

# Git

source code management system

( 소스관리 도구 , 실습위주 )

focused on training

Taeung Song

treeze.taeung@gmail.com

2016-03-24

# Speaker

---

**송태웅** (Taeung Song)

- Linux kernel - Contributor 활동 중 (perf)
- NIPA Open Frontier Lab 3 기 | (<http://devlab.oss.kr/>)

# Training mode Git 훈련방식

---

우리가 스마트폰을 ( 메뉴얼 없이 ) 사용하면서 이해하듯

**Git** 이란 프로그램도 일단 써보면서 이해 해보자 ( 실습위주 교육 )

# Contents

---



# Git 이란 간단한 정의

---

Git 개발과정 , 소스파일 등을 관리하는 도구

Histroy 관리가 되어 특정시점으로 복구가능 , 변화되어온 과정을 볼 수 있다 .

# Git 의 기능

Ctrl+c, v 나 Alzip 압축파일 관리법

Source code management **tool**

그 사이에 뭐가 바뀌었는지  
차이 (Diff) 를 알 수 없다.



과제 1\_ 최종 \_2016\_02\_28.zip

과제 1\_ 진짜최종 \_2016\_02\_29.zip

과제 1\_ 진짜진짜최종 \_2016\_03\_01.zip

# Git 의 기능

Ctrl+c, v 나 Alzip 압축파일 관리법

Source code management **tool**

그 사이에 뭐가 바뀌었는지  
차이 (Diff) 를 알 수 없다.

Ctrl + c, v 를 할수록  
차지하는 용량 X 2  
X 3 ... + diff



과제 1\_ 최종 \_2016\_02\_28.zip

과제 1\_ 진짜최종 \_2016\_02\_29.zip

과제 1\_ 진짜진짜최종 \_2016\_03\_01.zip

# Git 의 기능

Ctrl+c, v 나 Alzip 압축파일 관리법

Source code management **tool**

그 사이에 뭐가 바뀌었는지  
차이 (Diff) 를 알 수 없다.

Ctrl + c, v 를 할수록  
차지하는 용량 X 2  
X 3 ... + diff



과제 1\_ 최종 \_2016\_02\_28.zip

과제 1\_ 진짜최종 \_2016\_02\_29.zip

과제 1\_ 진짜진짜최종 \_2016\_03\_01.zip

< History 관리가능 >  
**차이 (Diff)** 가 무엇이고  
**수정 이유**를 Log 를 남길수있다.



# Git 의 기능

Ctrl+c, v 나 Alzip 압축파일 관리법

Source code management **tool**

그 사이에 뭐가 바뀌었는지  
차이 (Diff) 를 알 수 없다.

Ctrl + c, v 를 할수록  
차지하는 용량 X 2  
X 3 ... + diff



과제 1\_ 최종 \_2016\_02\_28.zip

과제 1\_ 진짜최종 \_2016\_02\_29.zip

과제 1\_ 진짜진짜최종 \_2016\_03\_01.zip

< History 관리가능 >  
**차이 (Diff)** 가 무엇이고  
**수정 이유**를 Log 를 남길수있다.

< 타임머신 가능 >  
**현재파일들은 안전한 상태로**  
**과거**상태 그대로 복원가능 (반대도 가능)  
( 각 버전별 **차이만 저장**해서 **size 감소** )

# Git 의 기능

Ctrl+c, v 나 Alzip 압축파일 관리법

Source code management **tool**

그 사이에 뭐가 바뀌었는지  
차이 (Diff) 를 알 수 없다.

Ctrl + c, v 를 할수록  
차지하는 용량 X 2  
X 3 ... + diff



과제 1\_ 최종 \_2016\_02\_28.zip

과제 1\_ 진짜최종 \_2016\_02\_29.zip

과제 1\_ 진짜진짜최종 \_2016\_03\_01.zip

< History 관리가능 >  
**차이 (Diff)** 가 무엇이고  
**수정 이유**를 Log 를 남길수있다.

< 특정 버전 관리 >  
tag 나 release 로  
관리가능

< 타임머신 가능 >  
**현재파일들은 안전한 상태로**  
**과거**상태 그대로 복원가능 (반대도 가능)  
( 각 버전별 **차이만 저장해서 size 감소** )

# Git 의 기능

Ctrl+c, v 나 Alzip 압축파일 관리법



Git 배우는데 시간소비하느니  
Code 한줄이라도 더 개발 ..)

Source code management **tool**



좋은건 알겠는데 ..  
Git 을 쓸 이유가 부족 ..



과제 1\_ 최종 \_2016\_02\_28.zip

과제 1\_ 진짜최종 \_2016\_02\_29.zip

과제 1\_ 진짜진짜최종 \_2016\_03\_01.zip



# Git 실습 각 단계별

---



## Git 실습 각 단계별

---

Don't think about git,  
just do git

똑같이 따라 해보자 Git (<https://github.com/Taeung/git-training>)

# Git 실습 방법 , 시나리오

---

Git 실습 방법은 ?

직접 C 프로그래밍을 짜면서 Git 을 사용한다는 시나리오

Knapsack problem ( 배낭 채우기문제 ) 알고리즘

- 1) 코딩은 Ctrl-c, v 로 하고
- 2) Git 은 직접 명령어 쳐서 (git-bash)

# Git 실습 준비하기

---

## 1. 예제소스 다운받기

[https://www.dropbox.com/sh/9q2emkxmyckoj6/AAA\\_H55BVhfRvGHOS9j7l9N2a?dl=1&pv=1](https://www.dropbox.com/sh/9q2emkxmyckoj6/AAA_H55BVhfRvGHOS9j7l9N2a?dl=1&pv=1)

## 2. Git 설치하기

▶ <https://git-scm.com/downloads>


## 3. Editor 다운받기

▶ <https://atom.io/>

## 4. Github 회원가입


▶ <https://github.com/join>


# Git 실습 Github 회원가입


 Personal Open source Business Explore Pricing Blog Support  [Sign in](#) [Sign up](#)

## Join GitHub

The best way to design, build, and ship software.

 **Step 1:**  
Set up a personal account

 **Step 2:**  
Choose your plan

 **Step 3:**  
Go to your dashboard

### Create your personal account

**Username**

This will be your username — you can enter your organization's username next.

**Email Address**

You will occasionally receive account related emails. We promise not to share your email with anyone.

**Password**

Use at least one lowercase letter, one numeral, and seven characters.

By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#).

[Create an account](#)

### You'll love GitHub

**Unlimited** collaborators  
**Unlimited** public repositories

- ✓ Great communication
- ✓ Friction-less development
- ✓ Open source community

가입정보 기입



<https://github.com/join>



# Git 실습

## Github 메일인증하기

본인 메일확인해서 버튼 누르기



Hi @yukult400!

Help us secure your GitHub account by verifying your email address (yukult400@naver.com). This lets you access all of GitHub's features.

Verify email address

Button not working? Paste the following link into your browser:

[https://github.com/users/yukult400/emails/21042026/confirm\\_verification/388b1d3024fc012d64848df72d2d8fc4d7dc3b5a](https://github.com/users/yukult400/emails/21042026/confirm_verification/388b1d3024fc012d64848df72d2d8fc4d7dc3b5a)

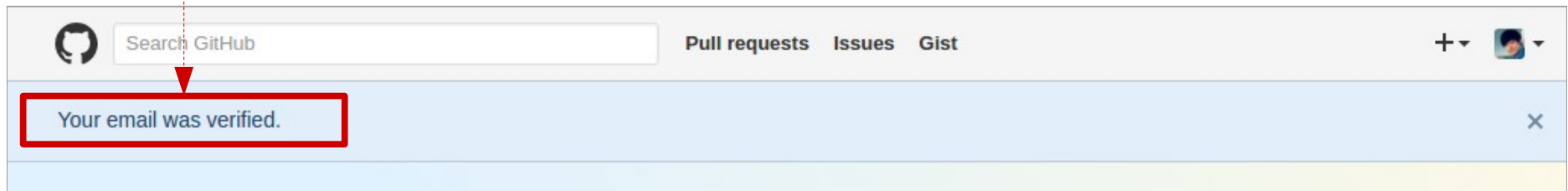
You're receiving this email because you recently created a new GitHub account or added a new email address. If this wasn't you, please ignore this email.

