

# 신입 개발자를 위한 Git

가짜머리 김기동

# 문제 1



airforce\_junho.txt

어떤 파일 하나가 있다

# 문제 1



airforce\_junho.txt

이 파일을 수정하는데, 변경 사항을 다 기록하고 싶다

# 문제 1



airforce\_junho\_v0.txt



airforce\_junho\_v1.txt



airforce\_junho\_v2.txt

가장 쉬운 방법은 역시 변경될 때 마다 백업하기

# 문제 1

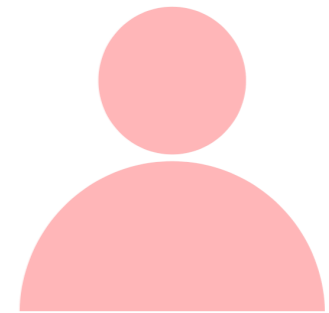


변경 사항이 너무 많아지면...?

## 문제 2

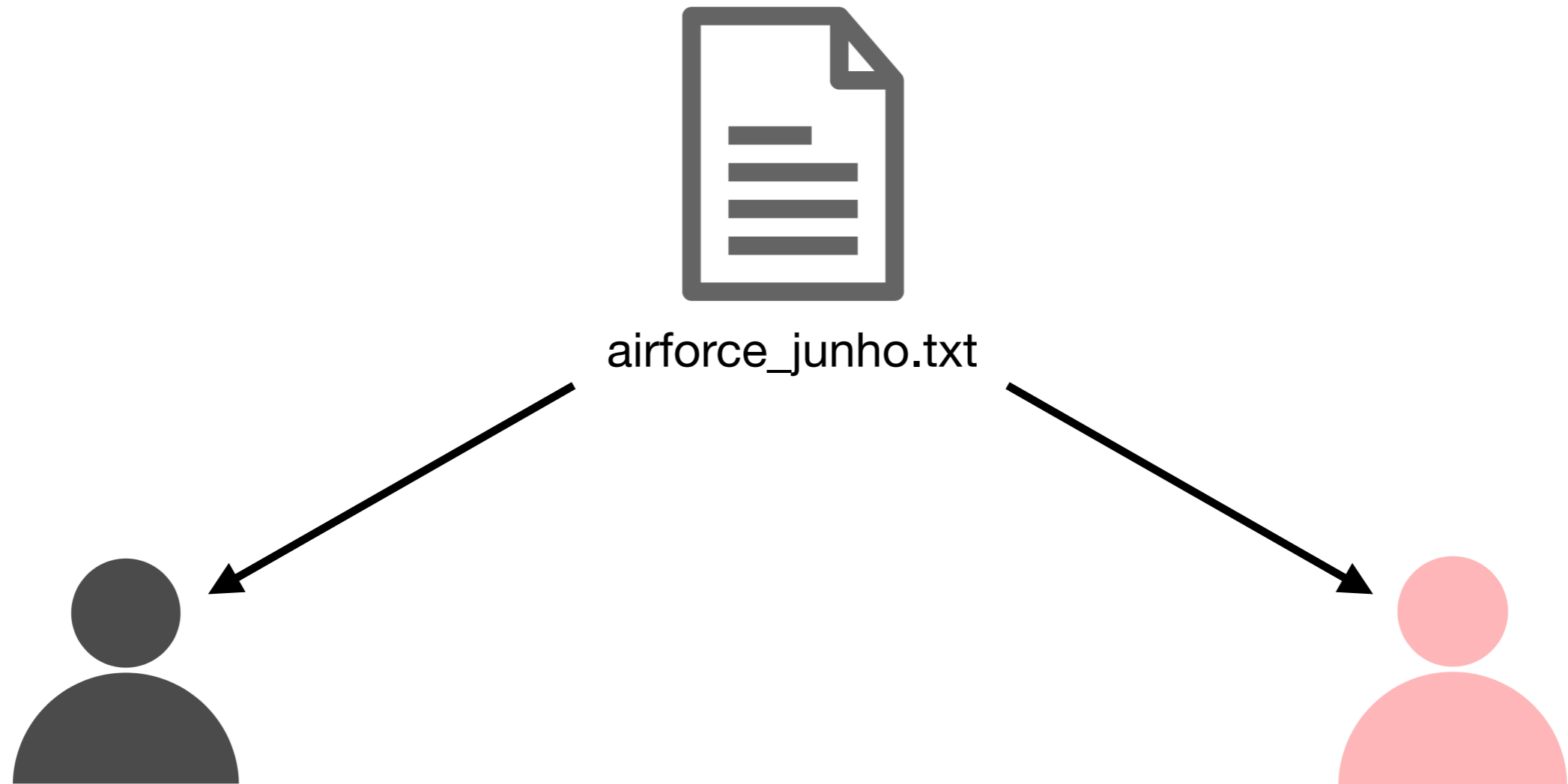


airforce\_junho.txt



어떤 파일 하나가 서버에 있다

## 문제 2



두 사람이 이 파일을 다운로드 받았다

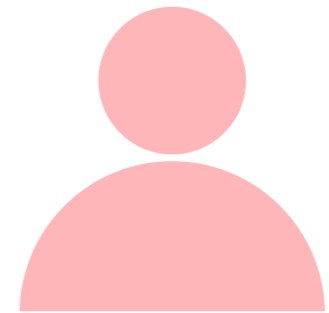
## 문제 2



airforce\_junho.txt



airforce\_junho.txt

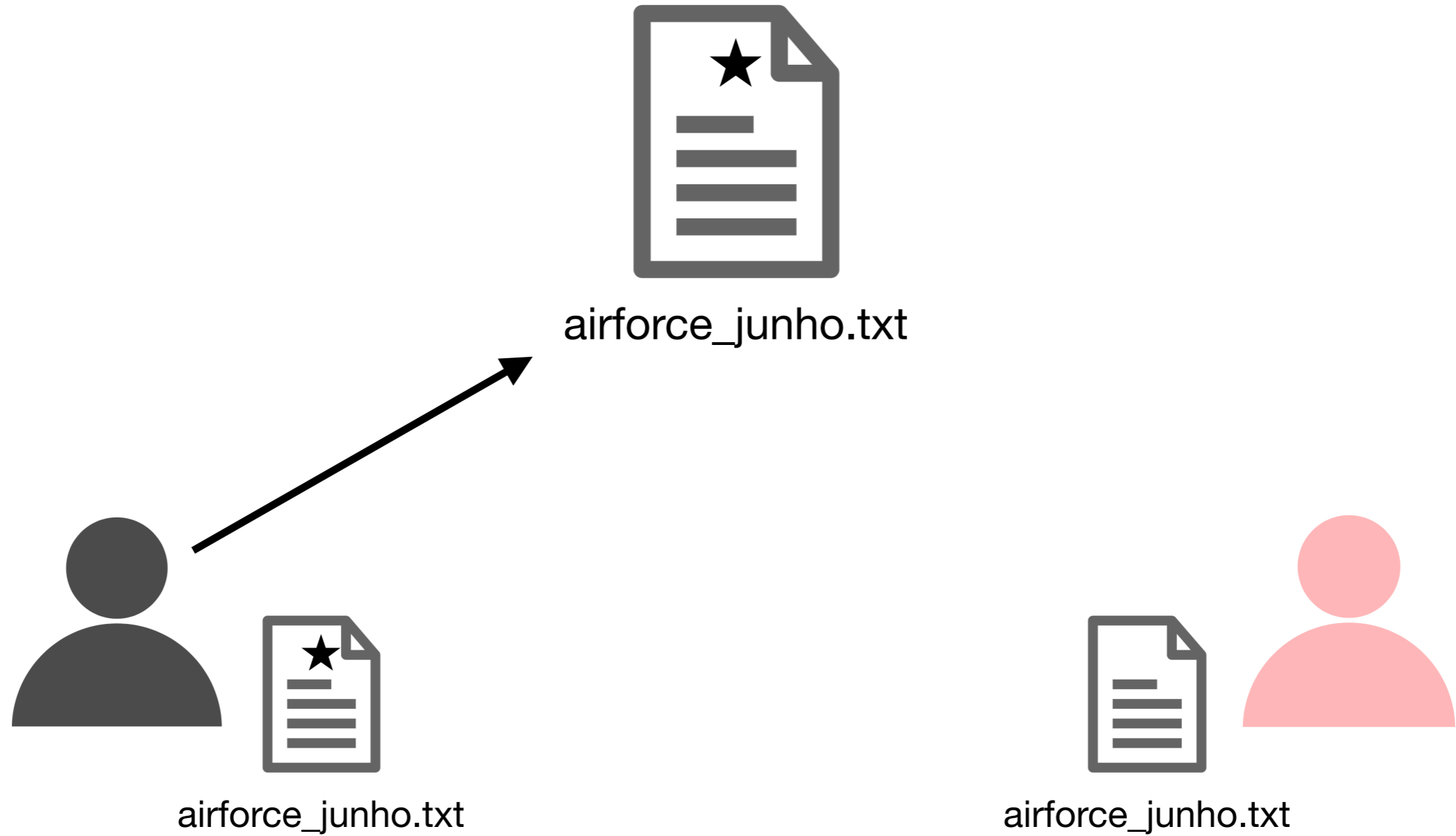


airforce\_junho.txt

두 사람이 이 파일을 다운로드 받았다

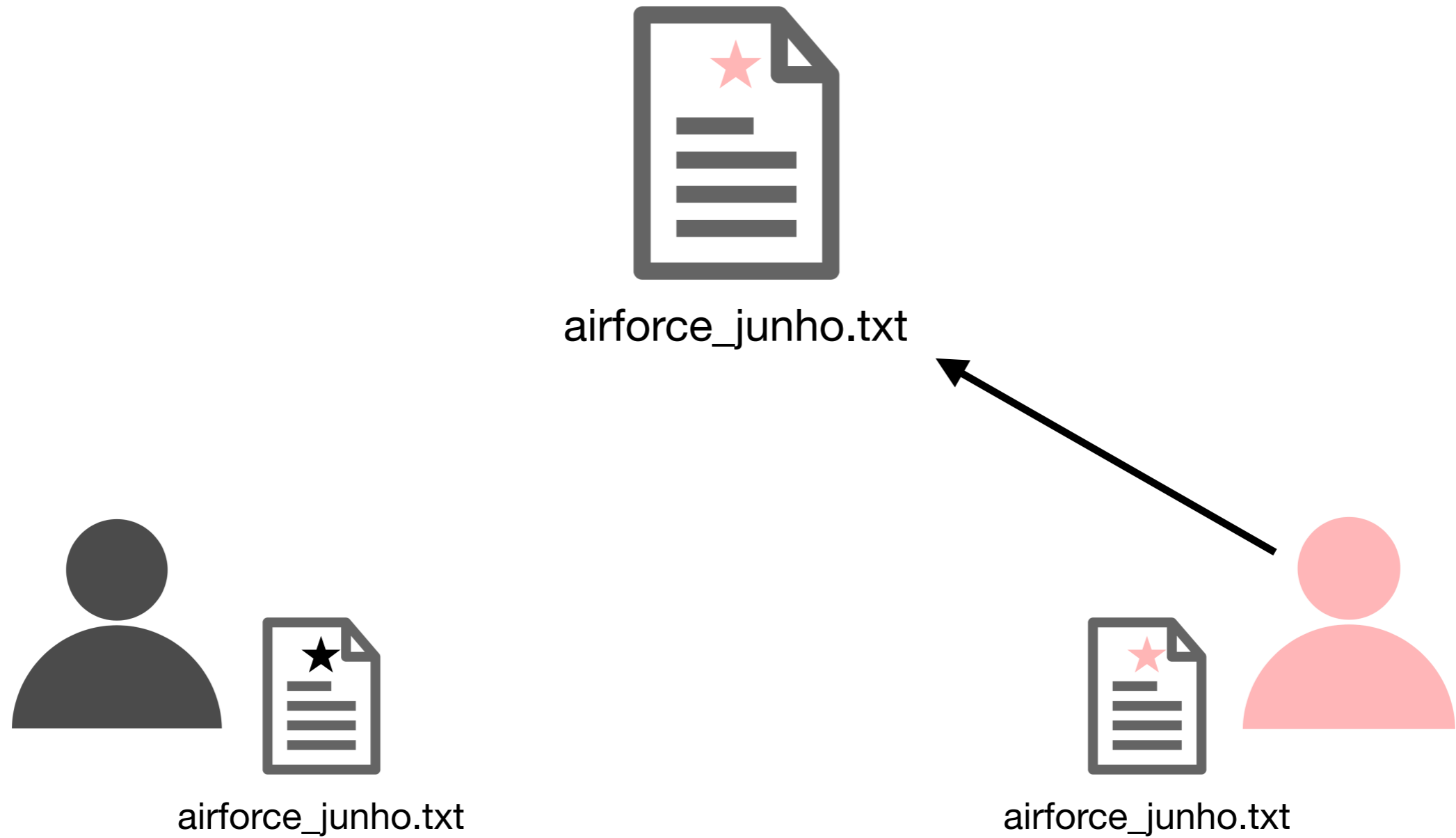


## 문제 2



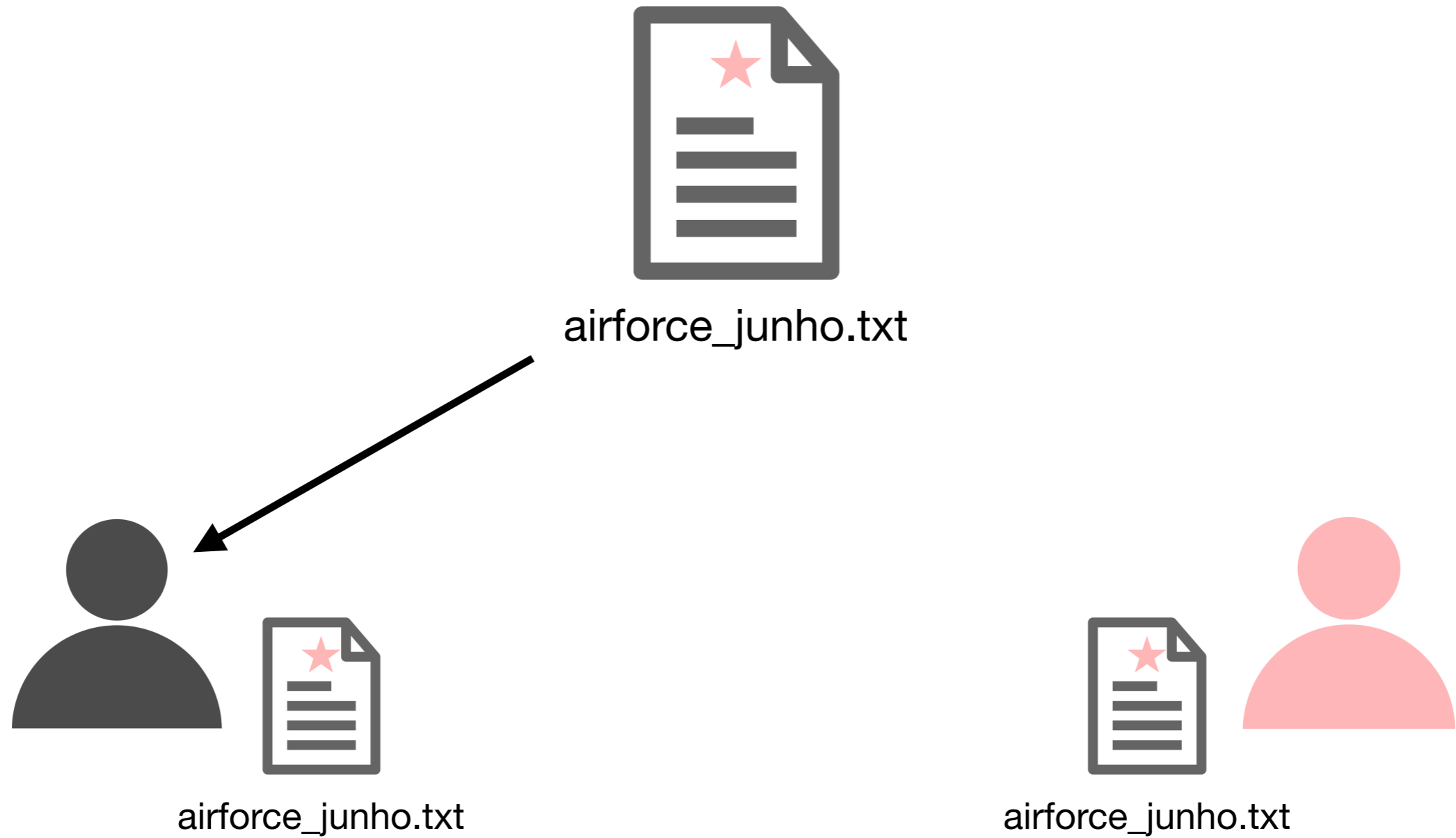
한 사람이 파일을 수정하고, 서버에 업로드한다

## 문제 2



다른 사람도 파일을 수정하고 서버에 업로드한다

## 문제 2



처음에 파일을 수정했던 사람이  
서버에서 파일을 다시 다운로드한다

# 문제 2



내가 수정한거 어디갔노 ㅅㅂ



(Linus Torvalds, 35세)

이런 문제를 해결하기 위해 우리에게 Git이 있습니다

# 기본적인 Git 사용법

# git init

```
~/과기서관코노 ➤ git init  
Initialized empty Git repository in /Users/blur/과기서관코노/.git/
