React: Styling Components



НАШИ ПРАВИЛА

Включенная камера

Вопросы по поднятой руке

Не перебиваем друг друга

Все вопросы, не связанные с тематикой курса (орг-вопросы и т. д.), должны быть направлены куратору

Подготовьте свое рабочее окружение для возможной демонстрации экрана (закройте лишние соцсети и прочие приложения)



Познакомиться с возможностями стилизации компонентов

план занятия

1. Styling Components



Styling Components

1 подход

Обычный CSS

Написание стилей происходит в отдельных CSS файлах и затем они импортируются в ваши компоненты

```
/* styles.css */
.myComponent {
  color: blue;
  font-size: 16px;
}
```

```
// MyComponent.js
import React from 'react';
import './styles.css'; // Импорт стилей

const MyComponent = () => {
  return <div className="myComponent">Пример компонента</div>;
};

export default MyComponent;
```

2 подход Inline Styles:

Стили указываются напрямую внутри JSX элемента.

```
const MyComponent = () => {
  return <div style={{ color: 'blue', fontSize: '16px' }}>Пример компонента</div>;
};
```

```
С помощью создания отдельной переменной
```

```
const MyComponent = () => {
  const style = {
    color: 'blue',
    fontSize: '16px',
  };

return <div style={style}>Пример компонента</div>;
};
```

3 подход CSS-in-JS библиотеки, такой как emotion

Она предоставляет множество возможностей для создания стилей, включая локальную область видимости и использование динамических стилей.

Установка

Для начала, установите emotion в вашем проекте с помощью прт

npm install --save @emotion/react

npm install --save @emotion/styled

3 подход - emotion

Основная концепция

Emotion позволяет создавать стилизованные компоненты с использованием синтаксиса тегов.

```
import styled from '@emotion/styled';

export const StyledButton = styled.button`
  color: white;
  background-color: #3498db;
  border: none;
  border-radius: 5px;
  cursor: pointer;
`;
```

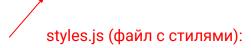
```
styles.js (файл с стилями):
```

MyComponent.js (компонент, использующий стили):

Возможности emotion Использование props

Вы можете передавать пропсы и использовать их для определения стилей.

```
import styled from
'@emotion/styled';
const StyledButton = styled.button `
 color: ${props => (props.primary ?
'white' : 'black')};
 background-color: $ {props =>
(props.primary ? 'blue' :
'lightgray')};
```



MyComponent.js (компонент, использующий стили):

Возможности emotion Использование css функции.

Вы можете задавать общие стили и затем переиспользовать их с помощью функции css

```
import { css } from '@emotion/react';
import styled from '@emotion/styled';
const commonStyles = css`
 margin: 5px;
const StyledButton = styled.button
```

Возможности emotion

Глобальные стили

В emotion глобальные стили могут быть заданы с использованием компонента Global.

<u>1 шаг</u>. Создаём стили и компонент в файле GlobalStyles.tsx

```
import { Global, css } from "@emotion/react";
const globalStyles = css`
 box-sizing: border-box;
function GlobalStyles() {
 return <Global styles={globalStyles} />;
export default GlobalStyles;
```

Возможности emotion

Глобальные стили

В emotion глобальные стили могут быть заданы с использованием компонента Global.

<u>2 шаг.</u> Используем глобальные стилей в нашем приложении, добавляя их с помощью компонента Global в App.tsx



Ваша новая IT-профессия – Ваш новый уровень жизни

Программирование с нуля в немецкой школе AIT TR GmbH