GitHub

# Git 실습 Github 메일인증하기

---

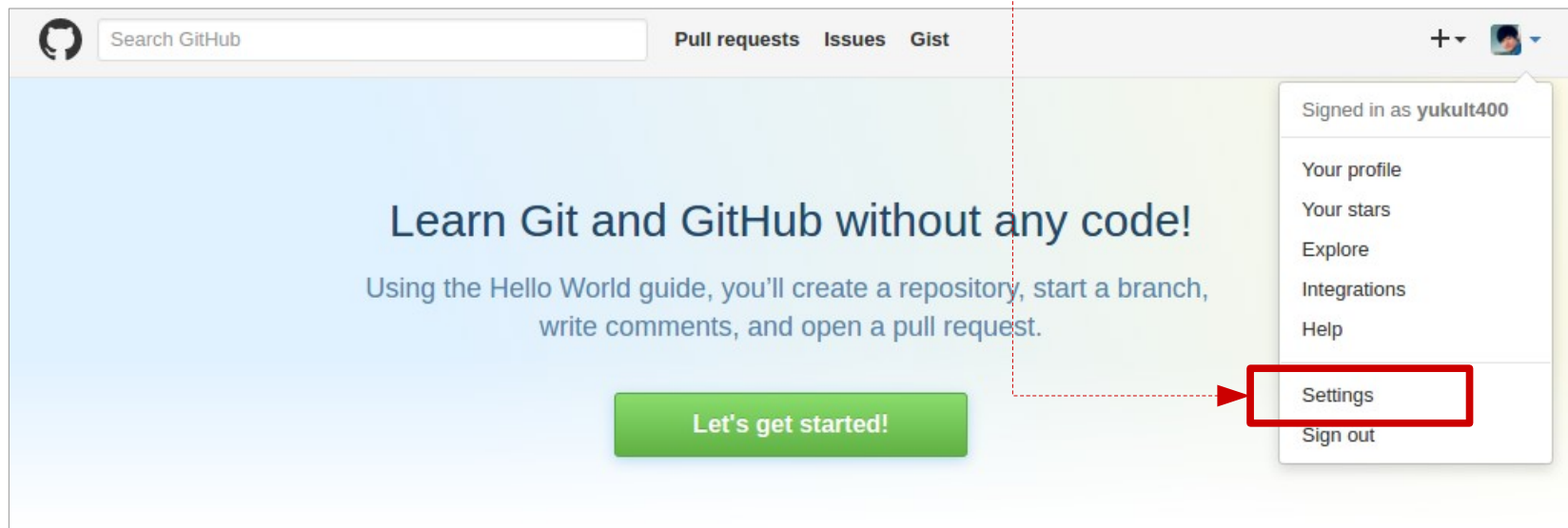
다시 Github 접속해서 메일인증 확인하기



<http://github.com>

# Git 실습 Github 메일인증하기

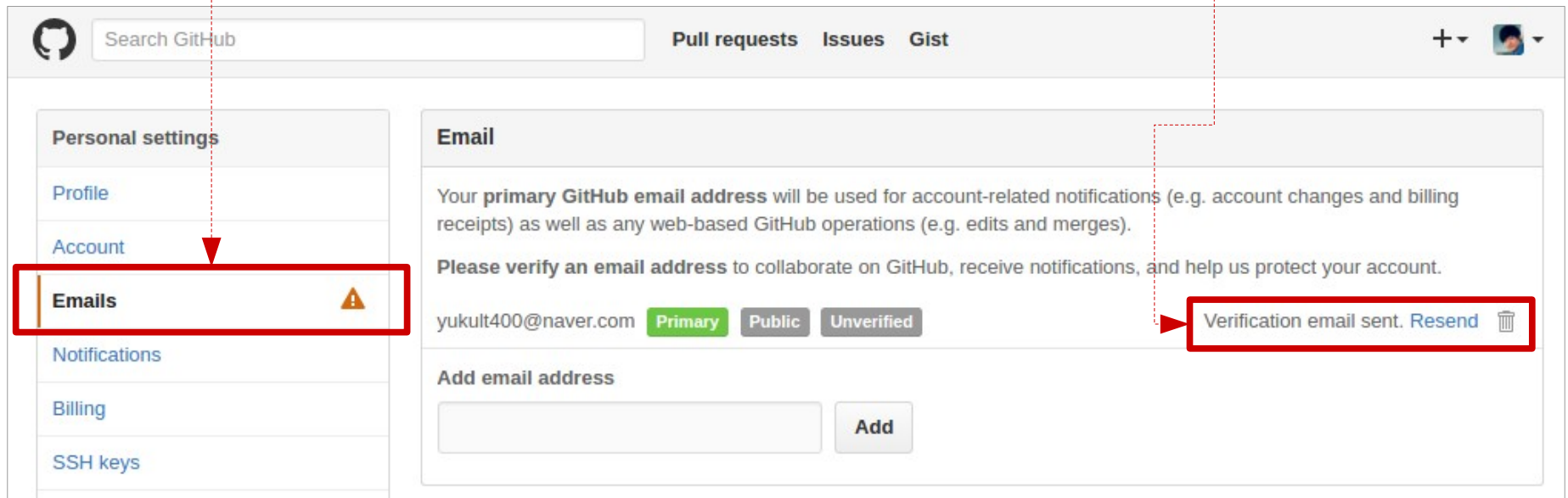
기다려도 메일이 혹시 안왔으면 설정버튼



<http://github.com>

# Git 실습 Github 메일인증하기

메일이 혹시 안왔으면 인증메일 다시 보내기



<https://github.com/settings/emails>

# Git 실습 준비하기

---

5. git\_training\_scenario.zip 압축풀고 폴더 열어두기

→ Ctrl-c, v 복사, 붙여넣기 할 소스들

6. 편집기 (atom, sublime text) 열어두기

→ Ctrl-c, v 복사, 붙여넣기 용도 편집기

# Git 실습 준비하기

7. Stage 단계단계 해결할때마다 “본인이름” 그래프에 색을 채우자

[https://docs.google.com/spreadsheets/d/1uPMCOKISMgj\\_svsoxG1LF1RbozA9RMKfx7h9vT80Atc/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1uPMCOKISMgj_svsoxG1LF1RbozA9RMKfx7h9vT80Atc/edit?usp=sharing)

[illegible]

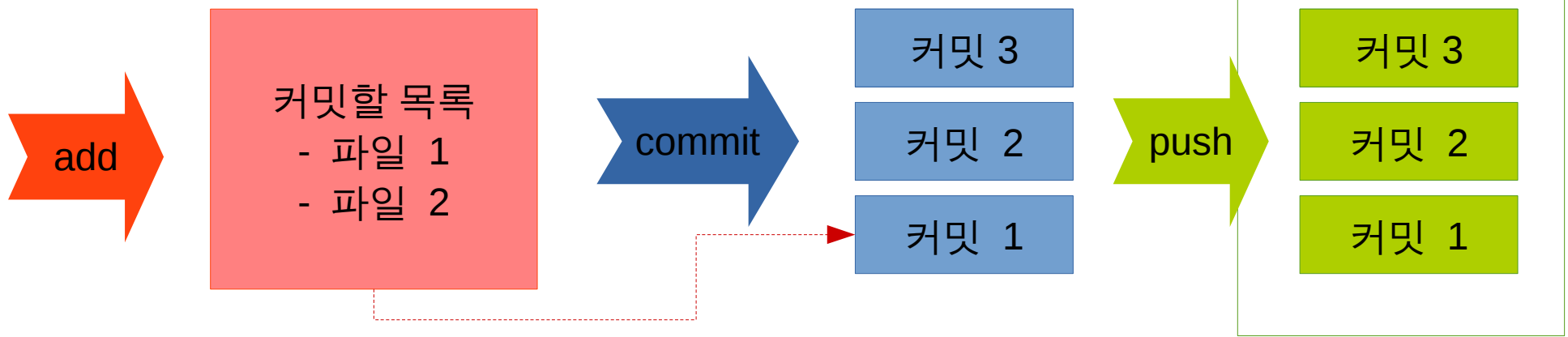
# Git 실습 준비하기

## < Git 필수 명령 >

**add** : 커밋할 목록에 추가

**commit** : 커밋 ( 히스토리의 한단위 ) 만들기

**push** : 현재까지 역사 (commits) Github 에 밀어넣기



# Git 실습

## Stage 1 초기화 및 첫 commit 하기 (Basic)

---

**몰라도 좋으니, 일단 진행을 해보자.  
( 막히면 조교를 부르세요.)**

1) Git bash 를 실행 ( 명령어칠 준비 ), 폴더생성하기

```
# mkdir git-training
```

2) 경로 이동

```
# cd git-training
```

3) 해당 폴더를 git 초기화

```
# git init
```

4) 편집기로 README.md 파일 (**commit\_1**) 만들고 본인 이름적고 나서 add 명령 입력

```
# git add README.md
```

5) 첫 commit 하기

```
# git commit -m "Add README file"
```



# Git 실습 Stage 2 knapsack 문제파일 추가하기 (Basic)

---

몰라도 좋으니, 일단 진행을 해보자.

1) 상태를 확인한다

```
# git status
```

2) knapsack 알고리즘 문제 파일 (commit\_2) 을 commit 할 준비시킨다.

```
# git add knapsack_problem.pdf
```

3) 다시한번 상태를 확인해서 1) 과 달라졌음을 느끼자

```
# git status
```

4) 준비된 파일을 커밋한다.

```
# git commit -m "Add knapsack problem PDF"
```

5) commit 들을 확인하자

```
# git log
```

# Git 실습 Stage 3 commit 에 본인서명 포함하기 (Basic)

**몰라도 좋으니, 일단 진행을 해보자.**

1) 상태를 확인한다

```
# git status
```

2) 소스파일 (**commit\_3**) 을 commit 할목록 에 준비시킨다 .

```
# git add packing_knapsack.c
```

3) 다시한번 상태를 확인해서 1) 과 달라졌음을 느끼자

```
# git status
```

4) 준비된 파일을 커밋한다 .

```
# git commit -sm "packing knapsack: Basic code sloving this question"
```

5) 오류가 난다면 대처하자 ( 설정하기 → 그리고 다시 커밋하자 )

```
# git config user.email " 본인메일적으세요 @gmail.com"
```

```
# git config user.name " 본인이름적으세요 Taeung Song"
```

# Git 실습 Stage 4 commit message 를 editor 로 적어보자 (Basic)

**몰라도 좋으니, 일단 진행을 해보자.**

1) 소스파일 (**commit\_4**) packing\_knapsack.c 으로 수정후에 diff 상태를 확인한다

```
# git diff
```

2) 소스파일 (**commit\_4**) 을 commit 할목록 에 준비시킨다 .

```
# git add packing_knapsack.c
```

3) 다시한번 상태를 확인해서 1) 과 달라졌음을 느끼자

```
# git status
```

4) 준비된 파일을 commit 하는데 이번엔 -m 쓰지말아보자 . (vi 와 같음 편집하고 :wq 로 나가자 )  
Commit message 는 “ packing knapsack: Change basic code”

```
# git commit -s
```

5) 방금한 commit 을 확인하자

```
# git show
```

# Git 실습 Stage 5 지금까지의 commit 을 push 하자 (Basic)

---

**몰라도 좋으니, 일단 진행을 해보자.**

1) 상태를 확인하고 현재 브랜치명 master 를 확인하자

```
# git status
```

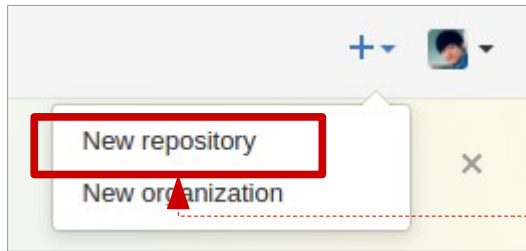
2) 지금까지한 commit 들을 확인하자

```
# git log
```

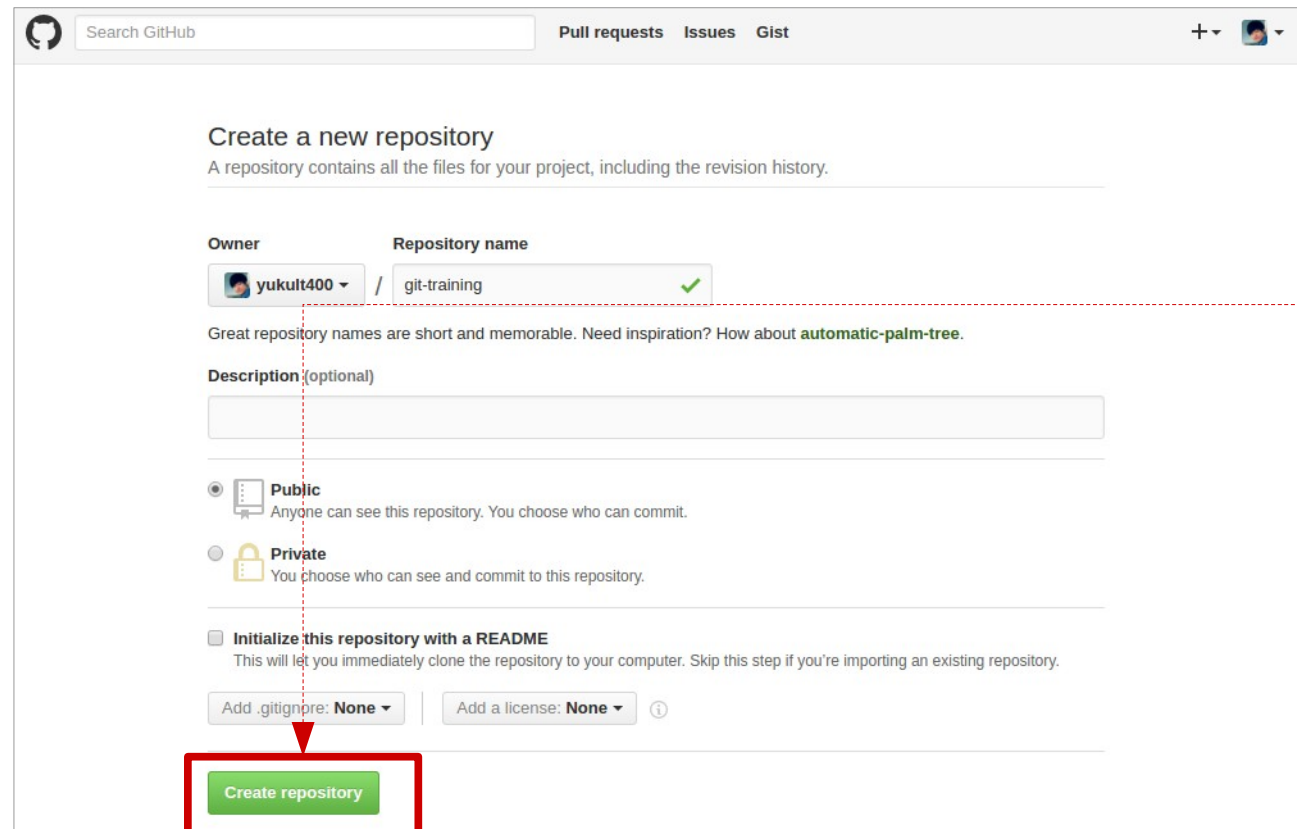
3) Github 원격저장소 URL 를 등록하자

**( 잠깐 멈추고 <http://github.com> 를 켜고 repository 새로 생성하자 )**

# Git 실습 Github 에서 원격저장소 만들기



새로운 원격 저장소를 생성하자 (클릭)



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: yukult400 / Repository name: git-training ✓

Great repository names are short and memorable. Need inspiration? How about [automatic-palm-tree](#).

