бібліотека React не надає інструментів для роутингу, для того, щоб забезпечити застосунок маршрутизацією, використовується бібліотека React Router. Для її встановлення всередині проєкту, в терміналі потрібно прописати команду `npm i react-router-dom`.
- Для створення маршрутизації потрібно створити компонент `Root`, в якому буде описана основна структура вашого проєкту та компоненти, які не повинні змінюватись (`header`, `footer`, `sidebar` і т. д.). В тому місці, де ви хочете бачити інші компоненти при зміні `url`-адреси, потрібно розмістити елемент `<Outlet />`.
- Для побудови маршрутизації React застосунку використовується метод `createBrowserRouter`. Щоб передавати атрибути та адреси у вигляді JSX-компонентів, використовується допоміжний метод `createRoutesFromElements`.
- Якщо за посиланням вебдодатка такої сторінки не передбачено, необхідно реалізувати шаблон сторінки з помилкою. Щоб даний компонент викликався при запиті до відсутньої сторінки, його потрібно передати в останній дочірній `Route` з атрибутом `path='*'`.

- Щоб застосунок не оновлювався при переходах між розділами, всі посилання в ньому повинні бути реалізовані за допомогою елементів Link або NavLink.
- Якщо в вашому застосунку є сторінки, які продовжують гілку посилань, наприклад, профіль конкретного юзера в адресі /users/ , то для маршрутизації таких сторінок в атрибуті path після головної гілки потрібно вказати двокрапку та типізовану назву такої сторінки.
- Метод loader дозволяє зробити запит до зовнішньої системи (API) та передати отримані дані компоненту, який викликається. Для цього створюється окрема функція, яка передається в атрибут loader. За аргумент функція loader може приймати ім'я сторінки, для якої необхідно зробити запит.
- Для коректного відображення помилки в компонентах, які мають згенеруватись динамічно, наприклад, url адреса неіснуючого юзера, в елементі Route, який повинен обробляти даний маршрут, потрібно додати атрибут errorElement з компонентом для відображення помилки.
- Після того як маршрутизація була налаштована в root.render, потрібно передати елемент RouterProvider з атрибутом router={ назва константи createBrowserRouter }.

### Закріплення матеріалу

- Як встановити react-router-dom в проєкт?
- За допомогою якого методу встановлюється роутинг?
- Як всередині методу loader отримати параметри динамічної сторінки, яка викликається?

### Додаткове завдання

Завдання

Створити власний React-додаток з кількома сторінками. Налаштувати маршрутизацію для цього додатку.

### Самостійна діяльність студента

Завдання 1

Повторити практичний приклад з уроку, для цього в матеріалах до уроку потрібно встановити директорію start\_materials.

Завдання 2

Додати декілька власних сторінок, встановити для них маршрути та створити посилання в компоненті Sidebar.

Завдання 3

Замість json файлу userData всередині методу loader реалізувати fetch за посиланням <https://jsonplaceholder.typicode.com/users> , отримані дані відфільтрувати та передати в return той об'єкт який необхідно. Для виконання цього завдання підготуйте компонент Users таким чином, щоб він обробляв посилання з іменами із <https://jsonplaceholder.typicode.com/users> . Також модифікуйте компонент UserPage, для виведення даних об'єкту із jsonplaceholder.

### Рекомендовані ресурси

Документація React Router

