

Ionic Study

Day 2

오늘 할 것들

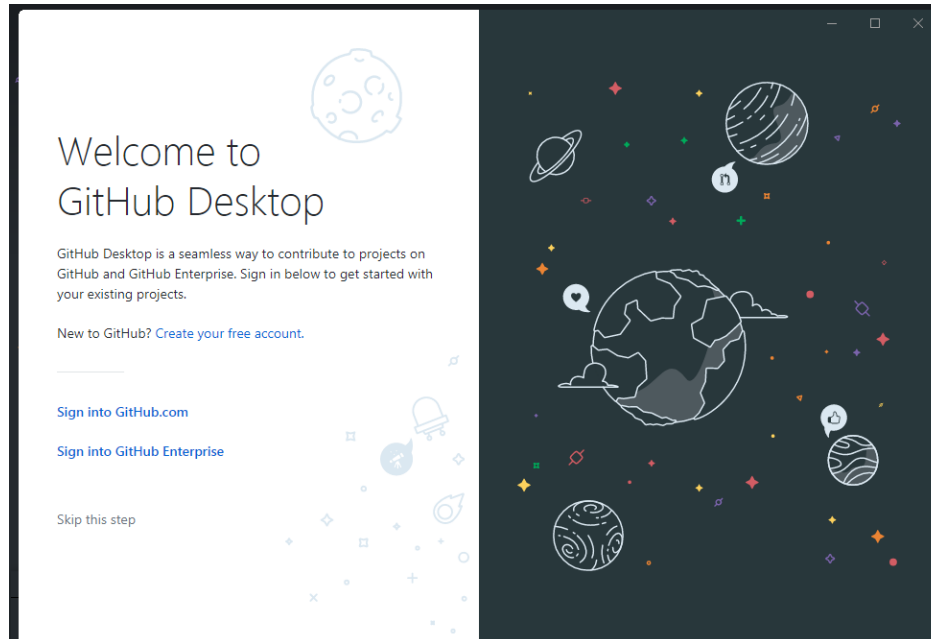
- Git과 친해져보기
- SPA (Single Page Application) 맛보기

Git

- Git 설치하기
 - Git 저장소, 클라이언트
 - 여러가지 서드파티 클라이언트들이 있다.
 - SourceTree, TortoiseGit, Github Desktop...
 - Github Desktop를 사용해보자.
 - <https://desktop.github.com/>

Git

- Git 설치하기
 - Skip this step 클릭



Git

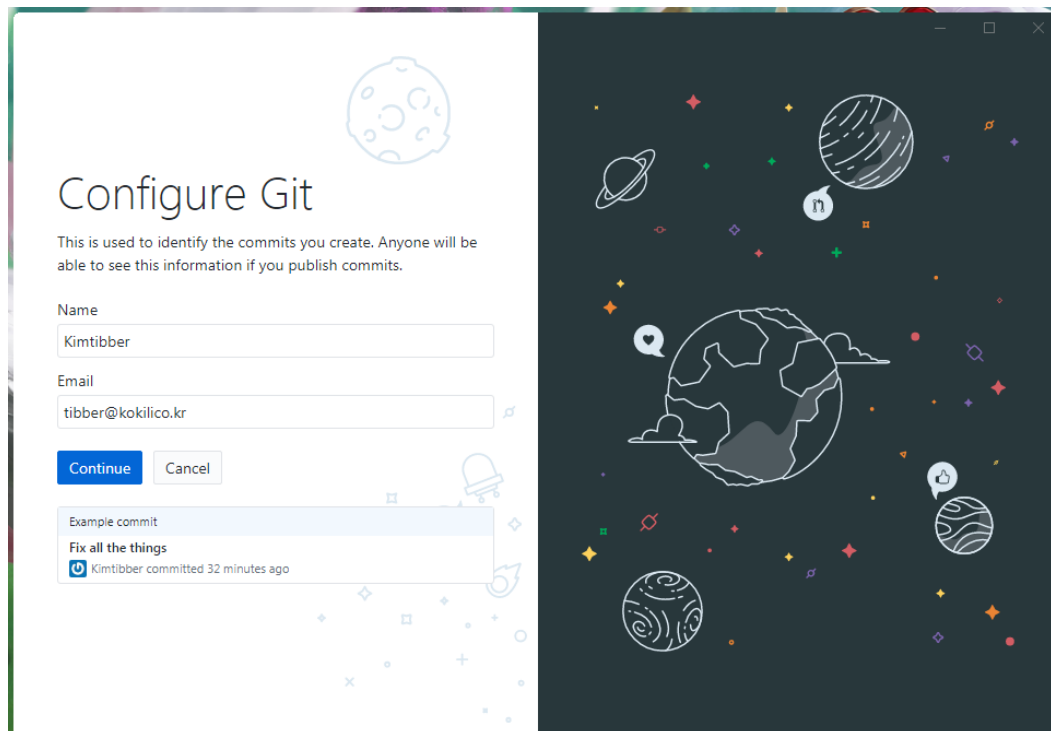
- Git 설치하기



Git

- 초기 설정

- 커밋을 위한 개인 정보 입력



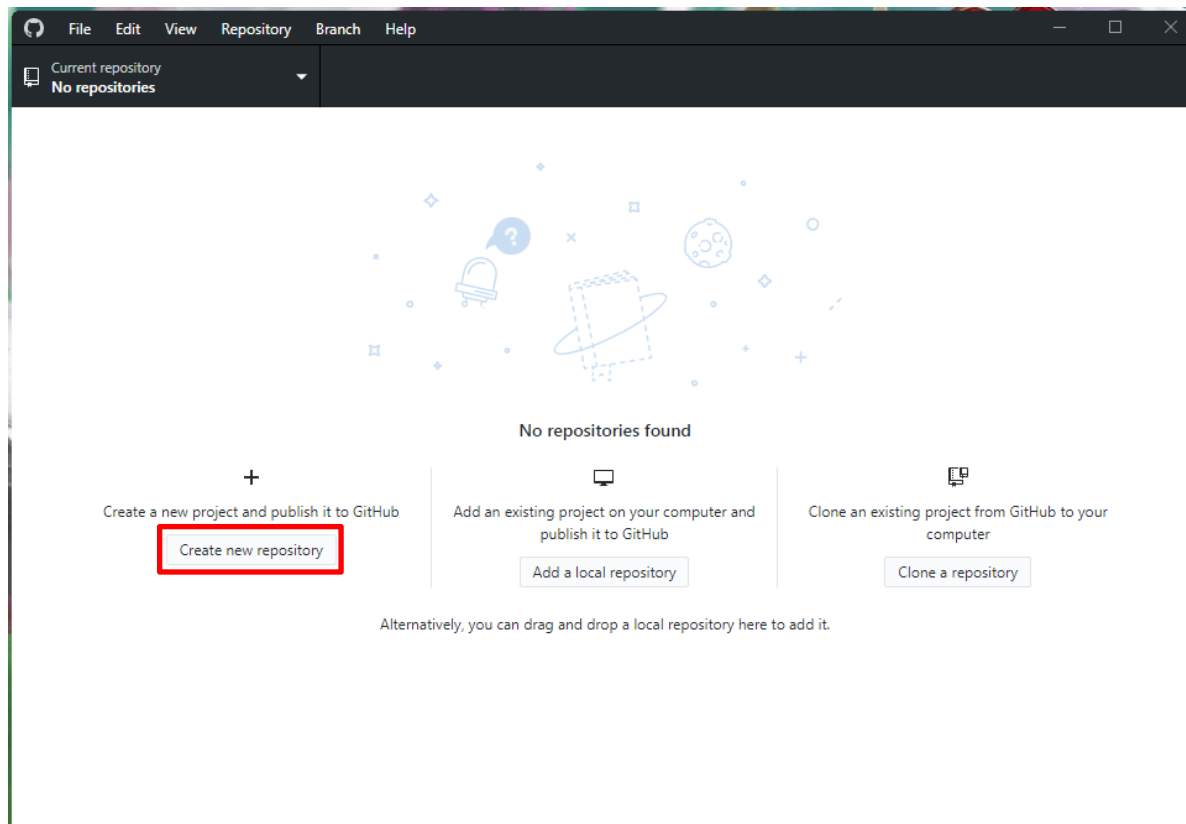
Git

- 새 저장소 만들기

- Git은 원격, 로컬 저장소를 가짐.
- 원격 저장소의 내용을 로컬 저장소와 동기화하여 사용하는 구조.
- 원격 저장소에 작업물을 올리기 위하여 로컬 저장소가 필요.