Description (optional)

☒ Public  
Anyone can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

☐ Initialize this repository with a README  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None Add a license: None ⓘ

Create repository

# Git 실습 Github 에서 원격저장소 만들기

해당 URL 복사해서 Stage5 의 4) 으로 이어서 진행

The screenshot shows the GitHub interface for the repository 'yukult400 / git-training'. At the top, there are tabs for 'Pull requests', 'Issues', and 'Gist'. Below the repository name, there are buttons for 'Unwatch', 'Star', and 'Fork'. The 'Code' tab is selected, showing options for 'Issues', 'Pull requests', 'Wiki', 'Pulse', 'Graphs', and 'Settings'. The 'Quick setup' section is highlighted, showing the SSH URL 'https://github.com/yukult400/git-training.git' in a red box. Below this, there is a section for creating a new repository on the command line with the following commands:

```
echo "# git-training" >> README.md
git init
git add README.md
```

# Git 실습 Stage 5 지금까지의 commit 을 push 하자 (Basic)

---

4) 아까복사한 URL 로 Github 원격저장소 등록하자

```
# git remote add origin < 아까복사한 URL>
```

5) Github 원격저장소 (origin) 에다가 밀어 넣자 .

```
# git push origin master
```

6) Github 웹페이지 열고 확인하자

# Git 실습 Stage 5 지금까지의 commit 을 push 하자 (Basic)

Github 들어가서 본인 프로젝트의 commit 기록 눌러보자

This repository Search Pull requests Issues Gist

Taeung / git-training Unwatch 1 Star 0 Fork 1

Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings

Don't think about git, just do git — Edit

10 commits 1 branch 0 releases 1 contributor

You force-pushed to **master** an hour ago. You can enable **branch protection** to prevent this. [Dismiss](#) [Protect branch](#)

Branch: master New pull request New file Upload files Find file HTTPS https://github.com/Taeu Download ZIP

Taeung test git-pull-request Latest commit 05297af an hour ago

packing\_knapsack packing knapsack: Implement code to solve this question an hour ago



# Git 실습 Stage 5 지금까지의 commit 을 push 하자 (Basic)

본인이 추가한 commit 들이 나오는걸 확인하자 ( 본인 Github)

The screenshot shows the GitHub interface for the repository 'Taeung / git-training'. The 'Code' tab is selected, and the 'master' branch is chosen. A list of commits is displayed, with a red box highlighting the commit history and a red arrow pointing to it from the text above.

Commit Message	Commit Hash
test git-pull-request	05297af
packing knapsack: Implement code to solve this question	bd5ac7e
packing knapsack: Add test script generator	096e810
packing knapsack: Add test script	c55860f
packing knapsack: Input & Output	36f19a9
packing knapsack: Rename packing_knapsack to pack_knapsack	02a0e75
packing knapsack: Change basic code	2706b76
packing knapsack: Basic code solving this question	450316f
Add knapsack problem PDF	4ddf1d5
Add README file	cb06f3f

# Git 실습 Stage 6 커밋 수정하기 (Basic)

---

1) 커밋 메시지 (**commit\_4**) 로 수정해보자 (vi 로 수정후 :wq 로 나온다)

```
# git commit -amend
```

2) 바로 push 해보자 ( 충돌 오류발생 )

```
# git push origin master
```

3) 강제로 push 해서 수정하자

```
# git push origin master -f
```

4) 다시 Github 가서 제대로 변경되었는지 확인해보자

# Git 실습 Stage 7 소스변경후 add→commit→push (Basic)

1) 소스파일 (commit\_5) packing\_knapsack.c 으로 수정후에 diff 상태를 확인한다

```
# git diff
```

2) 소스파일 (commit\_5) 을 commit 할목록 에 준비시킨다 .

```
# git add packing_knapsack.c
```

3) 준비된 파일을 commit 한다 .

```
# git commit -sm "packing knapsack: Rename packing_knapsack to pack_knapsack"
```

4) Github 원격저장소 (origin) 에다가 밀어 넣자 .

```
# git push origin master
```

**add** : 커밋할 목록에 추가

**commit** : 커밋 ( 히스토리의 한단위 ) 만들기

**push** : 현재까지 역사 (commits) Github 에 밀어넣기

# Git 실습 Stage 8 소스변경후 add→commit→push (Basic)

반복만이 살길 ..

1) 소스파일 (commit\_6) packing\_knapsack.c 으로 수정후에 diff 상태를 확인한다

```
# git diff
```

2) 소스파일 (commit\_6) 을 commit 할목록 에 준비시킨다 .

```
# git add packing_knapsack.c
```

3) 준비된 파일을 commit 한다 .

```
# git commit -sm "packing knapsack: Input & Output"
```

4) Github 원격저장소 (origin) 에다가 밀어 넣자 .

```
# git push origin master
```

**add** : 커밋할 목록에 추가

**commit** : 커밋 ( 히스토리의 한단위 ) 만들기

**push** : 현재까지 역사 (commits) Github 에 밀어넣기

# Git 실습 Stage 9 파일추가후 add→commit→push (Basic)

1) test 파일 (commit\_7) test.sh 을 추가후에상태를 확인한다

```
# git status
```

2) test.sh (commit\_7) 을 commit 할목록 에 준비시킨다 .

```
# git add test.sh
```

3) 준비된 파일을 commit 한다 .

```
# git commit -sm "packing knapsack: Add test script"
```

4) Github 원격저장소 (origin) 에다가 밀어 넣자 .

```
# git push origin master
```

**add** : 커밋할 목록에 추가

**commit** : 커밋 ( 히스토리의 한단위 ) 만들기

**push** : 현재까지 역사 (commits) Github 에 밀어넣기

# Git 실습 Stage 9 파일추가후 add→commit→push (Basic)

5) test 파일 (commit\_8) test.sh 을 추가후에상태를 확인한다

```
# git status
```

6) test.sh (commit\_8) 을 commit 할목록 에 준비시킨다 .

```
# git add generator_test_script.py
```

7) 준비된 파일을 commit 한다 .

```
# git commit -sm "packing knapsack: Add test script generator"
```

8) Github 원격저장소 (origin) 에다가 밀어 넣자 .

```
# git push origin master
```

**add** : 커밋할 목록에 추가

**commit** : 커밋 ( 히스토리의 한단위 ) 만들기

**push** : 현재까지 역사 (commits) Github 에 밀어넣기

# Git 실습 Stage 10 마지막 수정후 add→commit→push (Basic)

---

1) 소스파일 (commit\_9) packing\_knapsack.c 으로 수정후에 diff 상태를 확인한다

```
# git diff
```

2) 소스파일 (commit\_9) 을 commit 할목록 에 준비시킨다 .

```
# git add packing_knapsack.c
```

3) 준비된 파일을 commit 한다 .

```
# git commit -sm "packing knapsack: Implement code to solve this question"
```

4) Github 원격저장소 (origin) 에다가 밀어 넣자 .

```
# git push origin master
```

프로그램 실행은 Linux 환경에서 가능

# Git 자유실습 반복연습 or Keep going

---





# Git 자유실습 반복연습 or Keep going

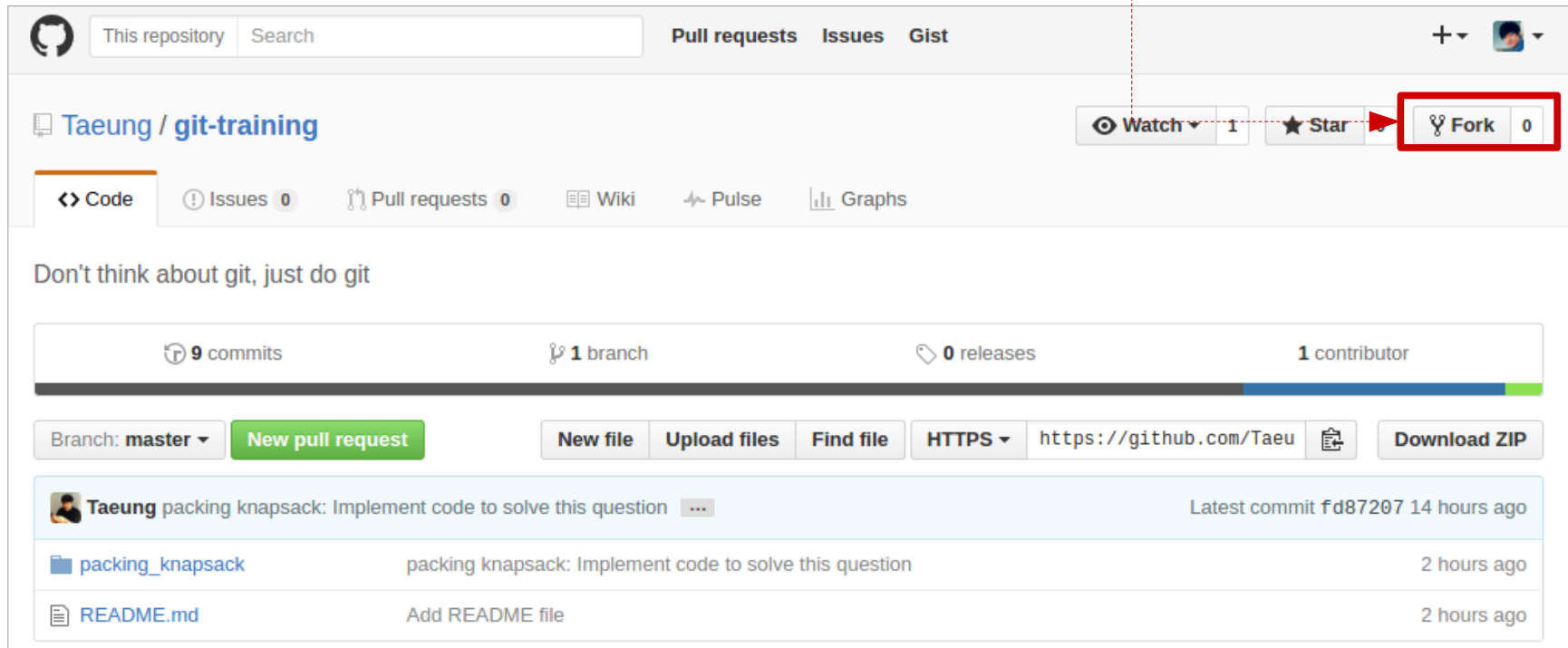
---

다시 Stage 1 부터 새롭게 반복연습 하거나

조금은 어려울 수 있으나 고급과정을 계속해서 진행 하거나

# Git 실습 Github 에서 fork 하기

<http://github.com/taeung/git-training> 가서 fork 버튼 누르자



# Git 실습 Github 에서 fork 하기

fork 가 되면 내 원격저장소가 추가 된다

The screenshot shows the GitHub interface for a repository named 'yukult400 / git-training-1', which is a fork of 'Taeung/git-training'. The repository page includes a header with the repository name, a search bar, and navigation tabs for 'Pull requests', 'Issues', and 'Gist'. Below the header, there are buttons for 'Unwatch', 'Star' (0), and 'Fork' (1). The main content area displays the repository's metadata: '9 commits', '1 branch', '0 releases', and '1 contributor'. A 'New pull request' button is visible. The repository's URL is shown as 'https://github.com/yuku'. The commit history is visible, showing a commit by 'Taeung' with the message 'packing knapsack: Implement code to solve this question'.

