

Vim: 유행을 타지 않는 명품 에디터

김재호

2022.10.07

Abstract

대 동기화의 시대가 찾아왔다. 어디서든 같은 환경을 만들도록 해주는 동기화는 개발자들의 에디터 설정 방식 역시 바꾸었다. VS Code의 깃허브 계정 연동을 통한 설정 및 추가기능 동기화가 대표적인데, 이러한 VS Code도 GUI 기반 에디터이기 때문에 그래픽 처리가 가능한 로컬 기기마다 따로 설치 및 연동을 해줘야 한다는 한계가 존재한다. 이 과정이 불편하다고 느끼는 사람들에게 vim을 추천한다.

바야흐로, 대 동기화 시대가 찾아왔다. 스마트폰 유저들은 계정 연동만으로 어느 기기에서든 사진 보관함을 동기화할 수 있게 되었다. PC에 있는 파일을 실시간으로 동기화해주는 드랍박스 등의 서비스 이용자도 빠르게 늘어가고 있다. 개발자들은 더 나아가, 본인들이 개발하는 프로젝트의 구조부터 진행과정까지 빠르게 반영할 수 있는 깃, 깃허브 등의 시스템을 적극적으로 사용하고 있다.

동기화의 마법은 개발자들의 에디터에도 영향을 줬다. VS Code는 깃허브 계정 로그인을 통해 깃허브 저장소 연동기능부터 코파일럿과 같은 추가기능까지 사용할 수 있게 해주며, 특히 설정을 계정에 종속시켜 새로운 기기에서 VS Code를 설치해도 로그인만으로 기존 설정을 불러올 수 있다. 언제 어디서나 한결같은 환경에서 개발하고 싶은 이들에겐 필수불가결한 요소일 것이다.

그러나 VS Code에는 근원적인 한계가 있다. GUI 기반 에디터라는 점이다. 물론 GUI는 직관적인 사용자 경험을 제공해줄 수 있다는 강점이 있지만, 틀이 무거워지고 느려지며 그래픽 처리가 가능한 기기에서만 사용할 수 있다는 단점이 있다. 만약 여러 로컬 기기에서 한 서버에 접속해 작업할 일이 있다면, 서버에 직접 VS Code를 설치할 수는 없기 때문에 각 로컬 기기에서 설치부터 연동, 서버 설정까지 반복해야 할 것이다.

이 과정이 불편하고 무가치하다고 느껴지는 사람들에게 vim을 추천한다. Vim은 텍스트 기반 에디터로 터미널을 통해 셸을 실행시킬 수 있는 환경이라면 어디서든 사용할 수 있다. 주로 작업하는 서버에 설정해 두면, 어디서든 같은 환경에서 작업이 가능하다는 것이다. 새로운 서버에서 vim을 새로 세팅할 일이 있더라도, 몇가지 파일만 옮기면 되기 때문에 깃허브에 자신의 설정 파일을 올려두고 사용하는 사람도 많다. 지금 당장 vim을 체험해 보고 싶다면, 주변의 vim 사용자를 찾아가 설정 파일을 받아보도록 하자. 신세계가 열릴 것이다.