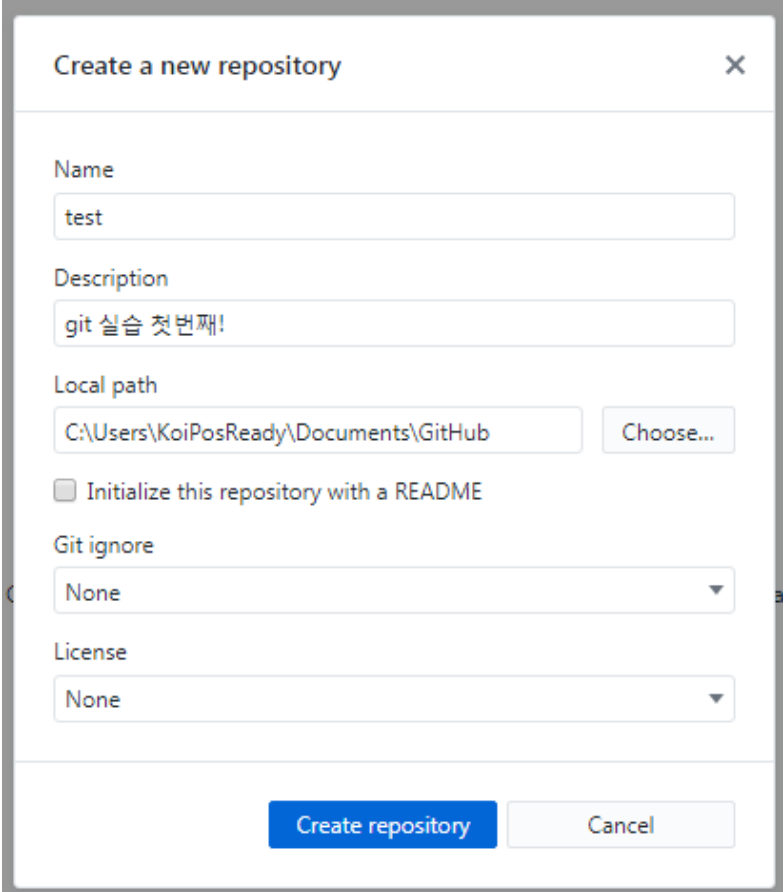
Git

- 'Create new repository' 클릭



Create a new repository

- Name: 작업 폴더 이름
- Description: 저장소 설명
- Local path: 작업 폴더가 위치할 경로
- Initialize this repository with a README: readme 파일을 만들것인지 여부.
- Git ignore: 파일 추적에서 제외할 파일들이 있을 수 있는데, 이를 언어 종류별로 선택이 가능.
- License: 이 저장소에 보관될 소스 코드의 라이선스를 지정.



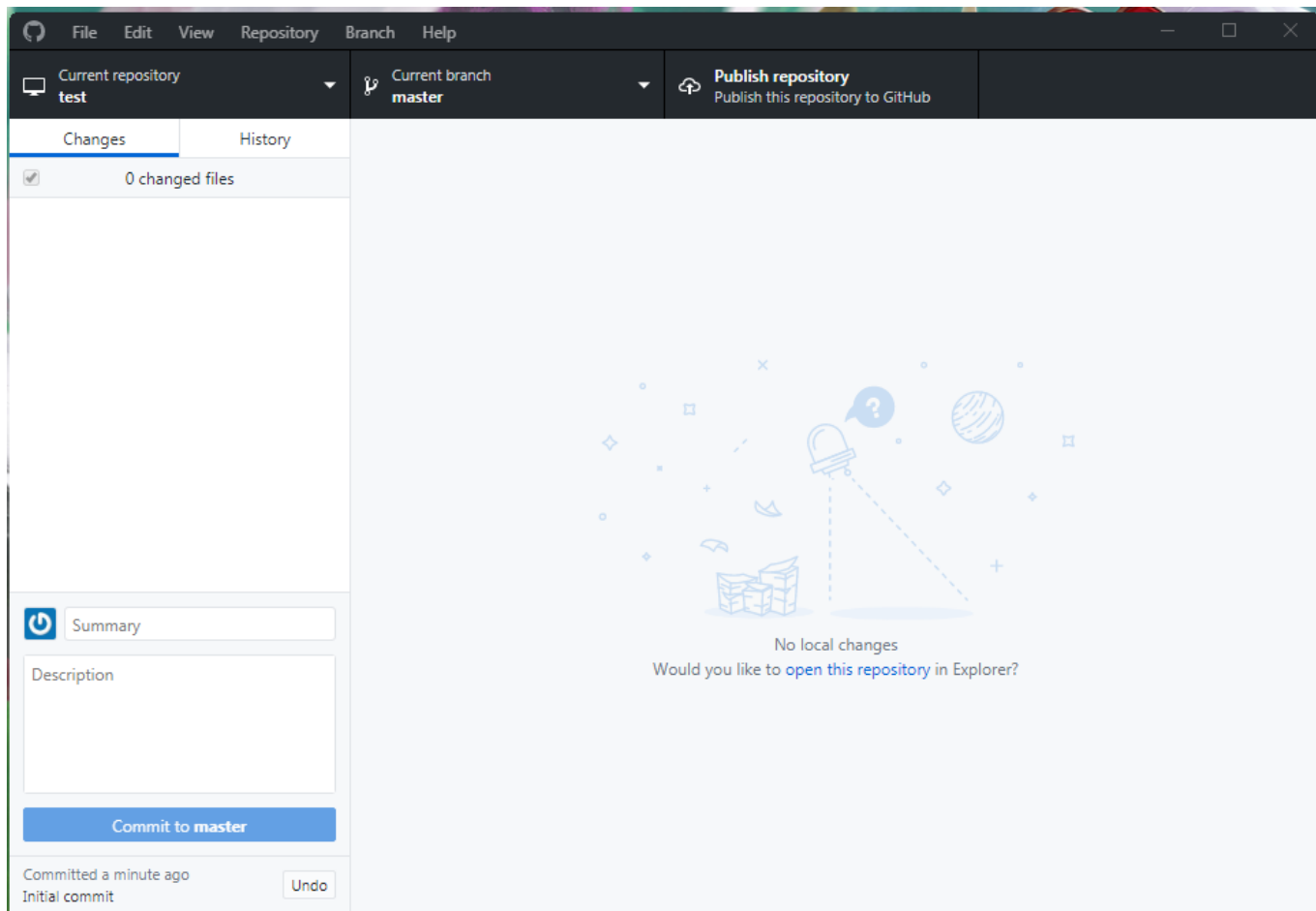
The screenshot shows a 'Create a new repository' dialog box with the following fields and options:

- Name:** test
- Description:** git 실습 첫번째!
- Local path:** C:\Users\KoiPosReady\Documents\GitHub (with a 'Choose...' button)
- ☐ Initialize this repository with a README
- Git ignore:** None (dropdown menu)
- License:** None (dropdown menu)

At the bottom, there are two buttons: 'Create repository' (in blue) and 'Cancel'.