fork 해서 만들어진 repo 의 URL 복사

나의 profile 페이지에서도 확인가능 (Github)

The screenshot shows the 'Popular repositories' section of a GitHub profile. It lists two repositories: 'git-training' and 'git-training-1'. The 'git-training-1' repository is highlighted with a red box, showing its name and the description 'Don't think about git, just do git'.

# Git 자유실습 Stage 11 pull-request 하기 (Advanced)

---

1) clone 으로 fork 한 repo 받아오기

```
# git clone < 아까 fork 한 repo 에서 복사한 URL>
```

2) pull-request 작업할 브랜치 따로 만들기

```
# git checkout -b knapsack
```

3) pull\_request\_test 폴더로 이동해서

```
# cd pull_request_test
```

4) 내 이름으로된 폴더 만들고

```
# mkdir taeung
```

5) 내이름으로된 폴더에 소스 넣고나서

# Git 자유실습 Stage 11 pull-request 하기 (Advanced)

---

6) 추가한 폴더 통째로 add

```
# git add packing_knapsack
```

7) 준비된 파일들 commit

```
# git commit -sm "test pull request"
```

8) 내가 fork 한 repo 의 knapsack 브랜치로 push (주의 : master 아님)

```
# git push origin knapsack
```

# Git 자유실습 Stage 11 pull-request 하기 (Advanced)

The screenshot shows the GitHub interface for a repository named 'yukult400 / git-training-1', which is a fork of 'Taeung/git-training'. The repository has 9 commits, 2 branches, 0 releases, and 1 contributor. The 'Compare & pull request' button is highlighted with a red box. A red arrow points from this button to a text box at the bottom of the image.

Don't think about git, just do git — Edit

9 commits 2 branches 0 releases 1 contributor

Your recently pushed branches:

- knapsack-v2 (less than a minute ago)

Branch: master New pull request New file Upload files Find file HTTPS https://github.com/yuku Download ZIP

This branch is 9 commits ahead, 10 commits behind Taeung:master.

Commit	Author	Message	Time
Latest commit fd87207	Taeung	packing knapsack: Implement code to solve this question	14 hours ago
			3 hours ago
			3 hours ago

Github 들어가서 fork 한 저장소에서 pull-request 버튼 누르기

# Git 자유실습 Stage 11 pull-request 하기 (Advanced)

This repository Search Pull requests Issues Gist

Taeung / git-training Watch 1 Star 0 Fork 1

Code Issues 0 Pull requests 0 Wiki Pulse Graphs

## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

base fork: Taeung/git-training base: master ... head fork: yukult400/git-training-1 compare: knapsack

✓ Able to merge. These branches can be automatically merged.

test git-pull-request

Write Preview

Signed-off-by: Taeung Song <treeze.taeung@gmail.com>

Attach files by dragging & dropping, [selecting them](#), or pasting from the clipboard.

Styling with Markdown is supported

Create pull request

Pull-request 보내기

# Git 자유실습 Stage 11 pull-request 하기 (Advanced)

The screenshot shows a GitHub pull request interface. At the top, the repository name 'Taeung / git-training' is highlighted with a red box. Below it, the 'Pull requests' tab is also highlighted with a red box. The pull request title is 'test git-pull-request #1'. The description indicates that 'yukult400' wants to merge 1 commit into 'Taeung:master' from 'yukult400:knapsack'. The interface shows 0 conversations, 1 commit, and 4 files changed. A comment from 'yukult400' is visible, stating 'Signed-off-by: Taeung Song treeze.taeung@gmail.com'. A green box with a checkmark indicates 'This branch has no conflicts with the base branch'. The right sidebar shows labels, milestones, assignees, and notifications.

만들어진 pull-request 확인하기



# Git 자유실습

Stage 12 blame 으로 추적하기 (Advanced)

---

1) 어떤 파일이든 누가 어느라인을 수정했는지 파악해보자

```
# git blame packing_knapsack.c
```

2) 해당 commit ID 로 그 당시 commit 정보확인

```
# git show <commit ID>
```

# Git 자유실습 Stage 13 브랜치 병합하기 (Advanced)

---

1) fork 한 repo 기준 일단 현 브랜치가 master 인지 확인

```
# git status
```

2) master 아니면 브랜치 변경

```
# git checkout master
```

3) 현 브랜치 master 에서 knapsack 브랜치 합치기

```
# git merge knapsack
```

# Git 자유실습 Stage 14 중간에 낀 commit 수정하기 (Advanced)

고난이도 기능 중 하나 **rebase -interactive**

1) commit 기록 맨위부터 8 번째 commit 을 수정해보자

```
# git rebase -i HEAD~8
```

2) vi 에디터가 열리면 수정하려는 commit 앞에 edit 이라 적자

```
# git rebase -i HEAD~8
```

3) 상태확인해서 rebase 진행 정상적인지 보고

```
# git rebase -i HEAD~8
```

4) commit 정보 수정 하고 --continue 로 마무리

```
# git commit --amend -sm "packing knapsack: Add knapsack problem PDF"
```

```
# git rebase --continue
```

# Git 이해하기 (Why, What)

---

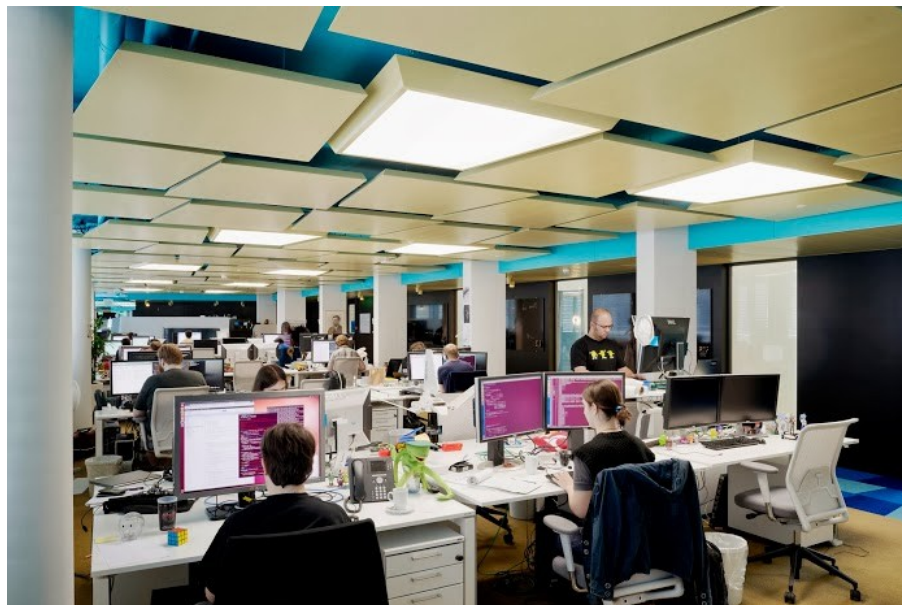


# Git 을 쓰는 이유

---

협업 때문에 Git 을 쓴다 . ( 집단지성의 극대화 )

현대적인 , 교양있는 , 문명화된 소스코드 개발과정을 위해 Git 을 쓴다 .



# Git 을 쓰는 이유

---

협업 때문에 Git 을 쓴다 . ( 집단지성의 극대화 )

현대적인 , 교양있는 , 문명화된 소스코드 개발과정을 위해 Git 을 쓴다 .

Coding Style 또는 Coding Convention 은 기본

Commits( 변화분 ) 단위 관리를 통해 ( 코드를 수정한 이유가 명확 )

이 변화분 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)

소스코드 품질상승 ( 다수의 버그 / 오류 미리 차단 , 추적용이 등 )

# Git 을 쓰는이유 (with Github)

\* 참고 : Linux kernel 은 Github 에서 mirror 까지는 되지만 다음 Repository 가 공식  
<http://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/>

Commits( **변화분** ) 단위 관리를 통해 ( 코드를 수정한 **이유**가 **명확** )

torvalds / linux

Watch 4,576 Unstar 30,768 Fork 12,172

Code Pull requests 96 Pulse Graphs

Linux kernel source tree

588,144 commits 1 branch 457 releases ∞ contributors

Branch: master New pull request New file Upload files Find file HTTPS https://github.com/torvalds/linux Download ZIP

torvalds Merge tag 'asm-generic-4.6' of git://git.kernel.org/pub/scm/linux/kernel... Latest commit 11caf57 9 hours ago













Failed to load latest commit information.

