

# React와 Material-UI를 사용한 간단한 웹 애플리케이션 만들기

이 튜토리얼에서는 `create-react-app` 을 사용하여 React 프로젝트를 설정하고, Material-UI 컴포넌트를 사용하여 간단한 웹 애플리케이션을 만드는 방법을 알아보겠습니다. 이 과정에서 버튼 클릭 시 알림 창이 표시되는 간단한 인터랙션도 구현해 볼 것입니다.

## 1. 프로젝트 생성

먼저, React 프로젝트를 생성합니다. 최신 버전의 `create-react-app` 을 사용하기 위해 `npx` 명령을 이용합니다.

```
npx create-react-app@latest my-material-ui-app
```

위 명령을 실행하면 `my-material-ui-app` 이라는 새로운 디렉토리에 React 프로젝트가 생성됩니다. 이 프로젝트 디렉토리로 이동합니다.

```
cd my-material-ui-app
```

## 2. Material-UI 설치

프로젝트에서 Material-UI를 사용하기 위해 필요한 패키지를 설치합니다.

```
npm install react react-dom  
npm install @mui/material @emotion/react @emotion/styled  
npm install --save-dev webpack webpack-cli webpack-dev-server babel-loader @babel/core @babel/preset-env @babel/preset-react
```

Material-UI는 React용 UI 컴포넌트 라이브러리로, 스타일링을 위해 `@emotion/react` 와 `@emotion/styled` 패키지를 함께 사용합니다.

### 3. 파일 구성 및 코드 작성

이제 React 컴포넌트를 작성하고, Webpack을 구성하여 애플리케이션을 빌드할 준비를 합니다.

### 3.1 App.js 작성

src/App.js 파일을 열고 아래 코드를 작성합니다. 이 코드에서는 Material-UI의 Button , Card , Typography , Container 컴포넌트를 사용합니다.

```

import React from 'react';
import Button from '@mui/material/Button';
import Card from '@mui/material/Card';
import CardContent from '@mui/material/CardContent';
import Typography from '@mui/material/Typography';
import Container from '@mui/material/Container';

function App() {
  // 버튼 클릭 시 실행될 함수
  const handleClick = () => {
    alert('버튼이 클릭되었습니다!');
  };

  return (
    <Container style={{ marginTop: '50px' }}>
      <Typography variant="h4" component="h1" gutterBottom>
        Material-UI 예제
      </Typography>

      <Card>
        <CardContent>
          <Typography variant="h5" component="div">
            React와 Material-UI
          </Typography>
          <Typography variant="body2" color="text.secondary">
            이 예제는 React와 Material-UI를 사용하여 만들어졌습니다.
          </Typography>
        </CardContent>
      </Card>

      <Button
        variant="contained"
        color="primary"
        style={{ marginTop: '20px' }}
        onClick={handleClick} // 버튼 클릭 시 handleClick 함수 호출
      >
        클릭하세요
      </Button>
    </Container>
  );
}

export default App;

```

이 코드에서 **Button** 컴포넌트를 클릭하면 **handleClick** 함수가 호출되어 알림 창이 표시됩니다.

## 3.2 index.js 작성

src/index.js 파일을 열고 아래 코드를 작성합니다. 이 파일은 App 컴포넌트를 렌더링하여 브라우저에 표시합니다.

```
import React from 'react';
import ReactDOM from 'react-dom';
import App from './App';

ReactDOM.render(<App />, document.getElementById('root'));
```

### 3.3 Webpack 설정 파일 작성 (`webpack.config.js`)

프로젝트의 루트 디렉토리에 `webpack.config.js` 파일을 생성하고, 아래와 같이 작성합니다.



```

const path = require('path');
const HtmlWebpackPlugin = require('html-webpack-plugin');

module.exports = {
  entry: './src/index.js',
  output: {
    path: path.join(__dirname, '/dist'),
    filename: 'bundle.js',
  },
  module: {
    rules: [
      {
        test: /\.?(js|jsx)$/,
        exclude: /node_modules/,
        use: {
          loader: 'babel-loader',
        },
      },
    ],
  },
  plugins: [
    new HtmlWebpackPlugin({
      template: './public/index.html',
    }),
  ],
  devServer: {
    contentBase: path.join(__dirname, 'dist'),
    compress: true,
    port: 3000,
  },
};

```

Webpack은 모듈 번들러로, JavaScript 코드를 번들링하고 빌드하는 데 사용됩니다. 이 설정에서는 `babel-loader` 를 사용하여 최신 JavaScript 코드를 변환하고, `HtmlWebpackPlugin` 을 사용하여 HTML 파일을 관리합니다.

### 3.4 HTML 파일 작성 (index.html)

public/index.html 파일을 열고 아래와 같이 작성합니다.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>React with Material-UI</title>
</head>
<body>
  <div id="root"></div>
</body>
</html>
```

이 HTML 파일은 React 애플리케이션이 렌더링될 기본 HTML 구조를 정의합니다.

### 3.5 .babelrc 파일 작성

프로젝트 폴더 루트 경로에 .babelrc파일을 작성한다.

```
{  
  "presets": ["@babel/preset-env", "@babel/preset-react"]  
}
```

## 4. 프로젝트 실행

모든 설정이 완료되었으므로, 이제 프로젝트를 실행할 수 있습니다.

```
npm start
```

이 명령어는 Webpack 개발 서버를 시작하고, `http://localhost:3000` 에서 애플리케이션을 확인할 수 있게 해줍니다.

브라우저에서 `http://localhost:3000` 을 열면, "Material-UI 예제"라는 제목과 함께 카드, 그리고 "클릭하세요" 버튼이 표시됩니다. 버튼을 클릭하면 "버튼이 클릭되었습니다!"라는 알림 창이 나타납니다.

# Material-UI 예제

## React와 Material-UI

이 예제는 React와 Material-UI를 사용하여 만들어졌습니다.

클릭하세요

localhost:3000

## Material-UI 예제

### React와 Material-UI

이 예제는 React와 Material-UI를 사용하여 만들어졌습니다.

클릭하세요

localhost:3000 내용:

버튼이 클릭되었습니다!

확인

## 5. 결론

이 튜토리얼에서는 `create-react-app` 을 사용하여 React 프로젝트를 생성하고, Material-UI 컴포넌트를 사용하여 간단한 웹 애플리케이션을 만드는 방법을 배웠습니다. Webpack을 설정하여 애플리케이션을 빌드하고, 버튼 클릭 시 발생하는 이벤트 처리까지 구현했습니다.

Material-UI는 React와 함께 사용하는 강력한 UI 라이브러리로, 다양한 컴포넌트를 활용하여 빠르고 쉽게 멋진 UI를 만들 수 있습니다. 이제 이 튜토리얼을 바탕으로 더 복잡한 컴포넌트와 상호작용을 구현해 보세요!