Git

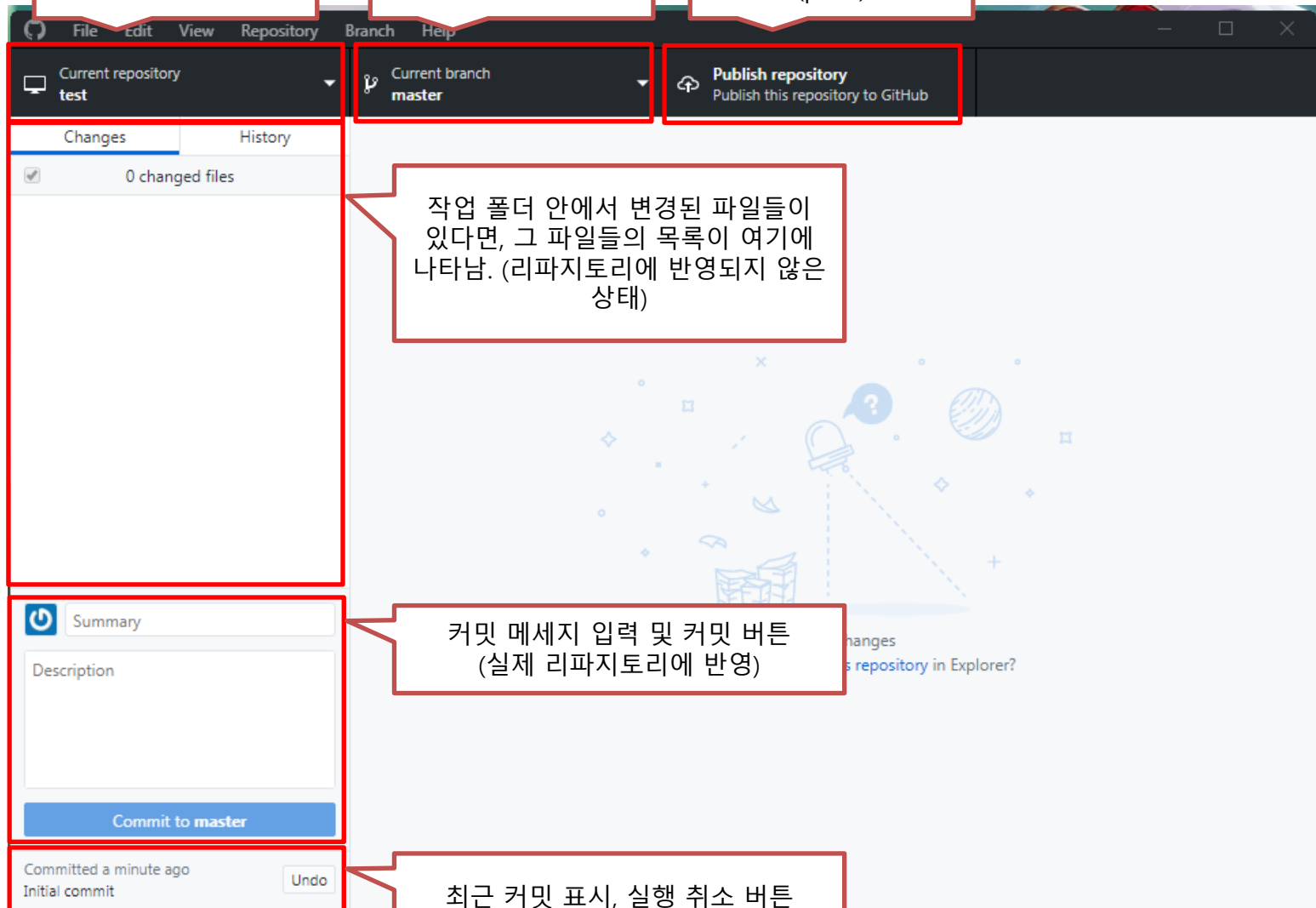
- Armed and ready!



작업중인 리파지토리

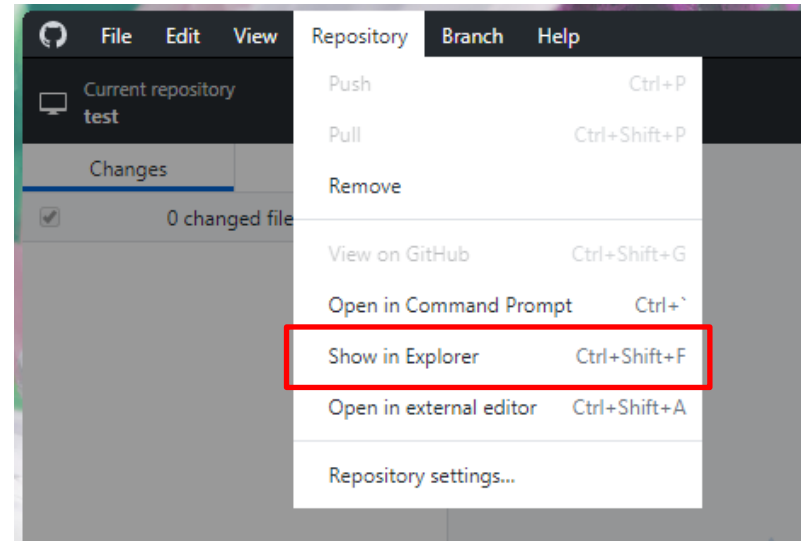
현재 브랜치

원격 저장소로 발행
(push)