```

▼	📁 .git	오늘 오후 10:23
	📄 config	오늘 오후 10:23
	📄 description	오늘 오후 10:23
	📄 HEAD	오늘 오후 10:23
▶	📁 hooks	오늘 오후 10:23
▶	📁 info	오늘 오후 10:23
▶	📁 objects	오늘 오후 10:23
▶	📁 refs	오늘 오후 10:23





# git status

```
~/과기서관코노 ↗ master ↗ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file> ..." to include in what will be committed)

    naver.txt

nothing added to commit but untracked files present (use "git add" to track)
```

**Untracked file == git이 관리하지 않는 파일**

# git add <파일/폴더 경로>

```
~/과기서관코노 ↗ master > git add naver.txt
~/과기서관코노 ↗ master + > git status

On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file> ..." to unstage)

        new file:   naver.txt
```

git add는 git의 스테이지 영역\*에 현재 파일의 변경사항을 기록한다



# git status

```
~/과기서관코노 > master ●+ > git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file> ..." to unstage)

    new file:   naver.txt

Changes not staged for commit:
  (use "git add <file> ..." to update what will be committed)
  (use "git checkout -- <file> ..." to discard changes in working directory)

    modified:   naver.txt
```

스테이지 영역에 있는 파일과, 내 로컬에서 변경된 파일은 다르다!  
git add 할 때 기준으로 파일의 변경 상태가 저장된다

# git diff

```
blur  master •  less ◀ git ◀ zsh
diff --git a/naver.txt b/naver.txt
index 29a2f05..dc3ca7d 100644
--- a/naver.txt
+++ b/naver.txt
@@ -1,2 @@
 김준호씨? 누구세요?
+from naver
~
(END)
```

git diff 명령어로 어떤 파일이 얼마나 바뀌었는지 알 수 있다

# git add

```
~/과기서관코노 | master + | git add naver.txt
~/과기서관코노 | master + | git status
On branch master
No commits yet
Changes to be committed:
  (use "git rm --cached <file> ..." to unstage)

    new file:   naver.txt
```

다시 `git add` 명령어로 변경 사항을 갱신해주자

# git commit

```
x ~/과기서관코노 ↗ master + git commit -m "[docu] write letter from naver"  
[master (root-commit) bf83732] [docu] write letter from naver  
1 file changed, 2 insertions(+)  
create mode 100644 naver.txt
```

`git commit -m "커밋 메시지"`  
위 명령어로 커밋이라는 것을 할 수 있다

커밋 메시지의 내용은 무슨 작업을 했는지 간략하게 요약  
(누구처럼 커밋 시간 썼다가 나중에 개삽질하지 마셈)

# git push

```
~/과기서관코노 > ↗ master > git push  
fatal: No configured push destination.  
Either specify the URL from the command-line or configure a remote repository using  
  
    git remote add <name> <url>  
  
and then push using the remote name  
  
git push <name>
```

**git push**로 원격 레포지터리에 변경 사항을 알려줘야 하는데...

**원격 레포지터리가 뭐죠?**



# GitHub





GitLab



Bitbucket

## Quick setup — if you've done this kind of thing before

 Set up in Desktop or HTTPS SSH `git@github.com:blurfx/oh-my-cono.git` 

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).


## ...or create a new repository on the command line

```
echo "# oh-my-cono" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:blurfx/oh-my-cono.git
git push -u origin master
```





## ...or push an existing repository from the command line

```
git remote add origin git@github.com:blurfx/oh-my-cono.git
git push -u origin master
```




## Quick setup — if you've done this kind of thing before

 Set up in Desktop or HTTPS SSH `git@github.com:blurfx/oh-my-cono.git` 

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).


