

## 9장 프론트엔드 프레임워크와 타입스크립트

### 리액트(React)

리액트와 타입스크립트의 호환성은 비교적 좋은 편입니다. 최근 들어 많은 개발자들이 리액트와 타입스크립트 조합을 선택하고 있습니다. 리액트 공식 홈페이지에서는 타입스크립트를 사용하기 위한 가이드를 제시하고 있습니다. 페이스북에서 공식적으로 배포하는 리액트 웹 개발용 보일러 플레이트(Boilerplate, 구조·설정 작업을 자동으로 진행해주는 도구이자 명령어)인 Create React App은 간단한 옵션 추가만으로 타입스크립트를 사용할 수 있도록 지원합니다.

### 뷰(Vue.js)

뷰 2.0에서는 타입스크립트를 사용할 수 있지만 몇몇 라이브러리의 도움을 받아야 하거나 구현 자체가 안 되는 문제도 다수 있었습니다. 다행스럽게도 최근 릴리즈된 뷰 3.0부터는 타입스크립트를 공식 지원합니다. 뷰 3.0 CLI(Command Line Interface)는 타입스크립트 도구화 지원을 기본으로 제공합니다.

### 앵귤러(Angular)

자바스크립트 기반인 앵귤러 버전 1과 달리 버전 2부터는 타입스크립트 기반으로 만들어졌고 타입스크립트를 권장하고 있습니다.

## 타입스크립트로 어떻게 전환하면 좋을까?

자바스크립트 프로젝트를 타입스크립트로 어떻게 전환할 수 있을까요? 유효한 자바스크립트는 유효한 타입스크립트이며 타입스크립트는 모든 타입을 명시적으로 선언하지 않아도 최선을 다해 타입을 추론하려고 합니다. 이러한 사실은 자바스크립트를 타입스크립트로 점진적, 단계적으로 전환할 수 있음을 의미합니다.

### 1. 비주얼 스튜디오 코드(VSCode)로 자바스크립트 페어링

비주얼 스튜디오 코드에는 편집기 또는 특정 작업 영역(프로젝트)에 대해 자바스크립트에서 타입스크립트 검사를 활성화 할 수 있는 설정이 포함되어 있습니다. `.vscode/setting.json`에 다음 코드를 추가할 수 있습니다.

```
{
  "JavaScript.projectConfig.checkJs": true
}
```

이제 타입이 안전하지 않은 줄에는 에디터에서 해당 코드를 에러로 표시합니다. 이러한 에러 메시지는 에디터에만 표시되며 코드에는 영향을 주지 않습니다.

@ts-ignore - 해당 줄

@ts-nocheck - 전체

해당 코드를 추가하여 타입 체크를 선택적으로 비활성화 할 수도 있습니다.

## 2. 자바스크립트용 타입스크립트 컴파일러 사용

타입스크립트 컴파일러를 설치하고 스크립트를 실행하거나 설정합니다. tsconfig.json 파일을 사용하여 컴파일러 설정을 세팅합니다.

```
{
  "compilerOptions": {
    "allowJs": true
    "checkJs": false
    "outDir": "./dist"
    "rootDir": "./src"
    "strict": false
  }
}
```

"allowJs": true //타입스크립트 컴파일러를 통해 실행할 자바스크립트 파일을 선택

"checkJs": false //자바스크립트 파일에 대한 타입 체크

타입스크립트 컴파일러를 실행하고 자바스크립트를 확인합니다.

"checkJs": true 타입 체크를 활성화하고 타입 체크를 진행합니다. (node.js 경우에는 @types/node를 설치합니다.)

## 3. 자바스크립트 파일을 타입스크립트 파일로 변환

자바스크립트에서 타입이 안전한 파일인지 확인한 후 하나씩 타입스크립트 파일로 변환합니다. (.js → .ts) allowJS를 활성화 했으므로 한번에 모두 이동할 필요 없이 점진적으로 이동할 수 있습니다. 이미 자바스크립트 코드에 대한 타입 체크를 실행했으므로 업데이트가 너무 많지 않을 것입니다.

## 4. 엄격한 타입 체크

타입 체크를 더 엄격하게 하는 몇 가지 옵션이 있습니다.

```
{
  "compilerOptions": {
    "noImplicitAny": true,
    "strictNullChecks": true,
    "strictFunctionTypes": true,
```

```
"strictBindCallApply": true,  
"strictPropertyInitialization": true,  
"noImplicitThis": true,  
"alwaysStrict": true  
}
```

하나씩 활성화 해 문제를 해결합니다.