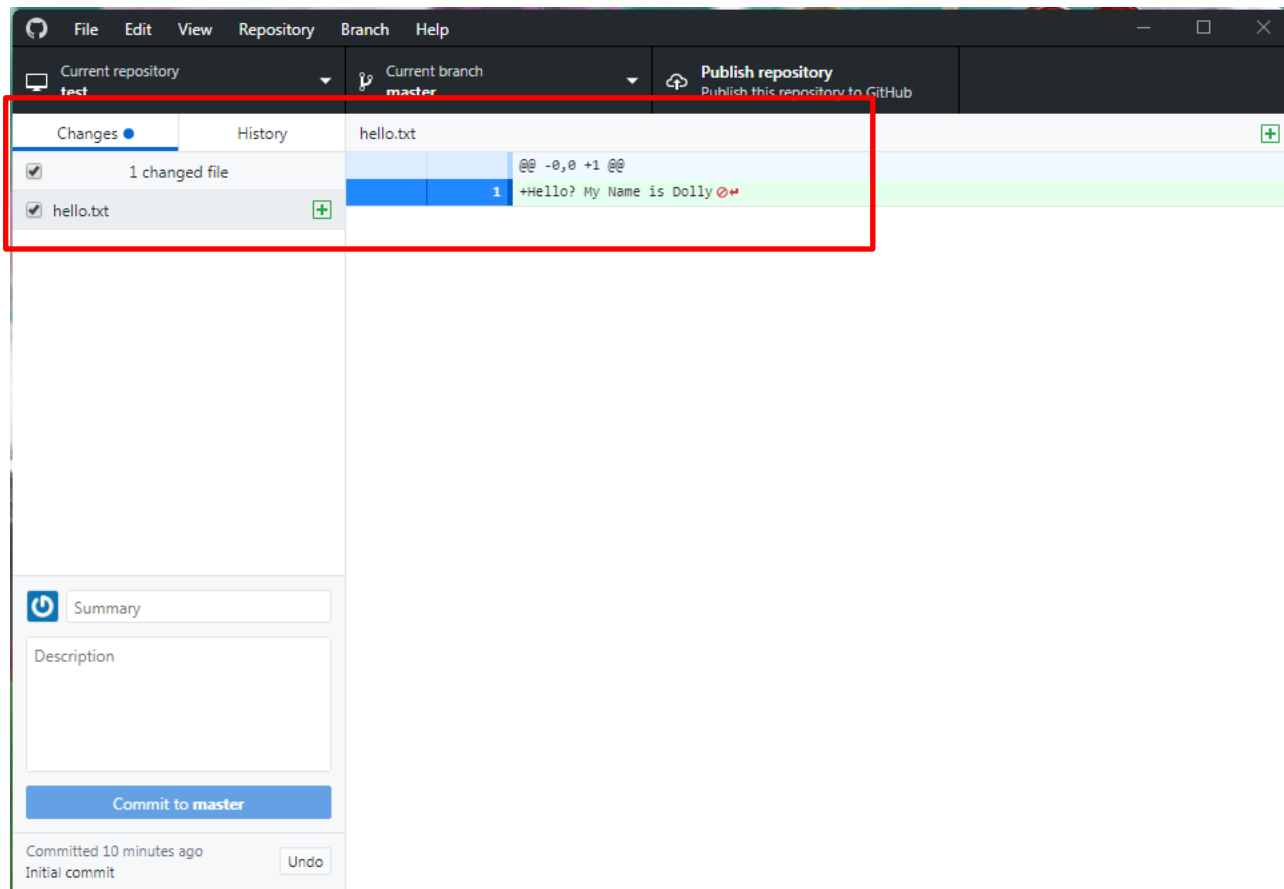
Git

- 작업 폴더에
파일을 넣어봅시다.
- [Repository] >
[Show in Explorer]