## ...or create a new repository on the command line

```
echo "# oh-my-cono" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:blurfx/oh-my-cono.git
git push -u origin master
```



## ...or push an existing repository from the command line

```
git remote add origin git@github.com:blurfx/oh-my-cono.git
git push -u origin master
```



# git push

```
x ~/과기서관코노 ↳ master git remote add origin git@github.com:blurfx/oh-my-cono.git
~/과기서관코노 ↳ master git push -u origin master
Warning: Permanently added the RSA host key for IP address '15.164.81.167' to the list of known hosts
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 268 bytes | 134.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:blurfx/oh-my-cono.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

**git remote add origin <git 주소>**

**git push -u origin master**

 blurfx / **oh-my-cono** Private

 Code

 Issues **0**

 Pull requests **0**

 Actions

cono should join the army 😂

[Manage topics](#)


 **1** commit

 **1** branch

Branch: **master** ▾

[New pull request](#)

 **blurfx** [docu] write letter from naver

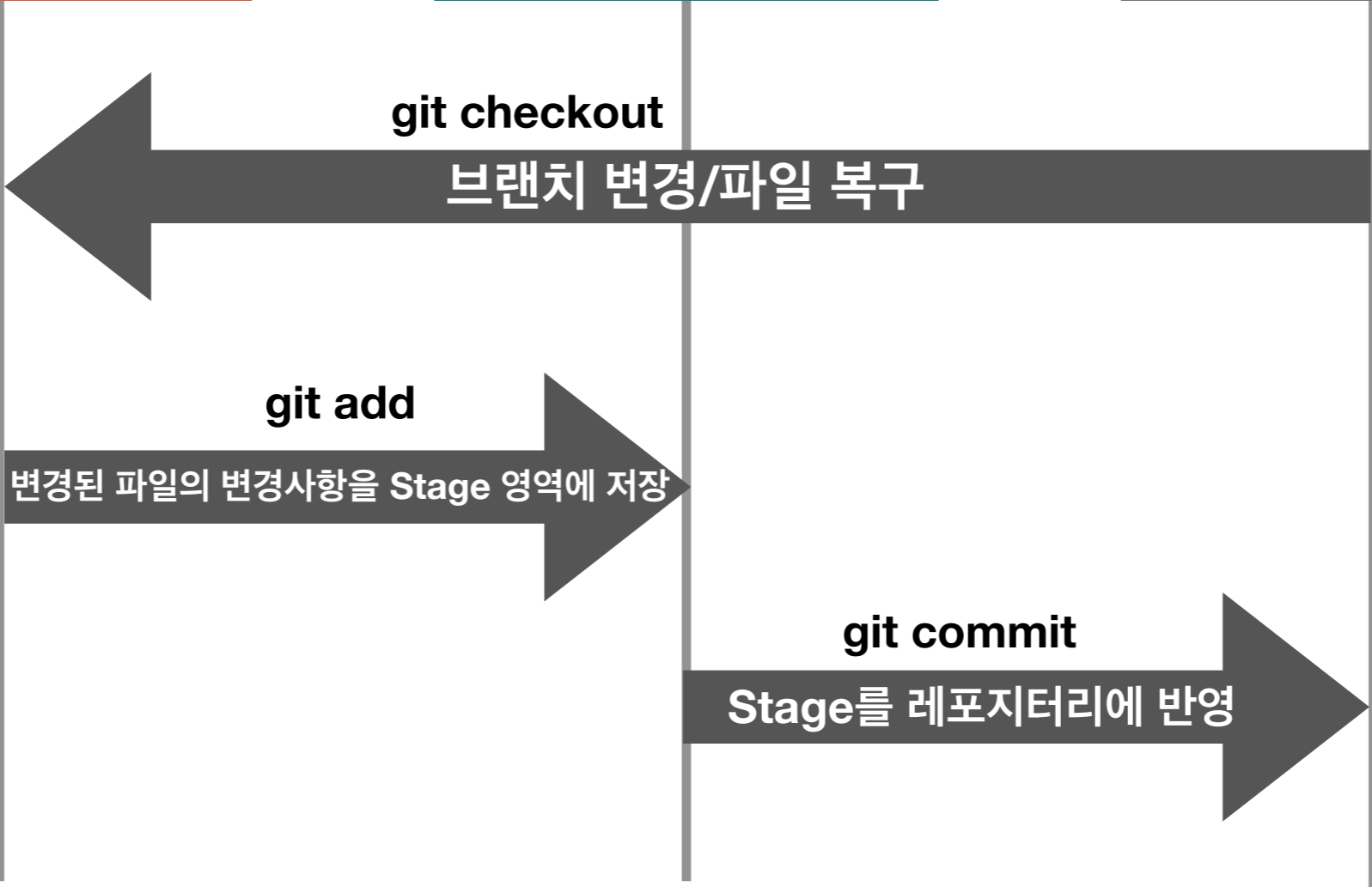
 [naver.txt](#)

[docu] write letter from naver

로컬 프로젝트 폴더  
(Working Tree)

Git 스테이지 영역  
(Staging Area)

로컬 레포지터리  
(.git 폴더)





### Untracked

아직 레포지터리에 이 파일이 추가가 되지 않음  
-> 파일이 변경되어도 git은 신경 안쓴다



### Tracked

Git이 이 파일의 변경 사항을 추적함

회사에 입사하고 새로운 기능을 추가해달라는 일을 받았는데



회사에 입사하고 새로운 기능을 추가해달라는 일을 받았는데  
master에 커밋하면 바로 서비스에 올라간대

회사에 입사하고 새로운 기능을 추가해달라는 일을 받았는데

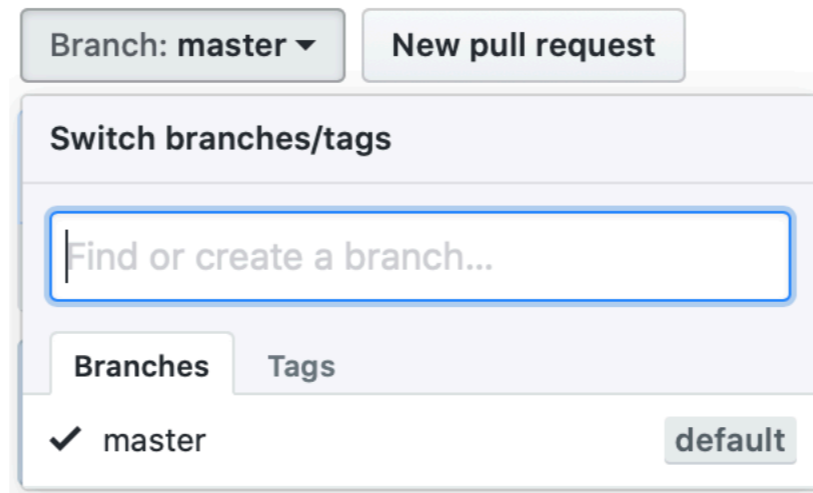
master에 커밋하면 바로 서비스에 올라간대

근데 이거 잘 되는지 테스트도 해야하고 코드 리뷰도 받아야하는데

어떻게 해야하지?

# 브랜치로 작업 분리하기

# 브랜치?



코드 수정을 동시에 독립적으로 작업 할 수 있는 기능

# git branch

**git branch** <브랜치 이름>으로 새 브랜치를 만들고