Documentation	Merge tag 'pm+acpi-4.6-rc1-2' of git://git.kernel.org/pub/scm/linux/kernel...	9 hours ago
arch	Merge tag 'asm-generic-4.6' of git://git.kernel.org/pub/scm/linux/kernel...	9 hours ago
block	Merge branch 'for-linus' of git://git.kernel.dk/linux-block	12 hours ago
certs	certs: Fix misaligned data in extra certificate list	25 days ago
crypto	Merge branch 'akpm' (patches from Andrew)	7 days ago
drivers	Merge tag 'asm-generic-4.6' of git://git.kernel.org/pub/scm/linux/kernel...	9 hours ago
firmware	WHENCE: use https://linuxtv.org for LinuxTV URLs	4 months ago

<https://github.com/torvalds/linux>

# Git 을 쓰는 이유 (with Github)

Commits( **변화분** ) 단위 관리를 통해 ( 코드를 수정한 **이유**가 **명확** )

	<b>Merge tag 'for-v4.5-rc' of git://git.kernel.org/pub/scm/linux/kernel/...</b> ... torvalds committed 29 days ago		1ebe383	
	<b>perf script: Remove duplicated code and needless script_spec__findnew()</b> ... Taeung committed with Arnaldo Carvalho de Melo 29 days ago		8560bae	
	<b>x86/mm: Fix slow_virt_to_phys() for X86_PAE again</b> ... dcui committed with Thomas Gleixner 29 days ago		bf70e55	
	<b>perf script: Exception handling when the print fmt is empty</b> ... Taeung committed with Arnaldo Carvalho de Melo 29 days ago		8579aca	



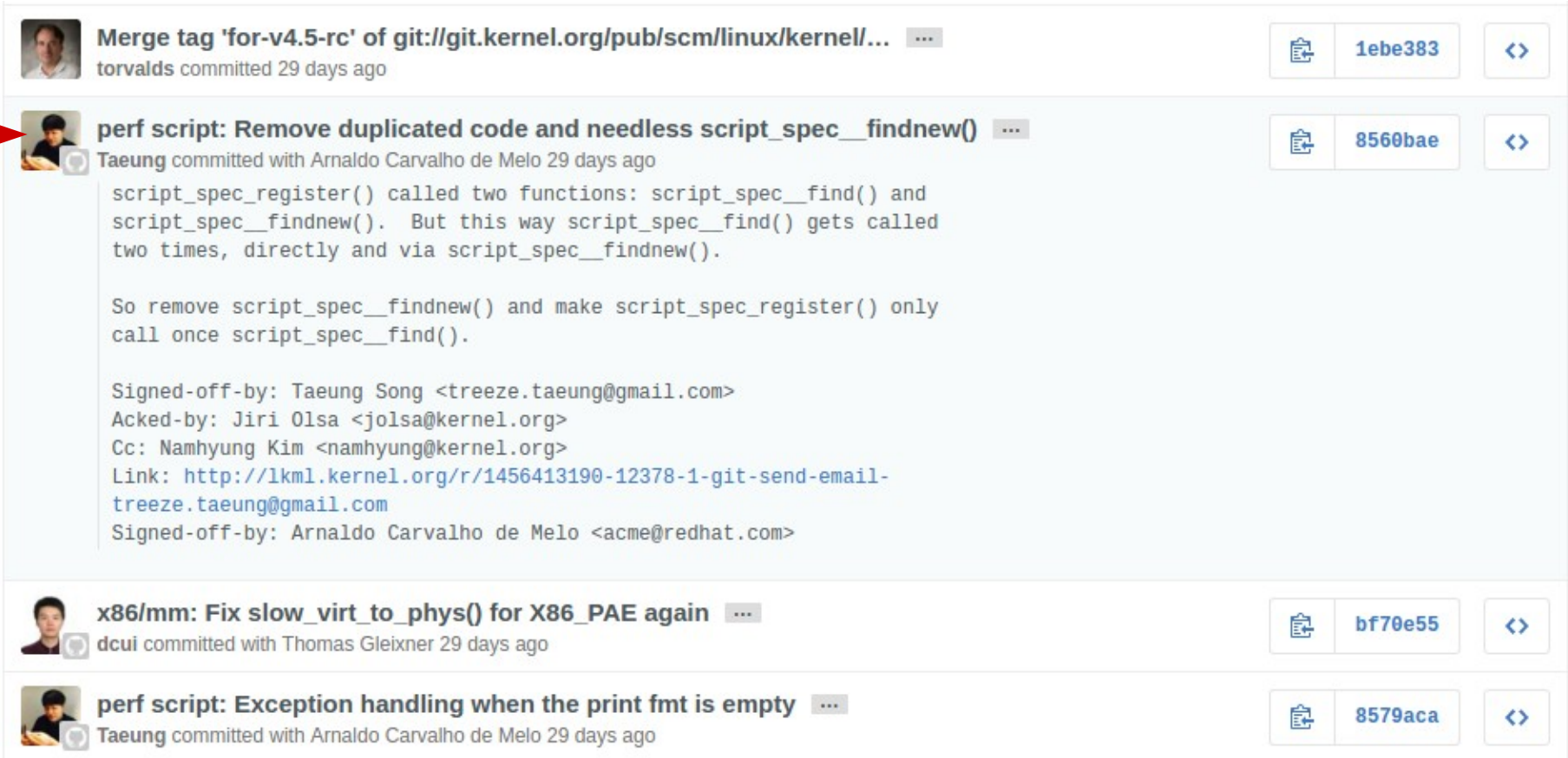
Commits on Feb 25, 2016

<https://github.com/torvalds/linux>



# Git 을 쓰는 이유 (with Github)

Commits( **변화분** ) 단위 관리를 통해 ( 코드를 수정한 **이유**가 **명확** )



**Merge tag 'for-v4.5-rc' of git://git.kernel.org/pub/scm/linux/kernel/...** **1e3e383** **<>**  
torvalds committed 29 days ago

**perf script: Remove duplicated code and needless script\_spec\_\_findnew()** **8560bae** **<>**  
Taeung committed with Arnaldo Carvalho de Melo 29 days ago

script\_spec\_register() called two functions: script\_spec\_find() and script\_spec\_findnew(). But this way script\_spec\_find() gets called two times, directly and via script\_spec\_findnew().

So remove script\_spec\_findnew() and make script\_spec\_register() only call once script\_spec\_find().

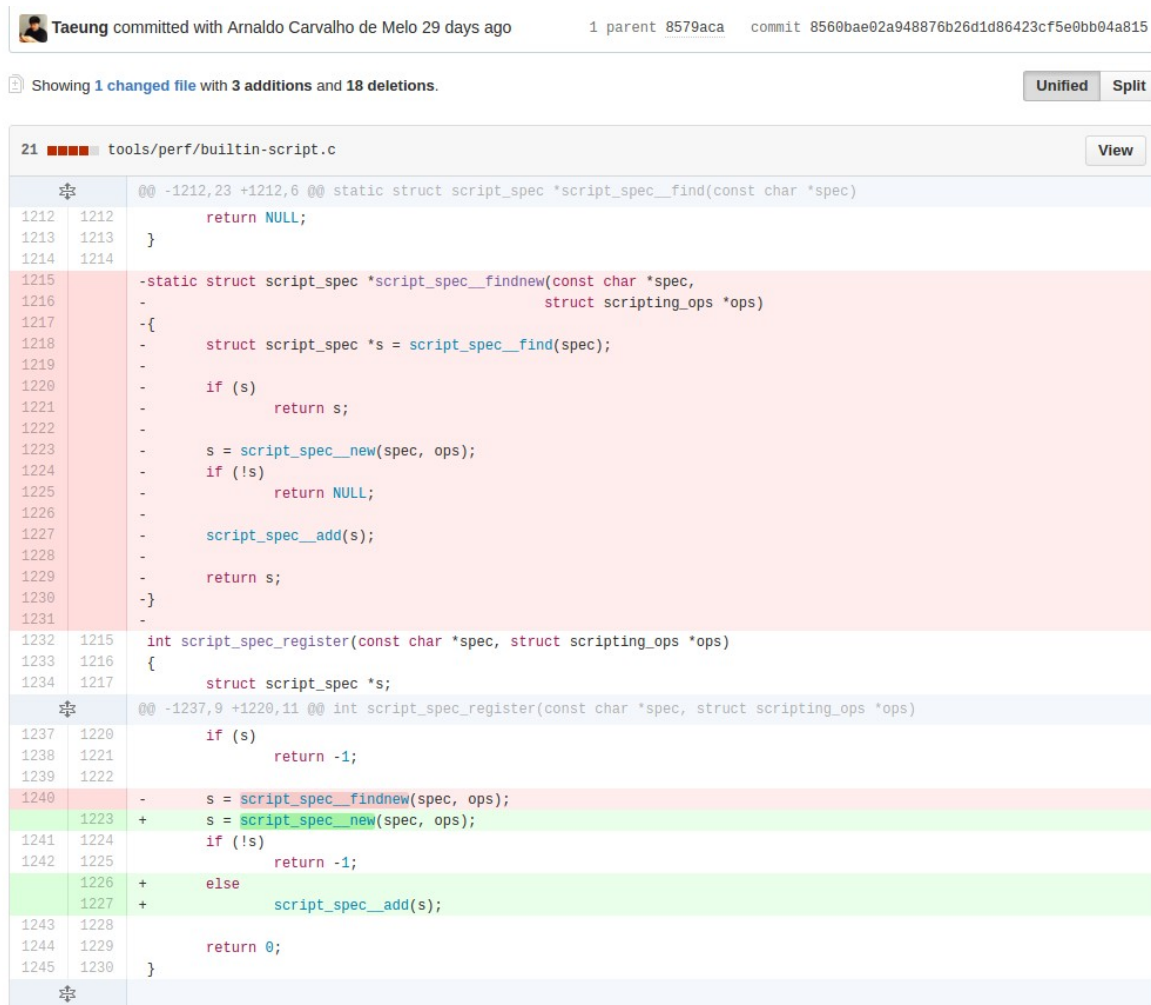
Signed-off-by: Taeung Song <treeze.taeung@gmail.com>  
Acked-by: Jiri Olsa <jolsa@kernel.org>  
Cc: Namhyung Kim <namhyung@kernel.org>  
Link: <http://lkml.kernel.org/r/1456413190-12378-1-git-send-email-treeze.taeung@gmail.com>  
Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>

**x86/mm: Fix slow\_virt\_to\_phys() for X86\_PAE again** **bf70e55** **<>**  
dcui committed with Thomas Gleixner 29 days ago

**perf script: Exception handling when the print fmt is empty** **8579aca** **<>**  
Taeung committed with Arnaldo Carvalho de Melo 29 days ago

# Git 을 쓰는 이유 (with Github)

Commits( **변화분** ) 단위 관리를 통해 ( 코드를 수정한 **이유**가 **명확** )



Taeung committed with Arnaldo Carvalho de Melo 29 days ago 1 parent 8579aca commit 8560bae02a948876b26d1d86423cf5e0bb04a815

Showing 1 changed file with 3 additions and 18 deletions. Unified Split

21 tools/perf/builtin-script.c View

```
@@ -1212,23 +1212,6 @@ static struct script_spec *script_spec__find(const char *spec)
1212 1212     return NULL;
1213 1213 }
1214 1214
1215 -static struct script_spec *script_spec__findnew(const char *spec,
1216 - struct scripting_ops *ops)
1217 -{
1218 -     struct script_spec *s = script_spec__find(spec);
1219 -     if (s)
1220 -         return s;
1221 -     s = script_spec__new(spec, ops);
1222 -     if (!s)
1223 -         return NULL;
1224 -     script_spec__add(s);
1225 -     return s;
1226 - }
1227 -
1228 -int script_spec_register(const char *spec, struct scripting_ops *ops)
1229 -{
1230 -     struct script_spec *s;
1231 -     if (s = script_spec__findnew(spec, ops);
1232 -         if (!s)
1233 -             return -1;
1234 -         else
1235 -             script_spec__add(s);
1236 -     return 0;
1237 - }
```

<https://github.com/torvalds/linux>

# Git 을 쓰는 이유 (with Github)

이 **변화분** (commits) 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)

atom / electron

Watch 1,261 Unstar 25,814 Fork 2,594

Code Issues 322 **Pull requests 13** Pulse Graphs

Build cross platform desktop apps with web technologies <http://electron.atom.io>

