

형상 관리 시스템 (Git)

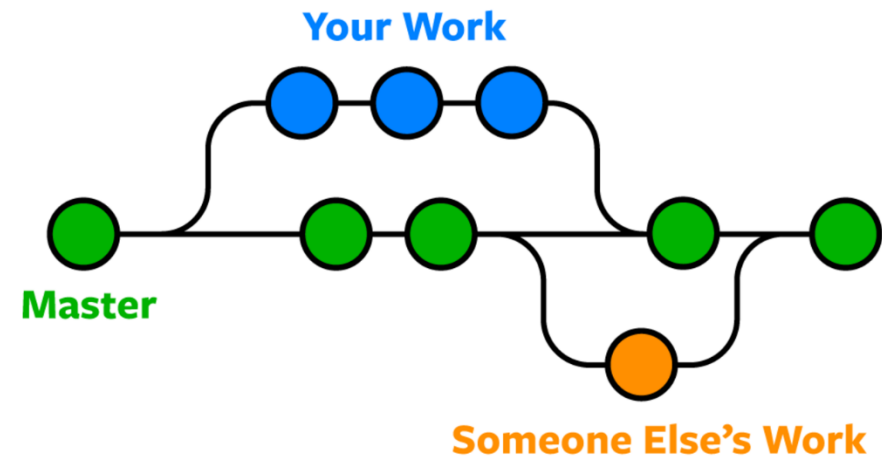
▶ 형상 관리란?

형상 관리

- 소프트웨어의 변경 사항을 체계적으로 추적하고 통제하는 것.

형상 관리 시스템

- 개발 중 발생하는 모든 산출물들이 변경됨으로써 점차 변해가는 소프트웨어 형상을 체계적으로 관리하고 유지하는 시스템.
- 개발 산출물들의 버전 관리와 변경 내역 조회 가능.





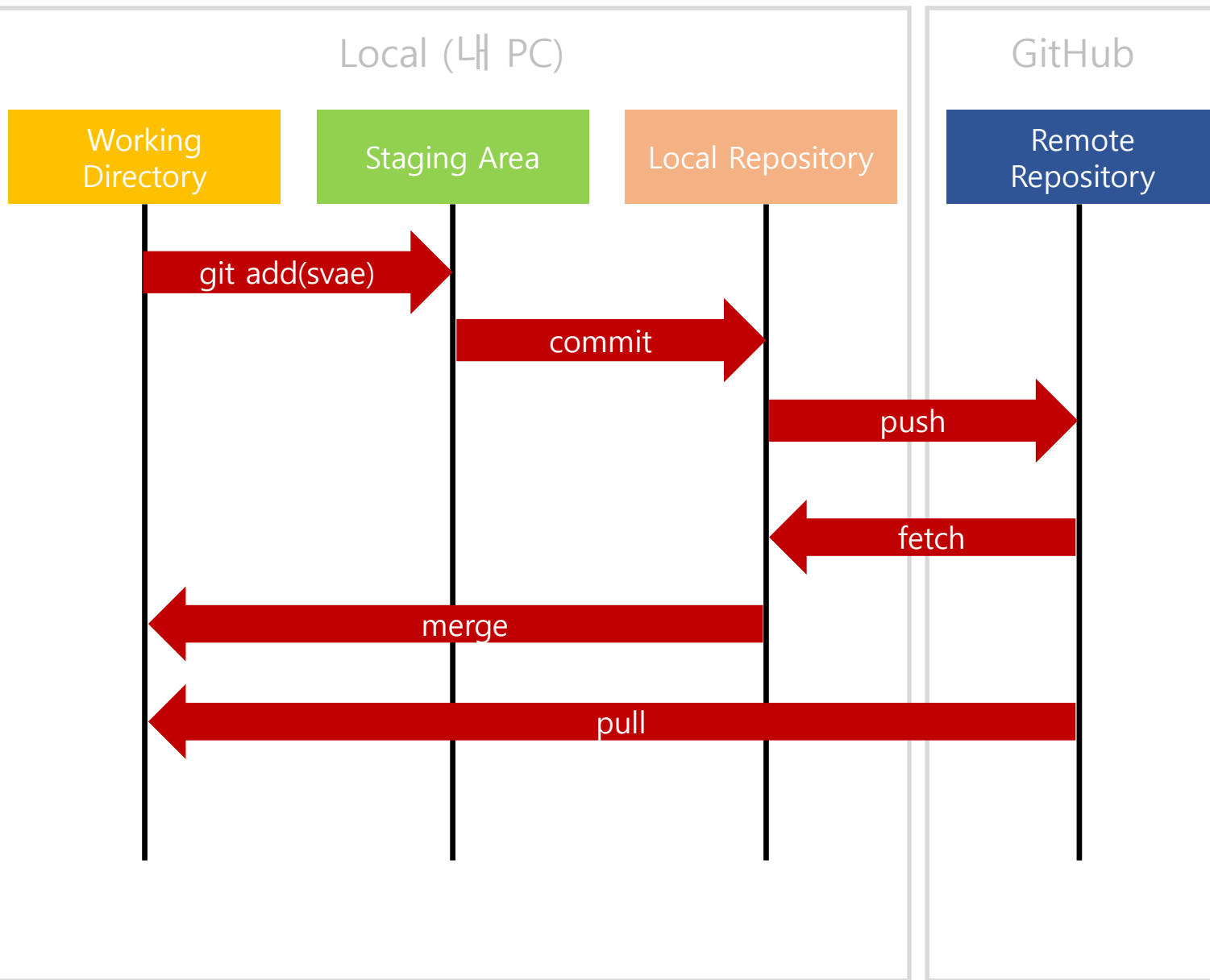
- 대표적인 형상 관리 툴
- 프로젝트(소스 코드)를 저장하고 이력을 관리하는 저장소(repository)
- 원하는 시점에 소스 코드를 저장하거나 저장 지점으로 돌아가는 기능을 제공하여 프로젝트의 버전을 관리할 수 있음.
- 여러 PC에 소스 코드를 저장시키거나 여러 PC에서 작성한 내용을 하나로 병합(merge)하는 것이 가능함.