```
git branch new-awesome-feature
```

**git branch**로 로컬에 있는 브랜치 목록을 볼 수 있다

```
* master  
  new-awesome-feature  
(END)
```

# git checkout

**git checkout** <브랜치 이름>으로 다른 브랜치로 이동할 수 있다

```
~/과기서관코노 > ↗ master > git checkout new-awesome-feature  
Switched to branch 'new-awesome-feature'  
~/과기서관코노 > ↗ new-awesome-feature > |
```

# git add, commit

새 파일 hello.c를 만들고, stage로 옮기고(add) 커밋(commit)하기

```
~/과기서관코노 > ⌵ new-awesome-feature > touch hello.c
~/과기서관코노 > ⌵ new-awesome-feature > git status
On branch new-awesome-feature
Untracked files:
  (use "git add <file> ..." to include in what will be committed)

    hello.c

nothing added to commit but untracked files present (use "git add" to track)
~/과기서관코노 > ⌵ new-awesome-feature > git add hello.c
~/과기서관코노 > ⌵ new-awesome-feature + > git commit -m "Add hello.c"
[new-awesome-feature 4b2f0f8] Add hello.c
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 hello.c
```

과기사관코노 (Git)

커밋 풀 푸시 페치 브랜치 병합 스테시 원격 보기 파인더에서 보기 터미널 설정

모든 브랜치 원격 브랜치 표시 원형 순 건너뛰기:

그래프	설명	커밋	작성자	날짜	
	feature/cono master 2개 앞 김준호 뽀뽀이	f16fb07	blurfx <its.blur...	오늘 오전 12:27	+
	Add hello.c	94b6772	blurfx <its.bluref...	오늘 오전 12:24	
	new-awesome-feature Add hello.c	4b2f0f8	blurfx <its.bluref...	오늘 오전 12:15	
	origin/master [docu] write letter from naver	bf83732	blurfx <its.bluref...	어제 오후 10:51	

경로 이름순 정렬

hello.c hello.c

Add hello.c

제출 ID: 4b2f0f8e59e3c7acb57f0e3f7086e64706113f84 [4b2f0f8e59e3c7acb57f0e3f7086e64706113f84]  
 상위 항목: bf837323a6  
 작성자: blurfx <its.blureffect@gmail.com>  
 날짜: 2019년 12월 28일 오전 12시 15분 14초 GMT+9  
 레이블: new-awesome-feature



# git checkout

```
~/과기서관코노 > ↪ new-awesome-feature > ll
total 8
-rw-r--r--  1 blur  staff   0 12 28 00:14 hello.c
-rw-r--r--  1 blur  staff  39 12 27 22:42 naver.txt
~/과기서관코노 > ↪ new-awesome-feature > git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
~/과기서관코노 > ↪ master > ll
total 8
-rw-r--r--  1 blur  staff  39 12 27 22:42 naver.txt
```

```
commit bf837323a640004ec86ff26656f17ae48091f12c (HEAD → master, origin/master)
Author: blurfx <its.blureffect@gmail.com>
Date:   Fri Dec 27 22:51:11 2019 +0900

    [docu] write letter from naver
(END)
```

new-awesome-feature 브랜치에는 있던 hello.c 파일이  
master 브랜치에는 커밋도 보이지 않고, 반영되지 않았다

# git merge

```
~/과기서관코노 > ↗ master > git merge new-awesome-feature
Updating bf83732..4b2f0f8
Fast-forward
 hello.c | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 hello.c
```

```
commit 4b2f0f8e59e3c7acb57f0e3f7086e64706113f84 (HEAD → master, new-awesome-feature)
Author: blurfx <its.blureffect@gmail.com>
Date: Sat Dec 28 00:15:14 2019 +0900