7,832 commits 4 branches 216 releases 331 contributors

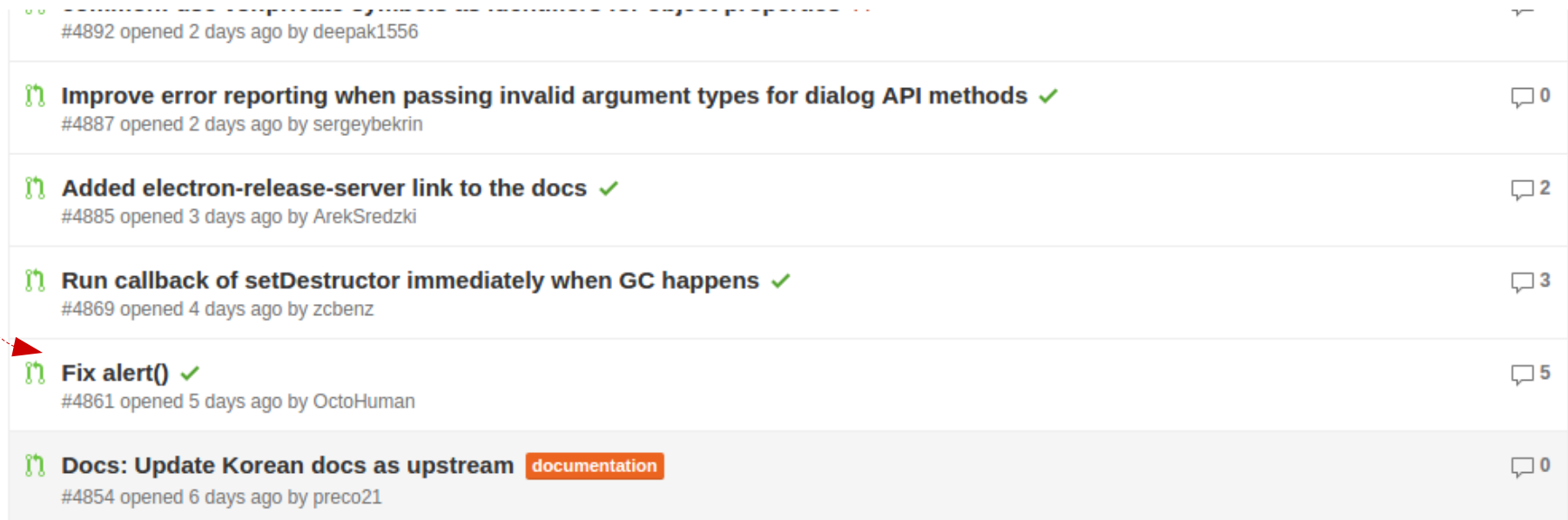
Branch: master New pull request New file Upload files Find file HTTPS <https://github.com/atom> Download ZIP






kevin sawicki Merge pull request #4883 from atom/repl Latest commit 1993139 12 hours ago

atom	Set _setDeprecatedOptionsCheck on exports	7 days ago
chromium_src	devtools: fix filesystem api usage and use prefs to track filesystem ...	11 days ago
default_app	Don't quit on window-all-closed when in repl mode	16 hours ago
docs-translations	fix link	6 days ago
...	...	...

# Git 을 쓰는 이유 (with Github)

이 **변화분** (commits) 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)



#4892 opened 2 days ago by deepak1556	
 <b>Improve error reporting when passing invalid argument types for dialog API methods</b> ✓ #4887 opened 2 days ago by sergeybekrin	0
 <b>Added electron-release-server link to the docs</b> ✓ #4885 opened 3 days ago by ArekSredzki	2
 <b>Run callback of setDestructor immediately when GC happens</b> ✓ #4869 opened 4 days ago by zcbenz	3
 <b>Fix alert()</b> ✓ #4861 opened 5 days ago by OctoHuman	5
 <b>Docs: Update Korean docs as upstream</b> <span>documentation</span> #4854 opened 6 days ago by preco21	0

# Git 을 쓰는 이유 (with Github)

이 **변화분** (commits) 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)

atom / electron

Watch 1,261 Unstar 25,814 Fork 2,594

Code Issues 322 Pull requests 13 Pulse Graphs

## Fix alert() #4861

Open OctoHuman wants to merge 2 commits into atom:master from OctoHuman:master

Conversation 5 Commits 2 Files changed 1 +4 -1

OctoHuman commented 5 days ago

Stop `alert()` from throwing an error when passing no parameters or when passing null or undefined as the message. For example if you run `alert()` in a browser it will display a blank dialog box, but in Electron it will throw a error because there is no `.toString()` method on `undefined`. Changing `alert()` like this will also make it more consistent with the major browsers implementations.

OctoHuman added some commits 5 days ago

- Fix alert() 9f65412
- Refix alert() 116d611

DerNivel commented on the diff 3 days ago

```
lib/renderer/override.js
...    ...    @@ -161,11 +161,14 @@ window.open = function(url, frameName, features) {
161    161    // Use the dialog API to implement alert().
162    162    window.alert = function(message, title) {
```

DerNivel added a note 3 days ago

Labels: None yet

Milestone: No milestone

Assignee: No one assigned

Notifications: Subscribe


3 participants

# Git 을 쓰는 이유 (with Github)

이 **변화분** (commits) 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)

Showing changes from 1 commit ▾ 1 changed file ▾ +1 -1 0 0 0 0 0 0 Diff options ▾

**Refix alert()** < Prev Next >


 OctoHuman committed 5 days ago commit 116d61185a887b50c2a214c0736e07dd6b806868

2 0 0 0 0 0 0 0 0 lib/renderer/override.js Show notes View

@@ -161,7 +161,7 @@ window.open = function(url, frameName, features) {


161 161 // Use the dialog API to implement alert().

162 162 window.alert = function(message, title) {


 DerNivel added a note 3 days ago +😊

Would it be possible to use default parameters. For example:


```
window.alert = function(message = '', title = '') {  
  // ...  
}
```

 OctoHuman added a note 3 days ago +😊


The problem with your code is if you passed `undefined` it still would be replaced by the default. So in your example passing `undefined` as the message would give me a blank `alert()` dialog. In a browser `alert()` will give me a blank dialog, but `alert(undefined)` will give me a dialog that says `undefined`

 DerNivel added a note 3 days ago +😊

I think no one would pass `undefined` to an `alert()` dialog.

 hermannplatz added a note 3 days ago +😊

you never know :)

 OctoHuman added a note 2 days ago +😊

# Git 을 쓰는 이유 (with Github)

이 **변화분** (commits) 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)

atom / electron

Watch 1,261 Unstar 25,814 Fork 2,594

Code Issues 322 Pull requests 13 Pulse Graphs

## Fix alert() #4861

Open OctoHuman wants to merge 2 commits into atom:master from OctoHuman:master

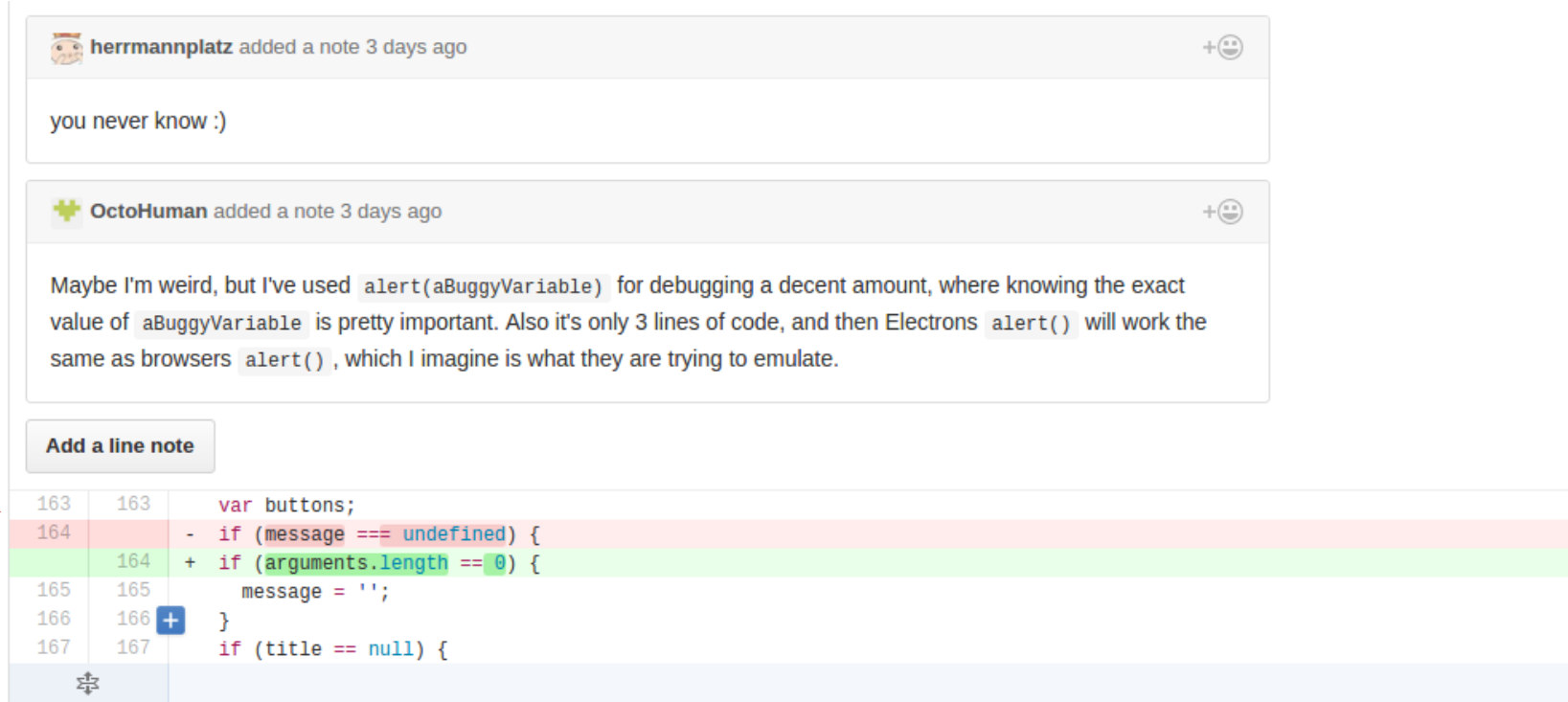
Conversation 5 Commits 2 Files changed 1 +4 -1

Commits on Mar 20, 2016

	Fix alert() OctoHuman committed 5 days ago ✖	9f65412
	Refix alert() OctoHuman committed 5 days ago ✔	116d611

# Git 을 쓰는 이유 (with Github)

이 **변화분** (commits) 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)



The screenshot displays a GitHub pull request interface. At the top, two users have added line notes to a code diff. The first note, from **herrmannplatz**, says "you never know :)". The second note, from **OctoHuman**, discusses the use of `alert(aBuggyVariable)` for debugging in Electron. Below the notes is a button labeled "Add a line note". The code diff shows a change on line 164, where a new condition `arguments.length == 0` is added to an `if` statement. A red dashed arrow points from the text "이 변화분 (commits)" to the code diff area.

herrmannplatz added a note 3 days ago

you never know :)

OctoHuman added a note 3 days ago

Maybe I'm weird, but I've used `alert(aBuggyVariable)` for debugging a decent amount, where knowing the exact value of `aBuggyVariable` is pretty important. Also it's only 3 lines of code, and then Electrons `alert()` will work the same as browsers `alert()`, which I imagine is what they are trying to emulate.