Git

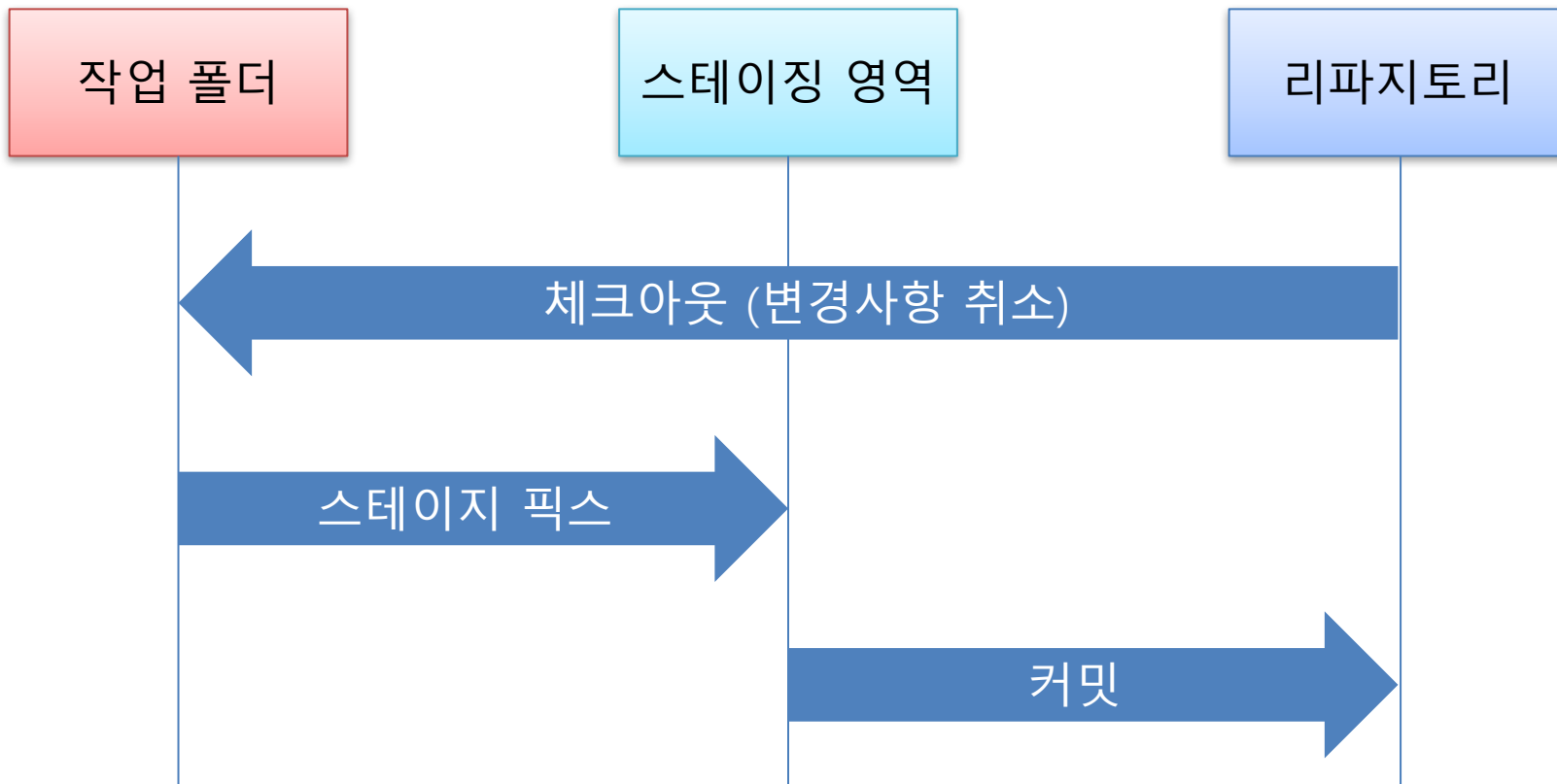
- 작업 폴더의 변경 내역을 자동으로 감지



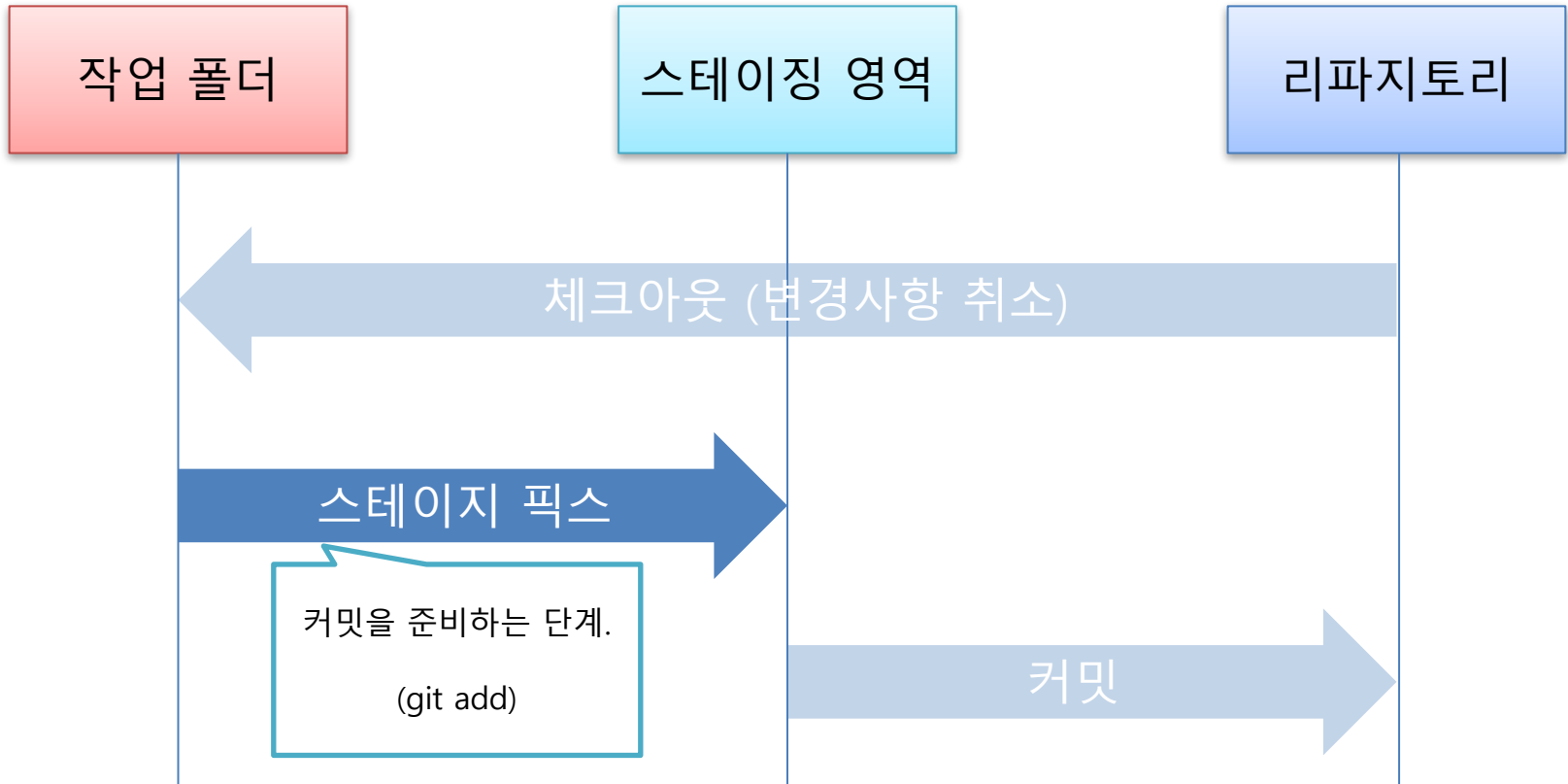
Git

- 하지만 이게 끝은 아니다?
 - 작업 폴더에서 작업을 한 것이지, 리파지토리에 반영을 한 것은 아님.
- 커밋을 해야 진짜 리파지토리에 변경 내용들이 반영됨.

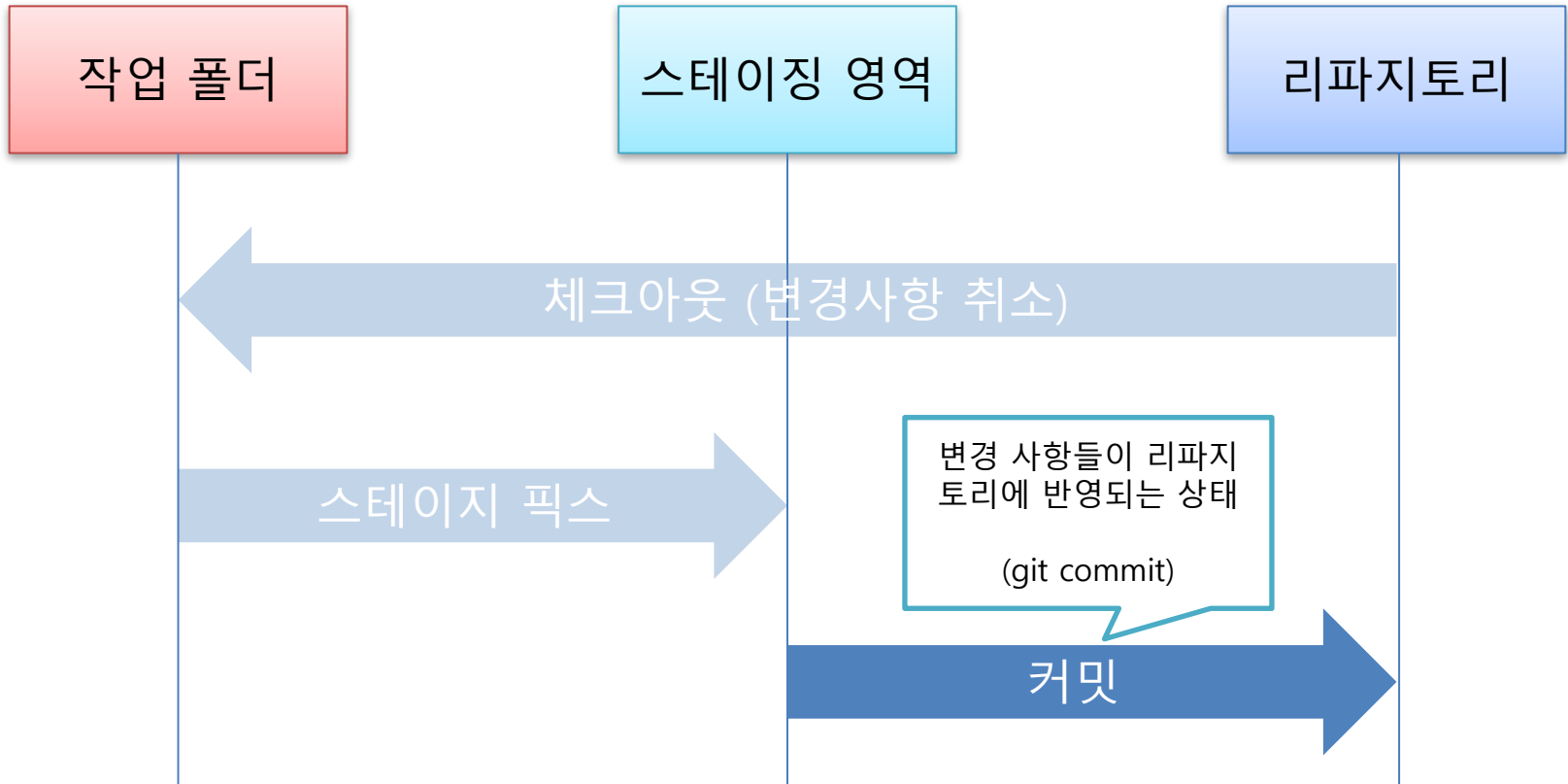
Git



Git



Git

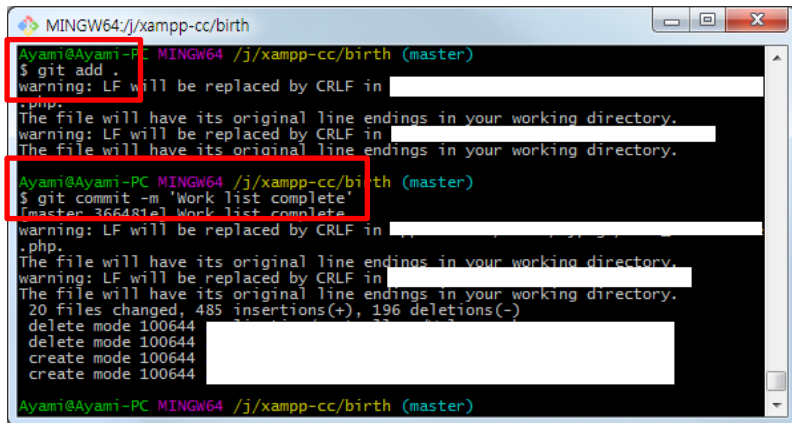


Git

- 명령 프롬프트 상에서 해당 명령어들을 입력하여 스테이징 영역에 변경사항을 올리고 커밋을 시도함.
 - `git add .`
 - `git commit -m 'Commit message'`
 - 커밋메세지를 생략하려면 `git commit`만 써준다.
- But, Github Desktop는 명령 프롬프트를 지원하지 않음.
 - 그냥 커밋 메시지 쓰고 커밋버튼을 누르면 `add`와 `commit` 두 작업을 같이 진행해줌.

