

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)

Повторим;)

■ Какие типы мы можем передавать в дженерики?

■ Как создать enum?

■ Что такое поднятие состояния?

ЦЕЛЬ

Познакомиться с возможностями стилизации компонентов

ПЛАН ЗАНЯТИЯ



1. Styling Components

Styling Components



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 npm

```
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):

```
import { StyledButton } from './styles';

const MyComponent = () => {
  return (
    <div>
      <StyledButton onClick={()=>{}}>Click
    </StyledButton>
    </div>
  );
};

export default MyComponent;
```



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')};
  padding: 10px;
`;
```




styles.js (файл с стилями):

```
const MyComponent = () => {
  return (
    <div>

      <StyledButton onClick={() =>{}}>Standart button
    </StyledButton>

      <StyledButton primary onClick={() => {}}>Primary
button</StyledButton>

    </div>
  );
};
```



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`
  padding: 10px;
  margin: 5px;
`;

const StyledButton = styled.button`
  ${commonStyles};
  color: white;
  background-color: #3498db;
  border: none;
`;
```

Возможности emotion

Глобальные стили

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

1 шаг. 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.

2 шаг. Используем глобальные стили в нашем приложении, добавляя их с помощью компонента Global в App.tsx

```
import GlobalStyles from 'styles/GlobalStyles';

const App = () => {
  return (
    <>
      <GlobalStyles/>
      ...
    </>
  );
};

export default App;
```



Ваша новая IT-профессия – Ваш новый уровень жизни

Программирование с нуля в
немецкой школе AIT TR GmbH