▶ GitHub



- 코드 저장소 Git을 웹으로 이용할 수 있게 만든 원격 저장소(remote repository)
- 인터넷이 제공 되는 환경에서 소스 코드의 공유 및 버전 관리가 가능.
- 여러 개발자가 하나의 원격 저장소에 있는 소스 코드를 받거나 변경한 코드를 병합하는 것이 가능. → 협업 가능
- <https://github.com/>

▶ 저장소(Repository) 종류와 관련 용어



Working Directory : 작업이 진행되는 폴더

- 프로젝트 폴더를 의미하며, 소스코드 변경 후 **git add**(그냥 **save**, ctrl + s) 시 Staging Area로 전달됨.

- Local Repository에 저장된 소스코드를 현재 프로젝트에 병합(**merge**) 가능.

Staging Area : 준비 영역, 중간 영역

- 변경된 코드가 Local Repository에 저장하기 전 머무르는 중간 영역.
- Local Repository에 저장될 코드를 선택하고 **commit**을 통해 저장할 수 있음.

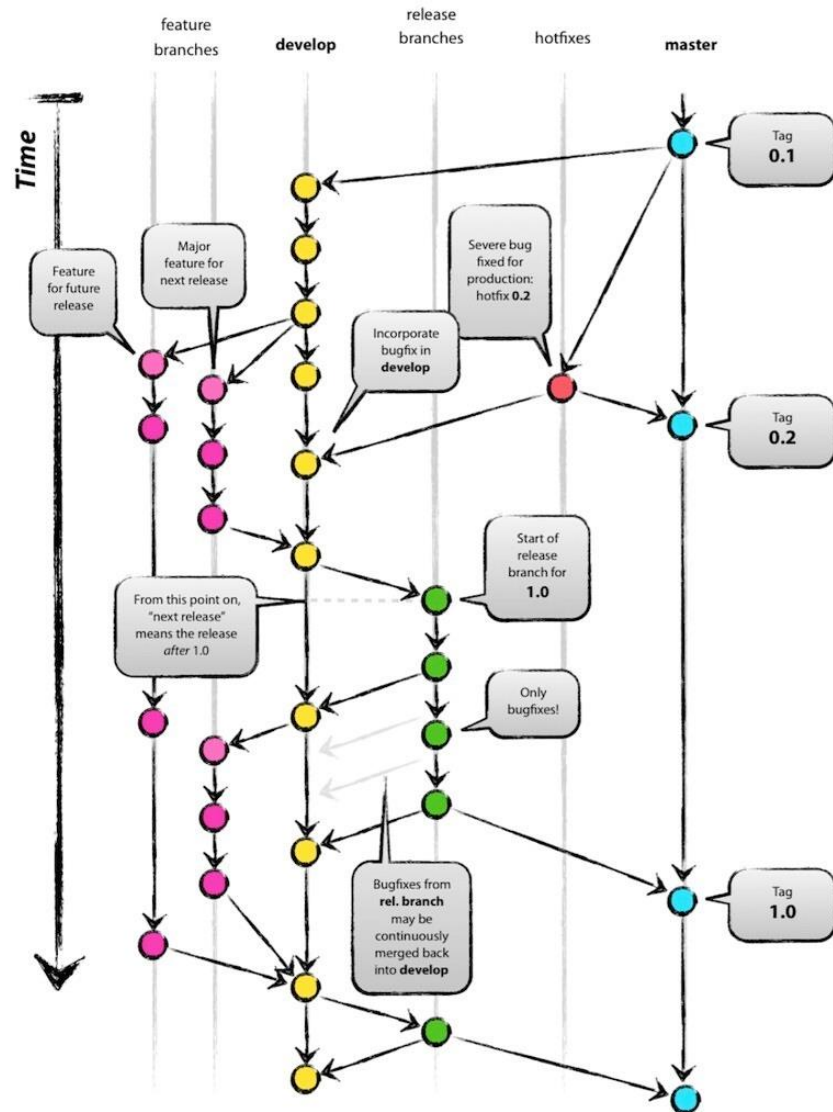
Local Repository : 내 PC 내에 있는 저장소

- 소스 코드의 추가/변경 사항을 기록하는 저장소.
- Local Repository에 저장된 내용은 Remote Repository에 **push**를 반영할 수 있음.

- Remote Repository에 저장된 소스 코드를 **fetch**를 통해 얻어와 Local Repository에 저장 가능.

Remote Repository : 원격 저장소(GitHub)

Branch



Feature Branch Model

- **하나의 저장소(Repository)** 내에 여러 버전을 만들어 작업할 수 있는 방법.
- 독립적으로 작업을 진행하기 위해 저장소를 분리하는 개념.
* branch : 나뉘어가지, 분기, 갈라지다
- 협업 진행 시 원본(master branch) 코드를 각자 분리된 작업 영역(branch)으로 가져가 서로 다른 작업을 진행.
- 각자 branch로 가져간 코드는 변경되어도 원본에 영향을 미치지 않음.
- 코드가 변경된 branch의 내용을 master branch에 적용해달라는 요청(Pull Request)을 보낼 수 있음.
- 요청이 받아 들여지면 master branch에 변경한 코드 내용이 병합됨.