Add a line note

163	163	<code>var buttons;</code>
164	-	<code>if (message === undefined) {</code>
164	+	<code>if (arguments.length == 0) {</code>
165	165	<code>message = '';</code>
166	166	<code>}</code>
167	167	<code>if (title == null) {</code>



# Git 을 쓰는 이유

---

소스코드 **품질** 상승 ( 다수의 버그 / 오류 미리 차단 , 추적용이 등 )

→ Git 을 활용하는 **오픈소스**가 대표적인 **증명** (IT 기업들의 높은 의존 , 소프트웨어 **역사**를 이끄는 )

# Git 을 쓰는 이유

---

▶ Junior 개발자 교육 효과 ( 프로개발자의 코드를 볼수 있는 기회 )

소스코드 **품질**상승 ( 다수의 버그 / 오류 미리 차단 , 추적용이 등 )

▶ Git 을 활용하는 **오픈소스**가 대표적인 **증명** ( IT 기업들의 높은 의존 , 소프트웨어 **역사**를 이끄는 )

# Github 란

---



Git 이라는 도구를 응용한 사이트

각종 Remote repository ( 원격저장소 ) 들의 집합소

# Opensource Github / Not Github

---

## Not Github

- Apache (<http://git.apache.org>)
- Linux kernel (<http://git.kernel.org>)
- GNU (<http://git.savannah.gnu.org/cgit/>)
- Webkit (<git://git.webkit.org/WebKit.git>)

...


## Github

- Node.js (<https://github.com/nodejs/node>)
- Angular.js (<https://github.com/angular/angular.js>)
- Spring-boot (<https://github.com/spring-projects/spring-boot>)
- Rust (<https://github.com/rust-lang/rust>)
- Redis (<https://github.com/antirez/redis>)
- Flask (<https://github.com/mitsuhiko/flask>)

...

# Not Github

Gitub 가 아닌 remote repo 관리 사이트

 <b>Kernel.org git repositories</b> Git repositories hosted at kernel.org				
<div>index</div> <div></div> <div>search</div>				
Name	Description	Owner	Idle	Links
<i>bluetooth</i>				
bluez-hcidump.git	Bluetooth packet analyzer	Marcel Holtmann	3 years	summary log tree
bluez.git	Bluetooth protocol stack for Linux	Marcel Holtmann	26 hours	summary log tree
obexd.git	OBEX Server	Marcel Holtmann	3 years	summary log tree
sbc.git	SBC library	holtmann	17 months	summary log tree
<i>boot</i>				
dracut/dracut.git	dracut - Initramfs generator using udev	Harald Hoyer	8 days	summary log tree
efilinux/efilinux.git	The efilinux UEFI boot loader	Matt Fleming	21 months	summary log tree
syslinux/syslinux.git	The Syslinux boot loader suite	Syslinux workgroup	18 months	summary log tree
<i>devel</i>				
pahole/pahole.git	Pahole and other DWARF utils	Arnaldo Carvalho de Melo	10 days	summary log tree
sparse/chrisl/sparse.git	Chris Li's sparse repository.	Christopher Li	5 weeks	summary log tree
sparse/sparse.git	C semantic parser	Christopher Li	13 months	summary log tree
<i>docs</i>				
kernel/kernel-docs.git	Kernel Documentation tree	Doc Group	2 years	summary log tree
kernel/ksmap.git	Kernel.org keysign map source	Kernel.org users	4 days	summary log tree
kernel/website.git	Kernel.org website source	Doc Group	13 days	summary log tree
man-pages/man-pages.git	Linux man pages Sections 2, 3, 4, 5, and 7	Michael Timothy Kerrisk	9 days	summary log tree
man-pages/website.git	Website files for /doc/man-pages	Michael Timothy Kerrisk	2 weeks	summary log tree
<i>editors</i>				
uemacs/uemacs.git	Micro-emacs	Linus Torvalds	16 months	summary log tree
<i>fs</i>				
ext2/e2fsprogs.git	Ext2/3/4 filesystem userspace utilities	Theodore T'so	3 days	summary log tree
ext2/xfstests-bld.git	Build framework and autorun scripts for xfstests	Theodore T'so	4 weeks	summary log tree
fat/fatattr/fatattr.git	FAT attribute set utility	H. Peter Anvin	7 years	summary log tree
fuse/dbfs.git	FUSE fs w/ Berkeley DB backend.	Jeff Garzik	7 years	summary log tree
fuse/fuse-ext2.git	FUSE ext2 filesystem driver.	Jeff Garzik	10 years	summary log tree
squashfs/squashfs-tools.git	squashfs tools development	Phillip Lougher	4 months	summary log tree
xfs/dmapi-dev.git	Data Management API runtime environment	Christoph Hellwig	5 years	summary log tree
xfs/xfs-documentation.git	XFS AsciiDoc Documentation tree	XFS FS Group	3 months	summary log tree

<https://git.kernel.org>

# Not Github PATCH mail 방식

이 **변화분** (commits) 을 통한 Review 와 Discussion 가능 (pull-request, PATCH mail)

**Araldo Carvalho de Melo** November 23, 2015 at 11:17 PM AC

To: Taeung Song Cc: Namhyung Kim, linux-kernel@vger.kernel.org, Ingo Molnar, Jiri Olsa  
Re: [PATCH v11 RESEND 02/24] perf tools: Add perf-config document

Em Sun, Nov 22, 2015 at 07:11:56PM +0900, Taeung Song escreveu:  
Add perf-config document to describe the perf configuration and a subcommand 'list'.

Worked this time, applied, will continue processing the other patches soon.

- Araldo

[See More](#)

☆ **tip-bot for Taeung Song** Today at 5:20 PM TF

To: linux-tip-commits@vger.kernel.org Cc: Ingo Molnar, and 7 more...

[tip:perf/core] perf config: Add initial man page

Commit-ID: 7d6852432acb3b09fc3ec45dd65421d34eebe3b5  
Gitweb: <http://git.kernel.org/tip/7d6852432acb3b09fc3ec45dd65421d34eebe3b5>  
Author: Taeung Song <[treeze.taeung@gmail.com](mailto:treeze.taeung@gmail.com)>  
AuthorDate: Sun, 22 Nov 2015 19:11:56 +0900  
Committer: Araldo Carvalho de Melo <[acme@redhat.com](mailto:acme@redhat.com)>  
CommitDate: Mon, 23 Nov 2015 18:31:25 -0300

perf config: Add initial man page

Add perf-config document to describe the perf configuration and a 'list' subcommand.

Signed-off-by: Taeung Song <[treeze.taeung@gmail.com](mailto:treeze.taeung@gmail.com)>  
Cc: Jiri Olsa <[jolsa@redhat.com](mailto:jolsa@redhat.com)>  
Cc: Namhyung Kim <[namhyung@kernel.org](mailto:namhyung@kernel.org)>  
Link: <http://lkml.kernel.org/r/63AD9B57-7B8C-46F8-8F18-0FFEB9A6A1BC@gmail.com>  
Signed-off-by: Araldo Carvalho de Melo <[acme@redhat.com](mailto:acme@redhat.com)>  
---

tools/perf/Documentation/perf-config.txt | 103 +++++  
1 file changed, 103 insertions(+)

**tip-bot for Taeung Song** Today at 5:20 PM TF

To: linux-tip-commits@vger.kernel.org  
Cc: tglx@linutronix.de, Jiri Olsa, linux-kernel@vger.kernel.org, Ingo Molnar, hpa@zytor.com, Taeung Song, Namhyung Kim, [acme@redhat.com](mailto:acme@redhat.com)

[tip:perf/core] perf tools: Add 'perf config' command

Commit-ID: 30862f2c5725c46afcfab5af710fdf5163bf0f81  
Gitweb: <http://git.kernel.org/tip/30862f2c5725c46afcfab5af710fdf5163bf0f81>  
Author: Taeung Song <[treeze.taeung@gmail.com](mailto:treeze.taeung@gmail.com)>  
AuthorDate: Tue, 17 Nov 2015 22:53:21 +0900  
Committer: Araldo Carvalho de Melo <[acme@redhat.com](mailto:acme@redhat.com)>  
CommitDate: Mon, 23 Nov 2015 18:31:24 -0300

perf tools: Add 'perf config' command

The perf configuration file contains many variables to change various aspects of each of its tools, including output, disk usage, etc.

But looking at the state of configuration is difficult and there's no documentation about config variables except for the variables in perftool.example exist.

So this patch adds a 'perf-config' command with a '--list' option.

perf config [options]

display current perf config variables.  
# perf config -l | --list

Signed-off-by: Taeung Song <[treeze.taeung@gmail.com](mailto:treeze.taeung@gmail.com)>  
Acked-by: Namhyung Kim <[namhyung@kernel.org](mailto:namhyung@kernel.org)>  
Cc: Jiri Olsa <[jolsa@redhat.com](mailto:jolsa@redhat.com)>  
Link: <http://lkml.kernel.org/r/1447768424-17327-1-git-send-email-treeze.taeung@gmail.com>  
Signed-off-by: Araldo Carvalho de Melo <[acme@redhat.com](mailto:acme@redhat.com)>  
---

tools/perf/Build | 1 +  
tools/perf/builtin-config.c | 66 +++++  
tools/perf/builtin.h | 1 +  
tools/perf/command-list.txt | 1 +  
tools/perf/perf.c | 1 +  
5 files changed, 70 insertions(+)

# Opensource 각종 오픈소스

The screenshot shows the Open Hub website interface. At the top, there's a navigation bar with links for PROJECTS, PEOPLE, ORGANIZATIONS, TOOLS, CODE, and BLOG. Below this is a large blue banner with the text "Discover, Track and Compare Open Source" and a search bar. The main content area is divided into several sections: "Join Now" with options to claim contributions, manage project data, and highlight FOSS use; "What's New" featuring a "FREE BLACK DUCK VULNERABILITY PLUGIN" advertisement; "Most Popular Projects" listing Mozilla Firefox, Apache HTTP Server, MySQL, Apache Subversion, PHP, and Linux Kernel; "Most Active Projects" listing Alfresco Content Management, Arch Linux Packages, Gentoo Linux, azure-content, and Mozilla Firefox; and "Most Active Contributors" listing Felix Yan, Translation updater bot, Luca suriano, Manfred Tournon, Mischa ter Smitten, and Ori Livneh.

**BLACKDUCK | Open HUB**

Follow @ OH SIGN IN JOIN NOW

PROJECTS PEOPLE ORGANIZATIONS TOOLS CODE BLOG

Discover, Track and Compare Open Source

Search Projects...

**Join Now**

- Claim your contributions
- Manage your project's data
- Highlight your use of FOSS

Join Now

**What's New**

**FREE BLACK DUCK VULNERABILITY PLUGIN**

Identify open source software in your projects and discover known vulnerabilities.