    Add hello.c

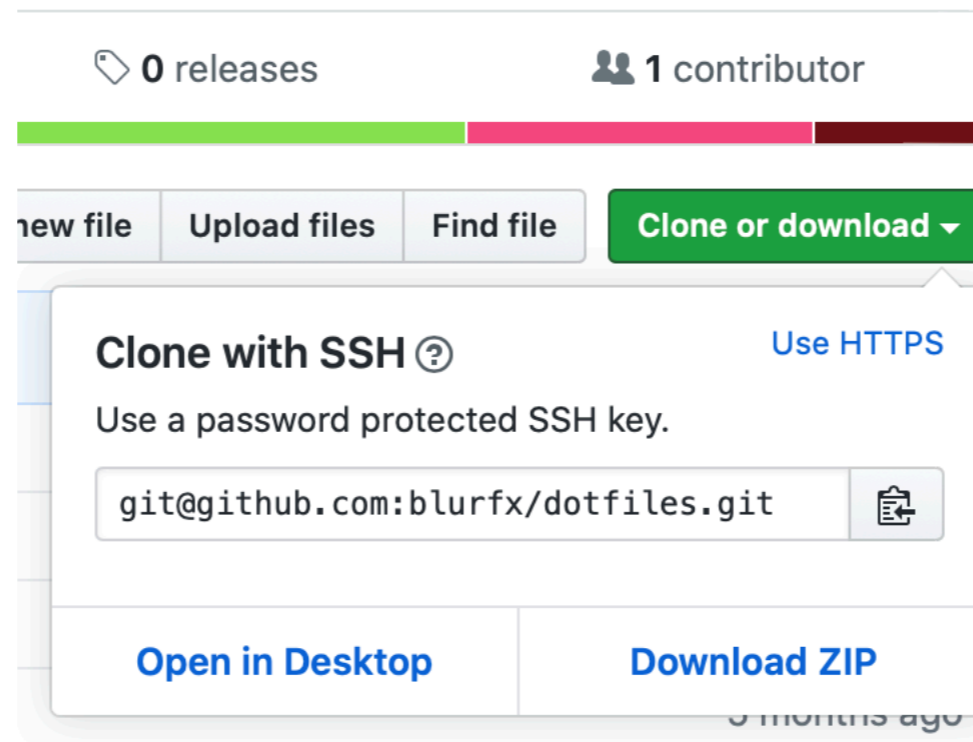
commit bf837323a640004ec86ff26656f17ae48091f12c (origin/master)
Author: blurfx <its.blureffect@gmail.com>
Date: Fri Dec 27 22:51:11 2019 +0900

    [docu] write letter from naver
```

`git merge <브랜치 이름>`으로 다른 브랜치의 변경 사항들을 동기화 할 수 있다

알면 좋은 git 명령어들

# git clone



`git clone` <Git 주소>로 서버에 있는 레포지토리를 가져올 수 있다

# git clone

