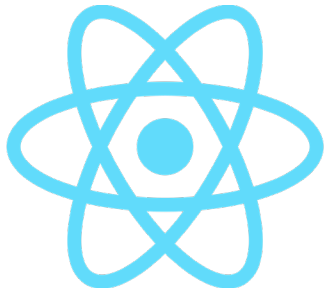




Microsoft Partner
Silver Learning



React Базовий

Основи роботи з React компонентами

React Базовий

Introduction

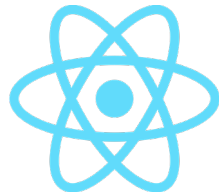


Кочев Денис

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

Основи роботи з React компонентами

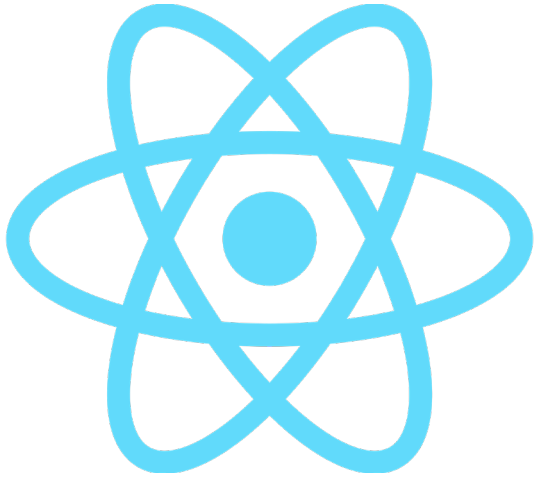
React Базовий

План уроку

1. Визначення компонентів в реакті
2. Які компоненти існують
3. Передача даних між компонентами
4. Застосування отриманих знань на практиці

React Базовий

Що таке компоненти в React?

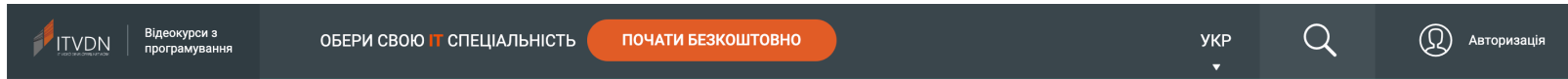


Компонент - це одиниця (будівельний блок) для побудови інтерфейсів на React. З компонентів будуються React-застосунки.

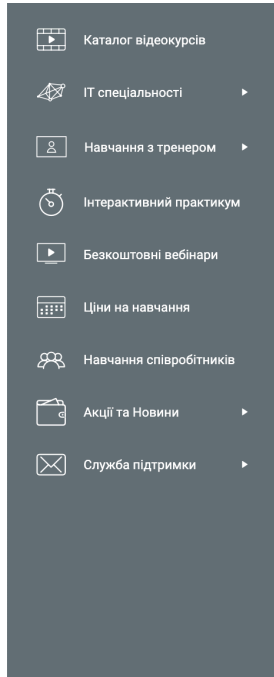
Використання компонентів при побудові інтерфейсів дозволяє розділити додаток на незалежні частини які можна розробляти та проектувати окремо.

Головною перевагою такого методу побудови веб-застосунків є можливість повторного використання коду (компонентів) та розподіленість при розробці.