Git

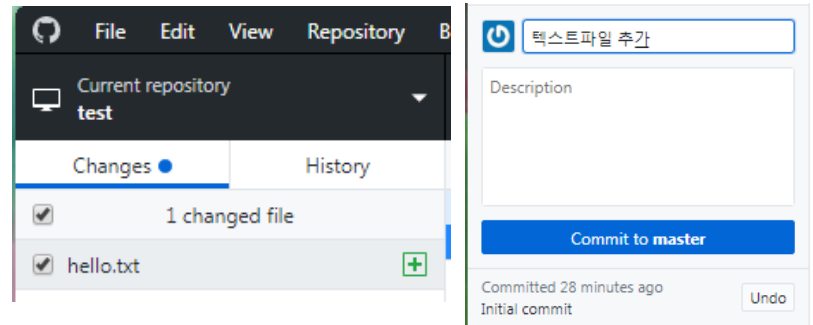
명령 프롬프트 사용



```
MINGW64 /j/xampp-cc/birth (master)
$ git add .
warning: LF will be replaced by CRLF in .
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .
The file will have its original line endings in your working directory.
$ git commit -m 'Work list complete'
[master 366481a1:1] Work list complete
warning: LF will be replaced by CRLF in .
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .
The file will have its original line endings in your working directory.
20 files changed, 485 insertions(+), 196 deletions(-)
delete mode 100644
delete mode 100644
create mode 100644
create mode 100644
Ayami@Ayami-PC MINGW64 /j/xampp-cc/birth (master)
```

- git add 명령어로 스테이징 영역에 변경 사항을 추가했고, git commit를 이용하여 스테이징 영역에 올라간 변경사항들을 레파지토리에 반영시킴

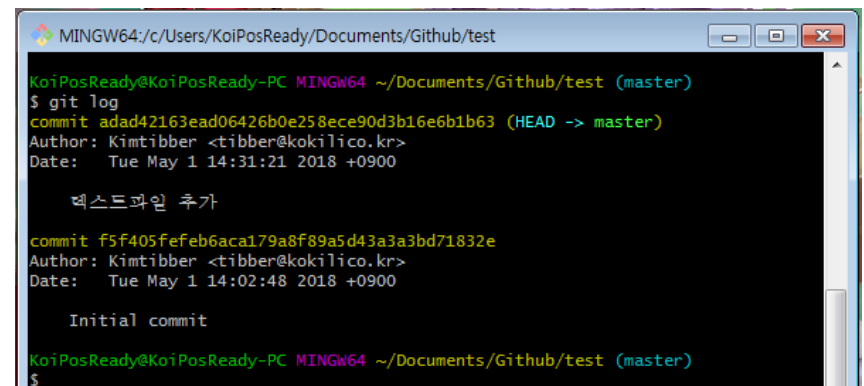
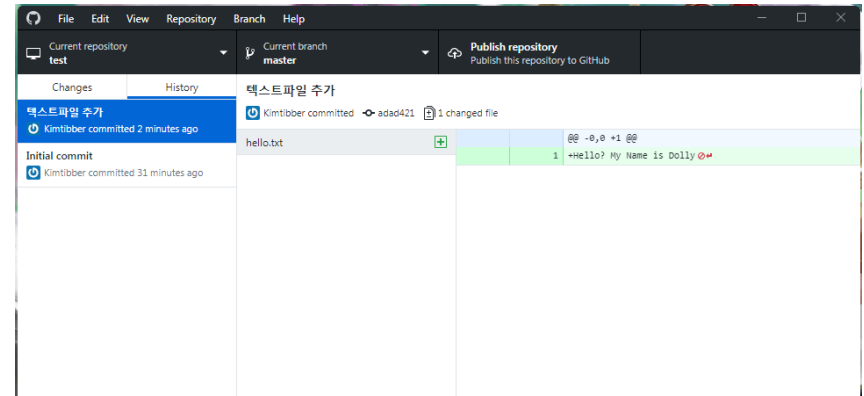
Github Desktop 사용