```
~ ➤ git clone git@github.com:blurfx/dotfiles.git
'dotfiles'에 복제합니다 ... dotfiles and settings
remote: Enumerating objects: 55, done.
remote: Counting objects: 100% (55/55), done.
remote: Compressing objects: 100% (48/48), done.
remote: Total 112 (delta 24), reused 14 (delta 5), pack-reused 57
오브젝트를 받는 중 : 100% (112/112), 25.46 KiB | 137.00 KiB/s, 완료.
델타를 알아내는 중 : 100% (41/41), 완료.
~ ➤ ls | grep dotfiles
dotfiles
```

git clone <Git 주소>로 서버에 있는 레포지토리를 가져올 수 있다

## **git rebase**

이전 커밋을 수정하거나, 여러 커밋을 하나의 커밋으로 합칠때 주로 사용

## **git cherry-pick**

다른 브랜치에 커밋된 내용 중 일부만 가져오고 싶을 때

## **git stash**

지금 작업중인 내용을 다른 곳에 임시로 저장하고 작업 내용을 날리고 싶을 때

## **git revert**

커밋을 되돌리고 싶을 때

**Git w/ GUI**

# SourceTree

The screenshot shows the SourceTree application window for a repository named "sourcetree-website (Git)". The interface includes a toolbar with buttons for Commit, Pull (98), Push, Branch, Merge, and Shelve. On the right side of the toolbar are "Show in Finder", "Terminal", and "Settings".

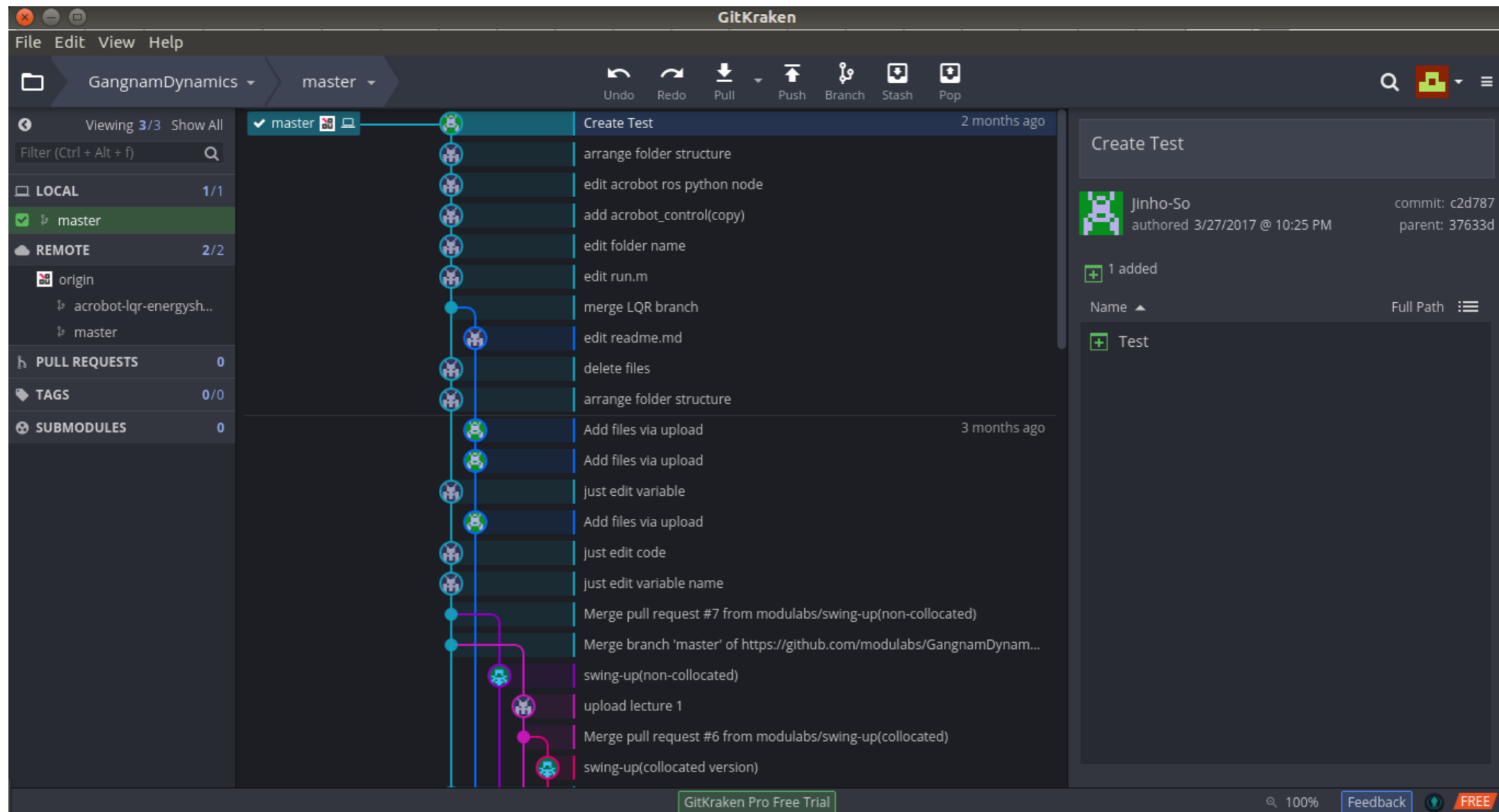
The left sidebar contains a "WORKSPACE" section with "File status" and "History" (highlighted). Below this are "Search", "BRANCHES", "BOOKMARKS", "TAGS", "REMOTES", "SHELVED", and "SUBREPOSITORIES".

The main area displays a commit graph and a table of commit details. The table has columns for Graph, Commit, Author, Description, and Date. The top commit is highlighted in blue.

Graph	Commit	Author	Description	Date
	b7358c7	Rahul Chha...	Removing ol... master origin/master origin/HEAD	Mar 3, 2016, 11:...
	bdb8bef	Rahul Chhab...	Merged in update-google-verification (pull request #14)	Feb 18, 2016, 1:3...
	dfe975d	Tyler Tadej...	origin/update-google-verification Update google verificati...	Feb 11, 2016, 2:2...
	3bc3290	Tyler Tadej...	Replace outdated Atlassian logo in footer with base-64 en...	Feb 11, 2016, 2:1...
	dba47f9	Tyler Tadej...	Add gitignore	Feb 11, 2016, 1:3...
	ff67b45	Mike Minns...	Updated Mac min-spec to 10.10	Feb 15, 2016, 11:...
	72d32a8	Michael Min...	Merged in hero_images (pull request #13)	Feb 15, 2016, 10:...
	246c4ff	Joel Unger...	origin/hero_images hero_images Used Tinypng to c...	Feb 11, 2016, 3:3...
	9d9438c	Joel Unger...	Replacing hero images with new version of SourceTree	Feb 9, 2016, 2:59...
	ce75b63	Michael Min...	Merged in bug/date-https (pull request #12)	Feb 15, 2016, 10:...
	85367bb	Patrick Tho...	origin/bug/date-https fixed date and https errors	Jan 7, 2016, 12:2...
	4f9b557	Joel Unger...	New Favicon	Feb 8, 2016, 3:55...
	384e6d5	Rahul Chhab...	origin/search-console-access search console google ver...	Feb 3, 2016, 2:09...
	6fa47a9	Mike Minns...	updated to move supported version to OSX 10.9+	Dec 15, 2015, 2:0...
	8dd87bb	Mike Minns...	remove extra , when a line is skipped due to empty server	Nov 23, 2015, 2:2...
	faa195e	Mike Minns...	Skip records with empty server/user id as gas rejects them	Nov 23, 2015, 2:1...
	0cdf96	Mike Minns...	corrected paths after merge	Nov 23, 2015, 2:0...
	051ab1b	Mike Minns...	corrected column counting	Nov 23, 2015, 1:5...
	a723bc2	Mike Minns...	Merge branch 'au2gex'	Nov 23, 2015, 1:5...
	65fd580	Mike Minns...	deal with invalid instanceids	Nov 23, 2015, 1:5...
	500a892	Michael Min...	Merged in au2gex (pull request #11)	Nov 23, 2015, 1:0...



# GitKraken



감사합니다