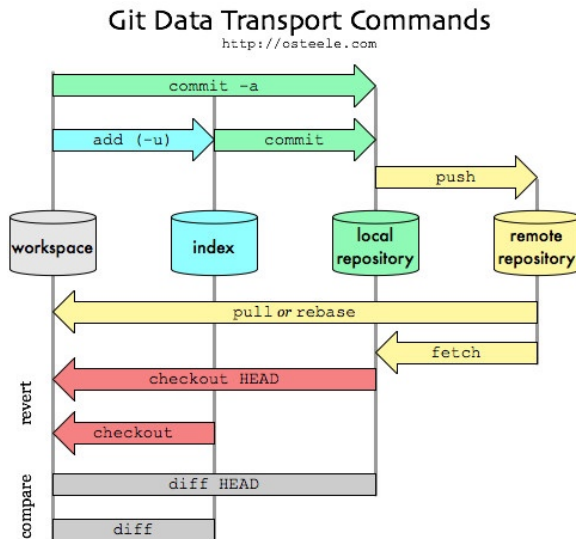


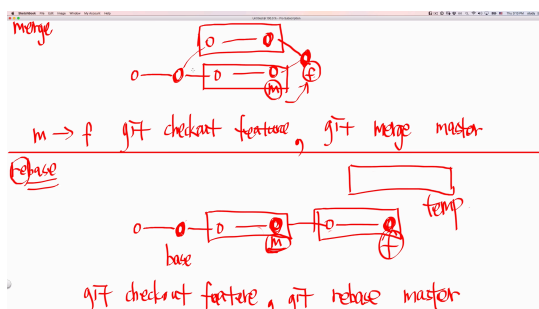
- git의 원리
 - git의 object는 내용을 sha1 알고리즘을 적용시켜 나온 결과에 대하여 앞 두 자리를 폴더 명으로, 이후를 id로 하여 저장함
- git의 object의 타입은 3개
 - 파일의 내용을 담고 있는 blob, 디렉토리의 파일명들과 그 파일의 내용에 해당하는blob의 정보를 담고 있는 tree, 마지막으로 commit
- git은 working directory <-> stage(index, cache <-> repository 사이의 관계를 통해 파일의 상태를 판단한다.
 - 파일을 add 하면 index에 추가되며, commit 하면 repo에 추가되는 식



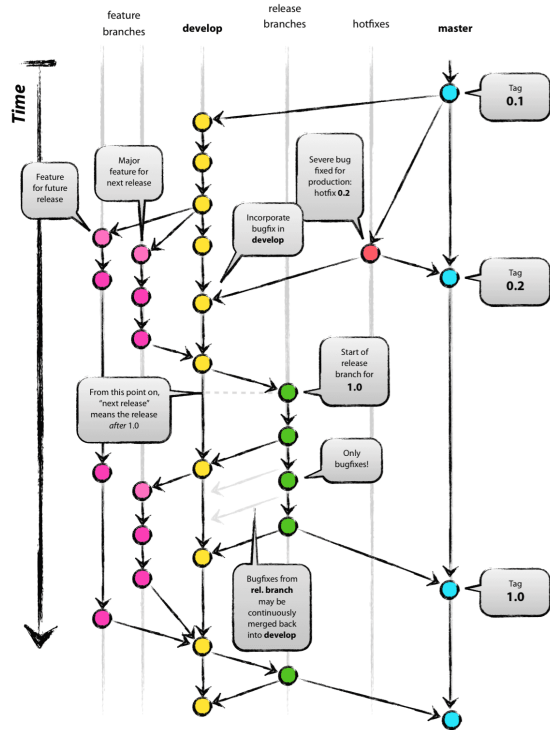
- git stash 는 브랜치 내에서 작업 중 다른 브랜치로 가야할 때(그냥 가면 수정중인 파일이 그대로 넘어감), 현재 상태를 임시 저장해두고 직전 커밋으로 돌리는 명령(단, 버전 관리 중인 파일에만 적용)
 - git stash apply 하면 돌아옴
- git reset 시 적용되는 범위는 다음과 같음

working directory	index	repository
working tree	staging area	history tree
working copy	cache	
		git reset --soft
		git reset --mixed
		git reset --hard

- git pull과 fetch의 차이는, pull은 origin의 상태를 최신으로 업데이트하며 동시에 최신 커밋을 로컬저장소에 merge하지만, fetch는 origin의 상태만을 업데이트함. 따라서 이후에 직접 merge해야함
- merge와 rebase는 다음과 같은 차이가 있음.



- git flow



- develop 브랜치에서 개발 후 master로 merge한다는 개념
- feature 브랜치에서 개발 시 항상 develop 브랜치의 변경 사항들을 pull하면서 진행해야함!