- Commit to master 버튼을 눌러서 즉시 커밋 완료.
- History 탭에서 커밋 이력 조회 가능

Git

- 커밋 이력 보기
 - Github Desktop
 - History 탭 클릭
 - 명령 프롬프트
 - git log 입력

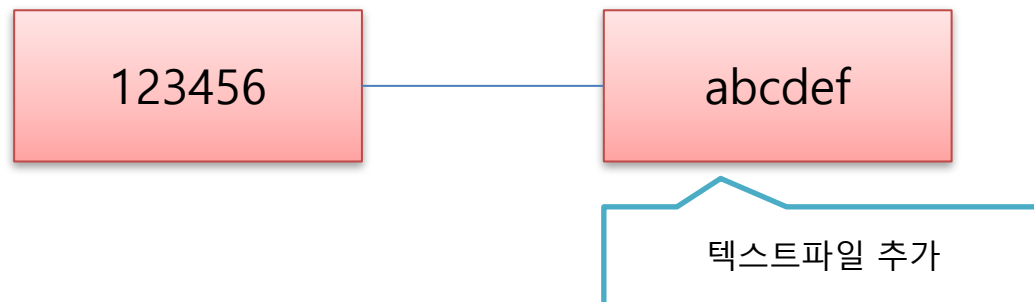


Git

- 저장소를 최초로 만들었을 때의 브랜치 상태

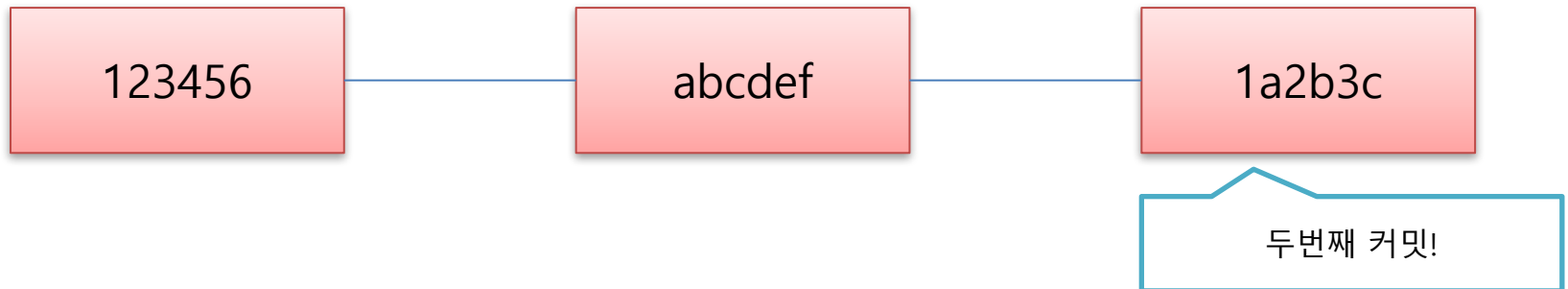


- 1회 커밋 후 브랜치의 상태



Git

- 한번 더 커밋해봅시다.
 - 아무 파일이나 작업 영역에 넣고 커밋해보기
- 현재 브랜치 상황

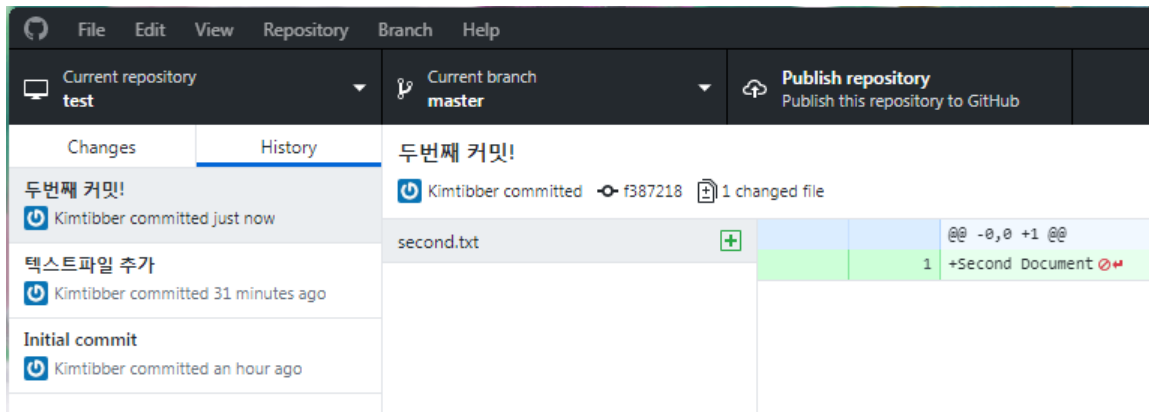


Git

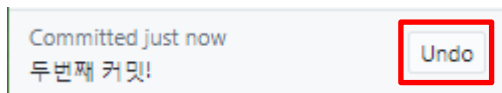
- 커밋 취소하기
 - 커밋을 잘못했을 때 취소할 수 있다.
 - Undo, Revert this commit.
 - Undo : 가장 최신 커밋을 취소
 - 최신 커밋이 사라지고, 그 이전의 커밋이 최신 커밋으로 변경
 - 새로운 커밋을 발행하지 않음.
 - Revert this commit : 해당 커밋과 변경된 내역을 취소
 - 파일을 추가했다면? : 삭제됨.
 - 파일을 삭제했다면? : 다시 복구됨.
 - 기존에 있던 파일의 내용을 수정했다면? : 변경된 내용이 취소됨.
 - 변경된 내용을 취소하는 내용으로 새로운 커밋을 발행함.

Git - Undo

- 현재 브랜치 상황

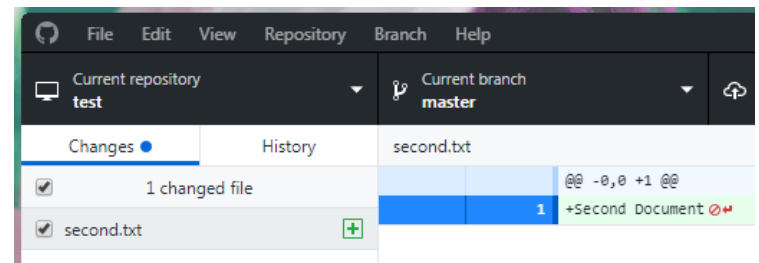
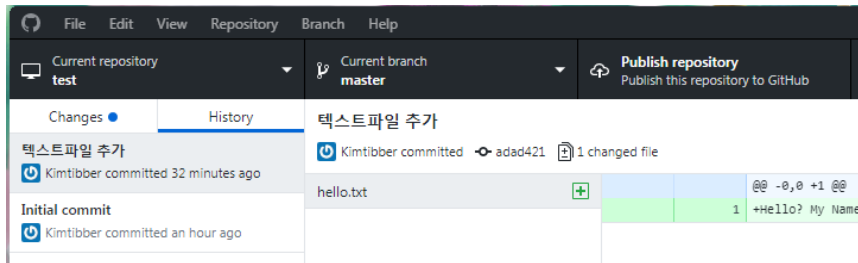