**Most Popular Projects**

Project	Users
Mozilla Firefox	13261 users
Apache HTTP Server	9466 users
MySQL	9180 users
Apache Subversion	8680 users
PHP	7789 users
Linux Kernel	7142 users

**Most Active Projects**

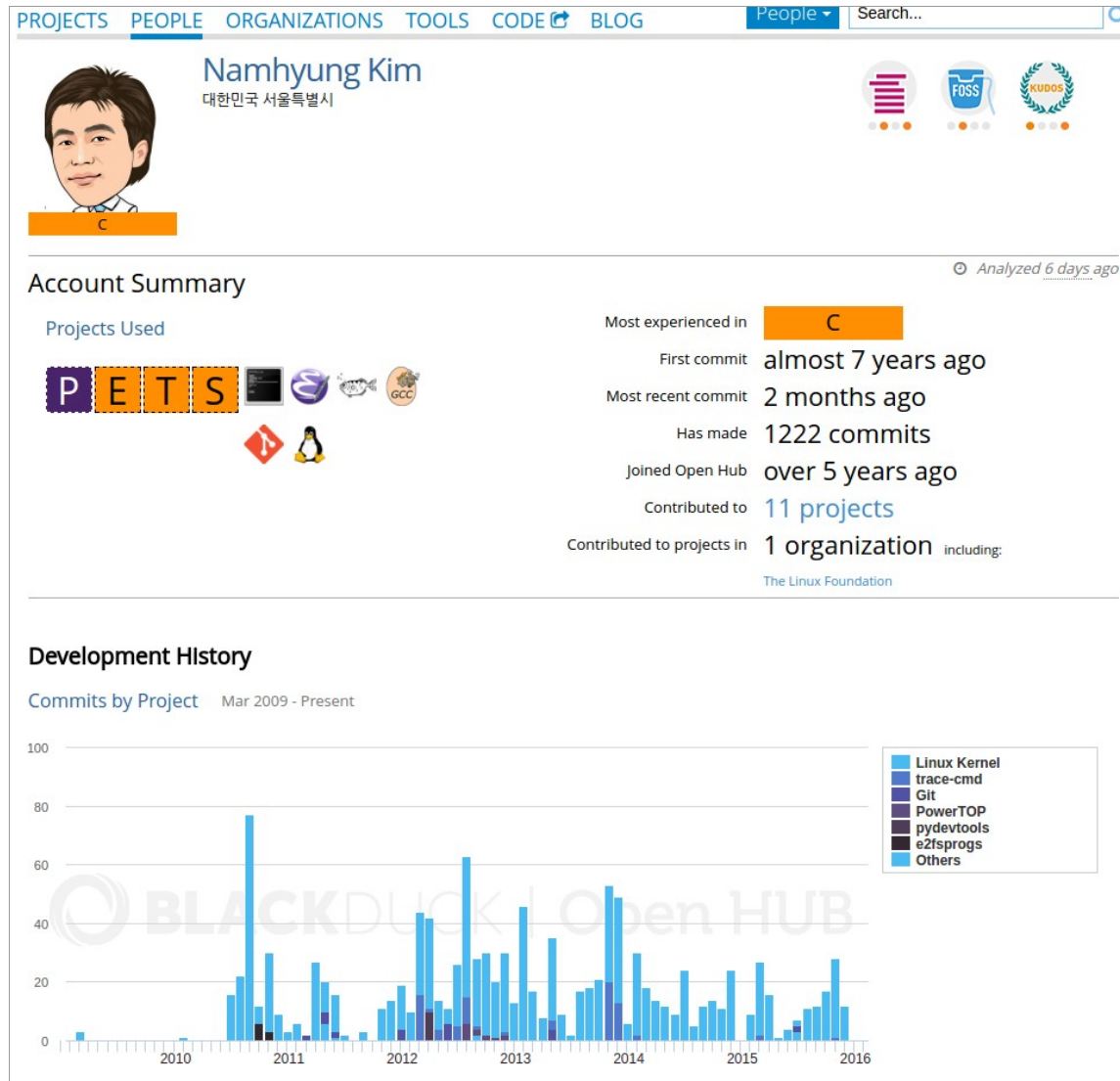
Project	Commits
Alfresco Content Management Community Edition	19909 commits
Arch Linux Packages	7343 commits
Gentoo Linux	6280 commits
azure-content	4740 commits
Mozilla Firefox	4558 commits

**Most Active Contributors**

Contributor	Commits
Felix Yan	7389 commits
Translation updater bot	1440 commits
luca suriano	1159 commits
Manfred Tournon	546 commits
Mischa ter Smitten	501 commits
Ori Livneh	501 commits

전세계 각종 오픈소스 통계 사이트 (<https://www.openhub.net/>)

# Opensource 각종 오픈소스에 기여 통계

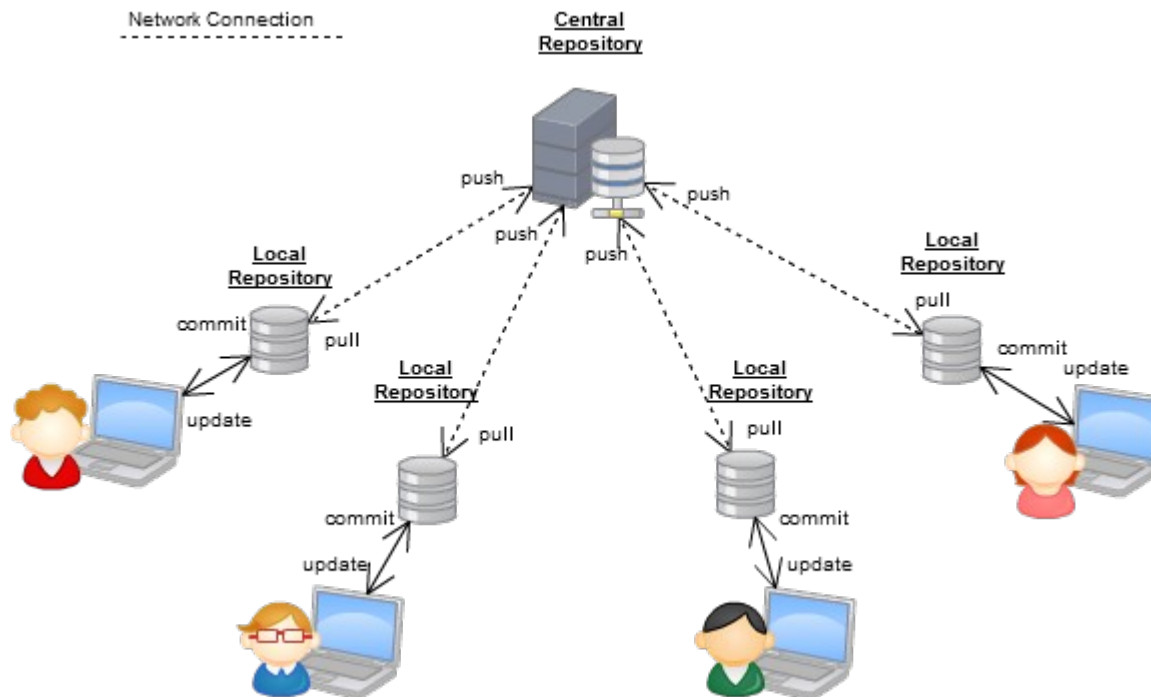


오픈소스 커미터 개인별 통계 뷰 (<https://www.openhub.net/accounts/namhyung>)



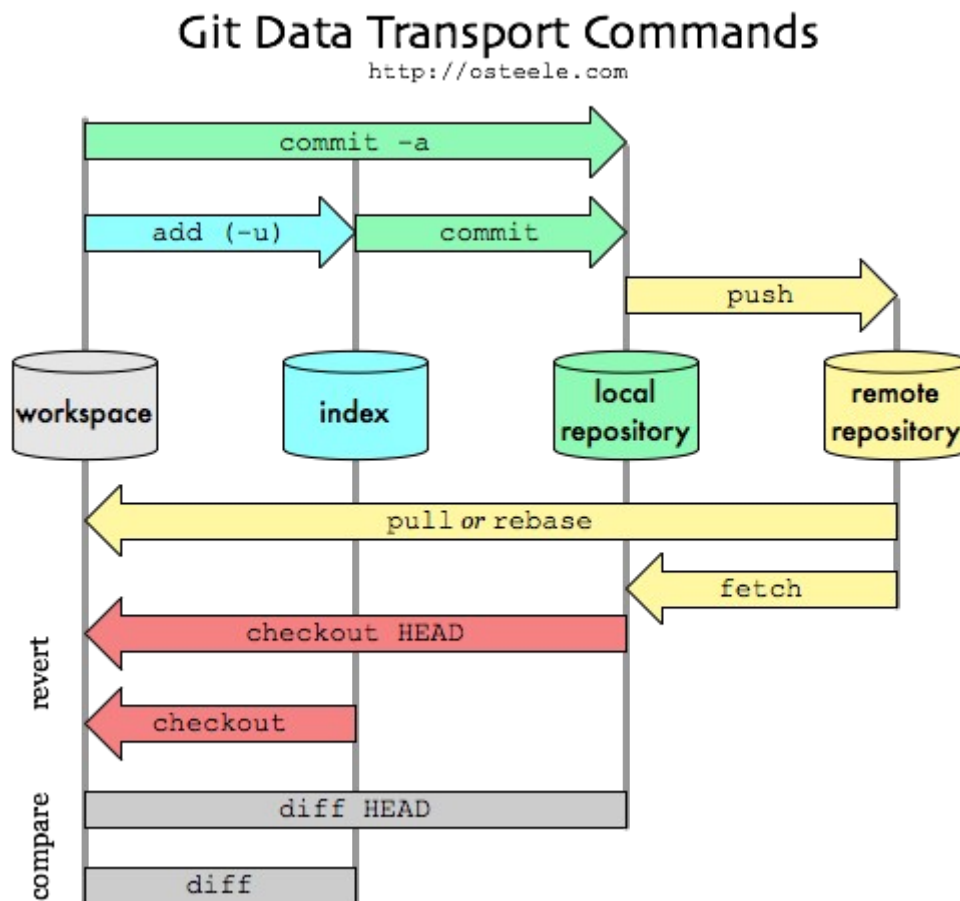
# Git 이해하기

이제는 이 그림을 이해할 수 있다 . (Git 을 통한 작업 흐름 )



# Git 이해하기

이제는 이 그림도 이해할 수 있다. (Git 을 통한 작업 흐름)



# Q&A - 자주 묻는 질문들과 간단한 대답

---

## 1) Git 과 Github 의 차이는 ?

Git 은 각 컴퓨터 (**local**) 에 설치되어 소스코드관리가 가능한 프로그램이고 Github 는 **remote** 저장소가 있는 외부서버를 지칭한다 .

## 2) Commit 과 Push 의 차이는 ?

commit 은 **local** 작업폴더에 history 를 쌓는것이어서 외부망 (internet) 을 안쓰고 Push 는 **remote** 저장소 (Github 등 ) 에 history 를 쌓는것이어서 외부망 (internet) 이 필요하다 .

## 3) Fetch 와 Pull 의 차이는 ?

Remote 저장소 (Github 등 ) 로 부터 최신 commit 정보들을 가져오는것은 매한가지이나 Fetch 는 가져와서 **임시폴더** (.git) 에 **저장**하고 Pull 은 바로 현재 branch 에 **merge 작업을 동반**한다 .

## 4) Rebase 와 Merge 의 차이는 ?

둘다 두 branch 의 차이점 (commits) 를 합치는것은 매한가지이나 Rebase 는 합치기 전에 **되감기** (rewinding) 를 하고 Merge 는 **안하고** 합친다 .

Other questions ?