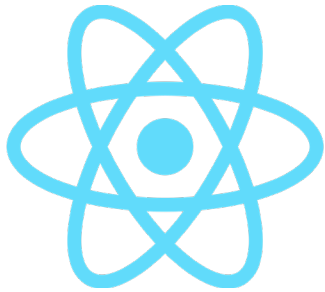




Microsoft Partner
Silver Learning



React Базовий

Управління станами функціональних компонентів

Введення в хуки

React Базовий

Introduction

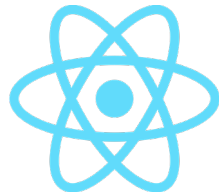


Кочев Денис

Web-розробник в Data Bridge

Магістр комп'ютерних наук Національного аерокосмічного університету ім. М.Є Жуковського

 [denkochev](https://www.linkedin.com/in/denkochev)



React Базовий

Тема уроку

Управління станами функціональних компонентів
Введення в хуки

React Базовий

План уроку

1. Що таке хуки в React?
2. Навіщо використовувати хуки?
3. Класифікація хуків
4. Правила використання хуків
5. Вбудовані в React хуки
6. Хук стану в функціональних компонентах

React Базовий

Що таке хуки в React?

Хуки - це готові програмні рішення для функціональних компонентів. Хуками в реакті, називають спеціальні методи які дозволяють застосовувати стани та інші розширення реакту до функціональних компонентів.

До появи React версії 16.8 у функціональних компонентів не було інструментів для роботи зі станами, рефами та ефектами, що значно зменшувало їх потенціал для використання, в той час коли у класових компонентів існував власний життєвий цикл, в середині якого можна було контролювати стан елементів та миттєво відображати їхні зміни в інтерфейсі.



React Базовий

Навіщо використовувати хуки?

Насамперед використання хуків при розробці React-застосунків дозволяє розробляти якісний, простий та масштабований код.

Хуки здатні значно полегшити повторне використання коду оскільки можуть вирішити загальні проблеми з якими стикається фронтенд розробник.

Хуки здатні зменшити загальну кількість компонентів за рахунок свого функціоналу.



React Базовий

Які хуки існують

```
const forUser = useRef();  
const [user, setUser] = useState([]);
```

Хуки useRef та useState

За замовчуванням в бібліотеці React вбудовано близько 10 основних хуків, але завдяки стороннім розробникам існують сотні готових хуків, які здатні вирішити більшість задач з якими ви можете зіткнутися в процесі розробки.

Існують навіть цілі бібліотеки з користувацькими хуками, якими користуються більшість реакт-розробників, одна з таких бібліотек - це бібліотека react-use.

Посилання на бібліотеку react-use:

<https://github.com/streamich/react-use>

React Базовий

Правила використання хуків

1. Хуки викликаються тільки **на верхньому рівні**, не потрібно викликати хуки всередині циклів або вкладених функцій.
2. Хуки викликаються тільки із функціональних компонентів в React, так як в класових компонентах є окремі інструменти для реалізації функціоналу хуків.
3. Хуки ні в якому разі не повинні використовуватись всередині умовних операторів.

React Базовий

Вбудовані в React хуки

За замовчуванням в React є вбудовані хуки, які повторюють функціонал класових компонентів. В рамках даного курсу ми розглянемо деякі з них, а саме:

- Хук для роботи зі станами `useState`;
- Хук для прямого доступу до Dom-елементу `useRef`;
- Хук роботи зі сторонніми ефектами `useEffect`.

React Базовий

Що таке хук стану та як його використовувати

Хук станів `useState()` дозволяє перерисовувати компонент, кожен раз коли state буде змінено.

Для роботи з даним хуком його потрібно імпортувати всередину компоненту.

Для цього використовується наступний синтаксис:

```
import React, {useState} from 'react'
```

Створення хуку станів відбувається через вказання імені змінної та імені методу для його зміни:

```
const [counter, setCounter] = useState(0);
```

В якості аргументу при виклику методу `useState` передається початкове значення стейту.

React Базовий

Підсумки уроку

1. Хуки - це методи для роботи зі станами життєвого циклу React всередині функціональних компонентів.
2. Хуки використовуються тільки на верхньому рівні та можуть викликатись тільки із React функцій.
3. Хук стану `useState` дозволяє створити об'єкт, при зміні значення якого буде перерисовано той DOM-елемент де це значення використовується.
4. Для використання масивів в якості типів даних для стейтів, при їх зміні потрібно наявно вказувати, що масив було змінено, для цього використовується синтаксис копіювання (оператор `spread`).

Інформаційний відеосервіс для розробників програмного забезпечення



Перевірка знань

TestProvider.com



Перевірте як ви засвоїли даний матеріал на [TestProvider.com](https://testprovider.com)

TestProvider – це online сервіс перевірки знань з інформаційних технологій. За його допомогою Ви можете оцінити ваш рівень та виявити слабкі місця. Він буде корисним як у процесі вивчення технології, так і для загальної оцінки знань IT спеціаліста.

Успішне проходження фінального тестування дозволить вам отримати відповідний сертифікат.

React Базовий

Дякую за увагу! До нових зустрічей!



Кочев Денис
Web-розробник в Data Bridge