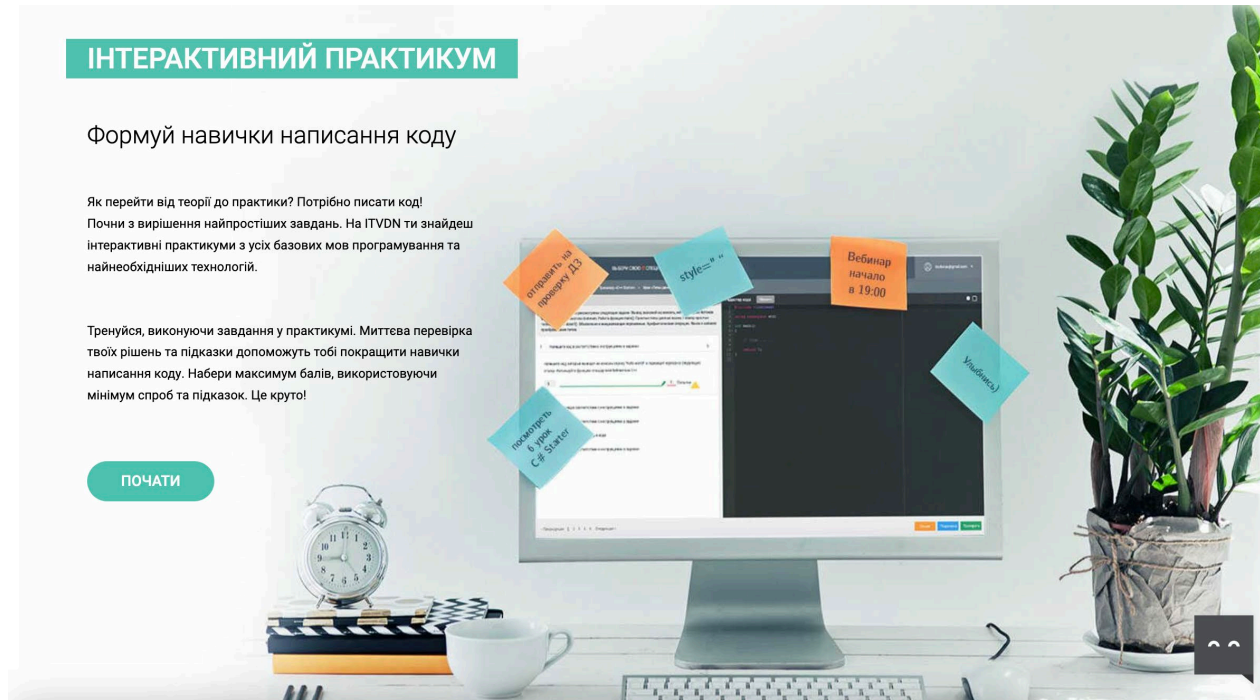
React Базовий



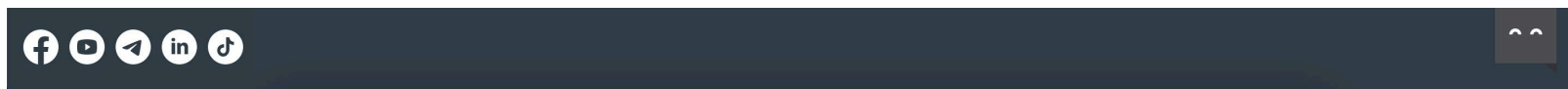
Компонент Header



Компонент Nav



Компонент Main



Компонент Footer

Інтернет-портал ITVDN можна представити у вигляді наступних компонентів:

1. Header
2. Nav
3. Main
4. Footer

React Базовий

Які компоненти існують

Компоненти в реакті можна створювати двома способами: у функціональному стилі програмування та у об'єктно орієнтовному стилі програмування. З першим способом ми вже знайомі, це створення компонентів за допомогою функції, другий же спосіб передбачає створення компоненту за допомогою класів.

На сьогоднішній день найбільш актуальним способом створення компонентів є функціональний стиль, але в деяких випадках повністю відмовитись від ООП стилю неможливо.

```
function App() {  
  return (  
    <>  
    <h1>Функціональний компонент!</h1>  
    </>  
  );  
}  
export default App;
```

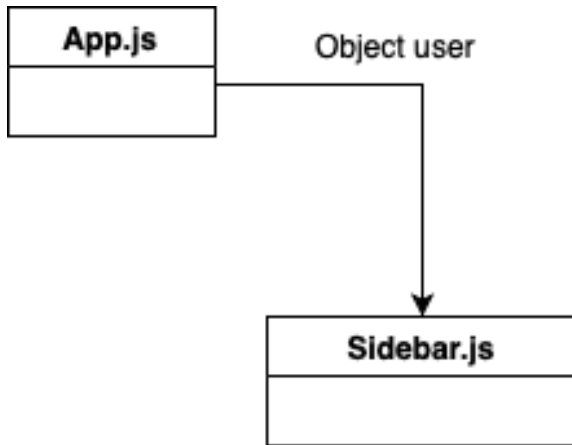
Функціональний компонент

```
class App extends React.Component{  
  render(){  
    return (  
      <>  
      <h1>Класовий компонент!</h1>  
      </>  
    );  
  }  
}
```

Класовий компонент

React Базовий

Передача даних між компонентами



Механізм передачі пропсів

Архітектура реакт-застосунків дозволяє передавати дані між компонентами. Для цього використовується об'єкт props. У даного методу передачі інформації є деякі обмеження, а саме:

1. Змінювати передану інформацію не можна (переданий об'єкт має залишитись в тому ж самому вигляді в якому і прийшов)
2. Дані можна передавати лише у односторонньому порядку

React Базовий

Практичне завдання

Розробити реакт-застосунок з відокремленими компонентами Header, Main, Sidebar та Footer.

React documentation page

Learn React

Початок роботи

З самого початку React був спроектований так, щоб його можна було впроваджувати поступово. Тобто ви можете додавати так мало або так багато React-у, як вам потрібно. Скористайтеся інформацією та посиланнями цього розділу, якщо просто бажаєте відчути React на смак або додаєте трохи “інтерактивності” до простої HTML-сторінки, чи, навіть, якщо стартуєте складний React-проект

Перенаправлення рефів

Перенаправлення рефів — це техніка для автоматичної передачі рефа від компонента до одного із його дітей. Для більшості компонентів, зазвичай, вона не є необхідною. Тим не менше, може бути корисною в деяких випадках, особливо якщо ви пишете бібліотеку. Даваймо розглянемо найбільш поширені сценарії.

Фрагменти

Повернення кількох елементів з компонента є поширеною практикою в React. Фрагменти дозволяють формувати список дочірніх елементів, не створюючи зайвих вузлів в DOM.

Початок роботи
Перенаправлення рефів
Фрагменти

React web-site

2023 copyrighting. All rights reserved.

Приклад побудови застосунка на реакті

React Базовий

Підсумки уроку

1. Інтерфейси веб-застосунків на React розробляються за допомогою компонентів. Компоненти - це незалежні елементи інтерфейсу, які можуть функціонувати та розроблятись незалежно один від одного.
2. Компоненти можуть бути розроблені за допомогою класу або функції. На сьогоднішній день найбільш популярним методом розробки компонентів є функціональний підхід, але в деяких задачах повністю відмовитись від класового підходу неможливо.
3. Між компонентами можна передавати дані в односторонньому порядку. Для цього використовуються пропси. Пропси не можна змінювати.

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 ІТ 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

