

# 프로젝트 시작하기

## Create React App

### Install

```
npx create-react-app project-name
# or
yarn create-react-app project-name
```

이전에 create-react-app을 전역으로 설치한 경우 항상 npx를 통해 최신 버전을 사용하도록 제거하는 것이 좋습니다.

```
warning "react-scripts > @typescript-eslint/eslint-plugin > tsutils@3.17.1" has unmet peer dependency "typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >= 3.6.0-dev || >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta".
success Uninstalled packages.
Done in 4.94s.

Success! Created sample-project at D:\Development\WPXD-FED\sample-project
Inside that directory, you can run several commands:

  yarn start
    Starts the development server.

  yarn build
    Bundles the app into static files for production.

  yarn test
    Starts the test runner.

  yarn eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd sample-project
  yarn start

Happy hacking!
```

프로젝트 생성이 완료되었습니다. 끝났어요. 참 쉽죠?

```
yarn start
  Starts the development server.
  개발모드에서 앱을 실행합니다. (http://localhost:3000/)
  코드를 수정하면 페이지가 자동으로 새로고침 됩니다.

yarn build
  Bundles the app into static files for production.
  build 폴더에 프로덕션용 앱을 빌드합니다.
  파일 이름에 해시가 포함됩니다.
```

```
yarn test
  Starts the test runner.

yarn eject
  Removes this tool and copies build dependencies, configuration files
  and scripts into the app directory. If you do this, you can't go back!
  빌드 도구 및 구성 선택을 변경하고자 하면 eject를 실행 할 수 있습니다.
  한번 eject 를 실행하면 초기의 상태로 돌아갈 수 있는 방법은 없어요.
```

이렇게 생성된 프로젝트는 모든 최신 브라우저를 지원합니다.

IE9~IE11 과 같은 브라우저는 최신 브라우저가 아니에요. 그래서 지원하지 않아요.  
(F\*\*king IE...!!)

구형 브라우저를 지원하기 위해 polyfill을 사용해야 합니다.

```
npm install react-app-polyfill
# or
yarn add react-app-polyfill
```

```
// src/index.js의 제일 위에 선언합니다.
import 'react-app-polyfill/ie9';
```

```
// src/index.js의 제일 위에 선언합니다.
import 'react-app-polyfill/ie11';
```

지원하려는 브라우저의 최소 버전을 `import` 합니다.

IE9를 가져오는 경우 IE10, IE11 지원이 포함됩니다.

- Promise 지원
- window.fetch
- Object.assign
- Symbol
- Array.from

## react-hot-loader

```
npm install react-hot-loader
# or
yarn add react-hot-loader
```

```
import { hot } from 'react-hot-loader';
...
...
export default hot(module)(Component);
```

## 개발시 HTTPS 사용

환경변수 설정으로 개발서버에서 https를 사용할 수 있습니다.

- Windows (cmd.exe)

```
set HTTPS=true&npm start
```

- Linux, macOS (Bash)

```
HTTPS=true npm start
```

## TypeScript Project Install

```
npx create-react-app my-app --template typescript
# or
yarn create react-app my-app --template typescript
```

참고 : `--template typescript` 를 추가하면 타입스크립트를 기반으로 한 프로젝트가 생성됩니다.