- Undo 클릭



Git - Undo

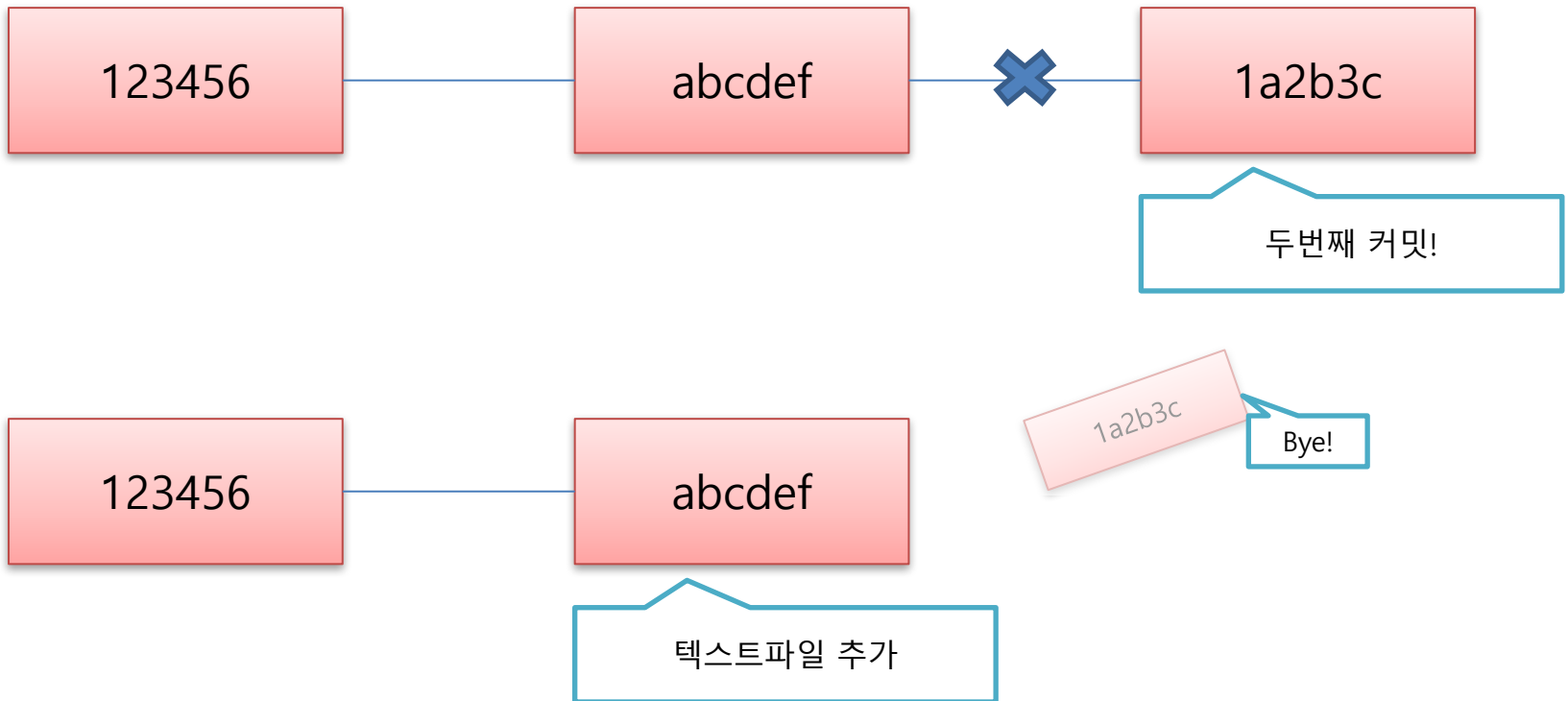
- 변경된 브랜치 상황



- '두번째 커밋!' 이 사라졌다!
 - 커밋이 취소되고, 변경된 내역이 작업 영역으로 다시 돌아옴

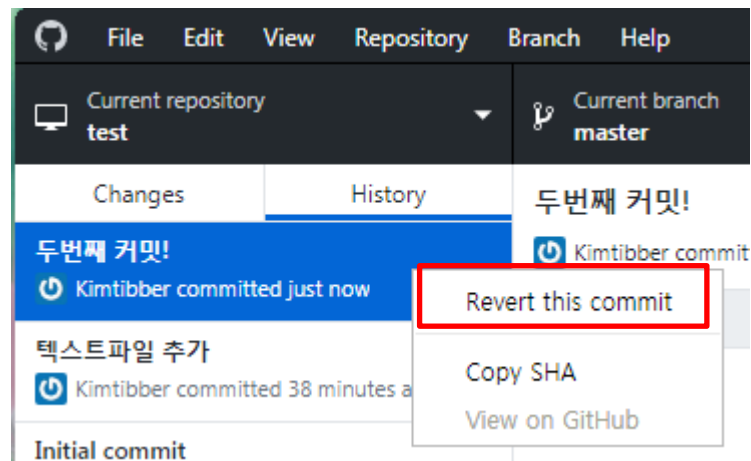
Git - Undo

- 현재 브랜치 상황



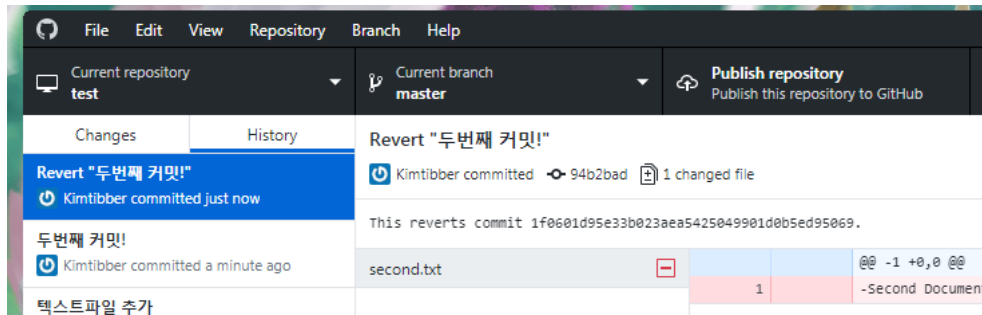
Git – Revert this commit

- Undo하여 작업영역으로 되돌아온 변경 내역을 다시 커밋.
- Revert할 커밋을 마우스 오른쪽 클릭 후, 'Revert this commit' 선택.



Git – Revert this commit

- 변경된 브랜치 상황



- 새로운 커밋이 추가되었다!
 - 해당 커밋을 취소하였음을 알리는 새로운 커밋이 자동으로 생성됨.

Git – Revert this commit

- Revert “두번째 커밋!”을 다시 Revert하면?
 - 여러분들의 상상에 맡겨볼게요.

Git 실행취소 명령어

Undo

- `git reset HEAD^`
 - 최종 커밋을 취소.
- `git reset HEAD~n`
 - 마지막 n개의 커밋을 취소

Revert this commit

- `git revert HEAD`
 - 변경한 내역을 취소하는 새로운 커밋 발행.

Git – checkout

- Github Desktop는 해당없음. (Github Desktop는 add와 commit를 한번에 수행하기 때문)
- git add 명령어로 변경 내역을 스테이지에 올렸을때 이것을 되돌리는 명령어
 - git checkout -- <파일>
- 개별 파일을 체크아웃 할 수도 있고, 전체 파일을 체크아웃할 수도 있음.
 - git checkout -- .

Git – 원격 저장소에 올리기




- Github 계정 생성
 - <https://github.com>
 - 상단의 Sign up 클릭.
 - 사용할 계정 정보 입력 후, 다음으로 넘어감.

Git – 원격 저장소에 올리기

- Unlimited public repositories for free : 퍼블릭 리파지토리를 무제한 생성할 수 있는 플랜 (무료)
 - 퍼블릭 리파지토리 : 외부에서 누구나 접근 가능한 리파지토리 – 오픈소스를 공개할 때 사용
- Unlimited private repositories for \$7/month : 개인 리파지토리를 무제한 생성할 수 있는 플랜 (월 7\$)
 - 프라이빗 리파지토리 : 리파지토리 소유주 본인이나 소유주가 초대한 사용자 외에는 접근이 불가능한 리파지토리 – 회사 프로젝트등에 사용
- 지금이 아니더라도 언제든지 플랜 변경이 가능함.
- 아래 두개 체크박스는 선택하지 않고 넘어감.

Welcome to GitHub

You've taken your first step into a larger world, @KimTibber.

 Completed Set up a personal account	 Step 2: Choose your plan	 Step 3: Tailor your experience
---	--	--

Choose your personal plan

☒ Unlimited public repositories for free.

☐ Unlimited private repositories for \$7/month. [\(view in KRW\)](#)

Don't worry, you can cancel or upgrade at any time.

☐ **Help me set up an organization next**
Organizations are separate from personal accounts and are best suited for businesses who need to manage permissions for many employees. [Learn more about organizations](#)

☐ **Send me updates on GitHub news, offers, and events**
Unsubscribe anytime in your email preferences. [Learn more](#)

Continue

Both plans include:

- ✓ Collaborative code review
- ✓ Issue tracking
- ✓ Open source community
- ✓ Unlimited public repositories
- ✓ Join any organization

Git – 원격 저장소에 올리기

- 묻는 설문에 성실하게 답변해주거나,
- 아니면 skip this step로 설문을 생략할 수 있음.

Welcome to GitHub

You'll find endless opportunities to learn, code, and create, @KimTibber.



Completed
Set up a personal account



Step 2:
Choose your plan



Step 3:
Tailor your experience

How would you describe your level of programming experience?

- ☐ Very experienced ☐ Somewhat experienced ☐ Totally new to programming

What do you plan to use GitHub for? (check all that apply)

- ☐ School projects ☐ Project Management ☐ Development
☐ Design ☐ Research ☐ Other (please specify)

Which is closest to how you would describe yourself?

- ☐ I'm a student ☐ I'm a hobbyist ☐ I'm a professional
☐ Other (please specify)

What are you interested in?

e.g. tutorials, android, ruby, web-development, machine-learning, open-source

Submit

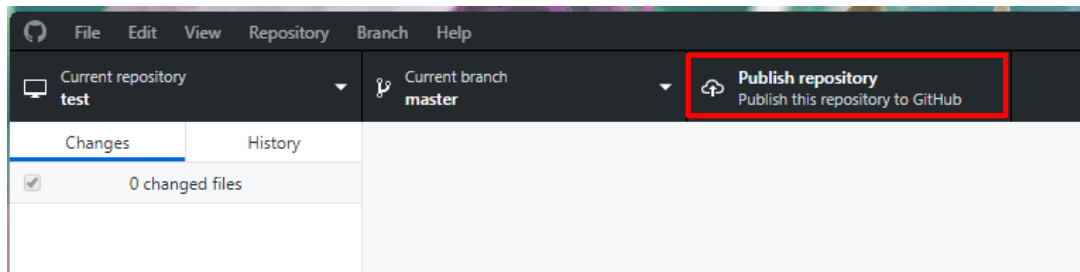
[skip this step](#)

Git – 원격 저장소에 올리기

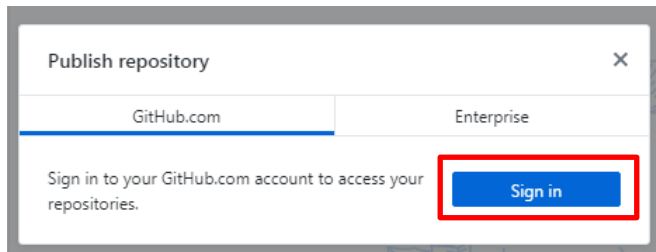
- 공짜로 프라이빗 저장소를 가지고 싶어요!
 - Github의 학생용 라이선스를 이용 (영문 재학증명서를 보내줘야 하며, 입학년도와 졸업예정년도를 정확하게 알려줘야 플랜이 변경됨.)
 - 다른 원격 저장소 사용
 - Bitbucket, Gitlab, Visual Studio Team Foundation Server
 - 사용자 수 제한이 있음. 아마 5명까지는 무료로 프라이빗 저장소를 쓸 수 있던걸로 기억...

Git – 원격 저장소에 올리기

- Publish repository 클릭

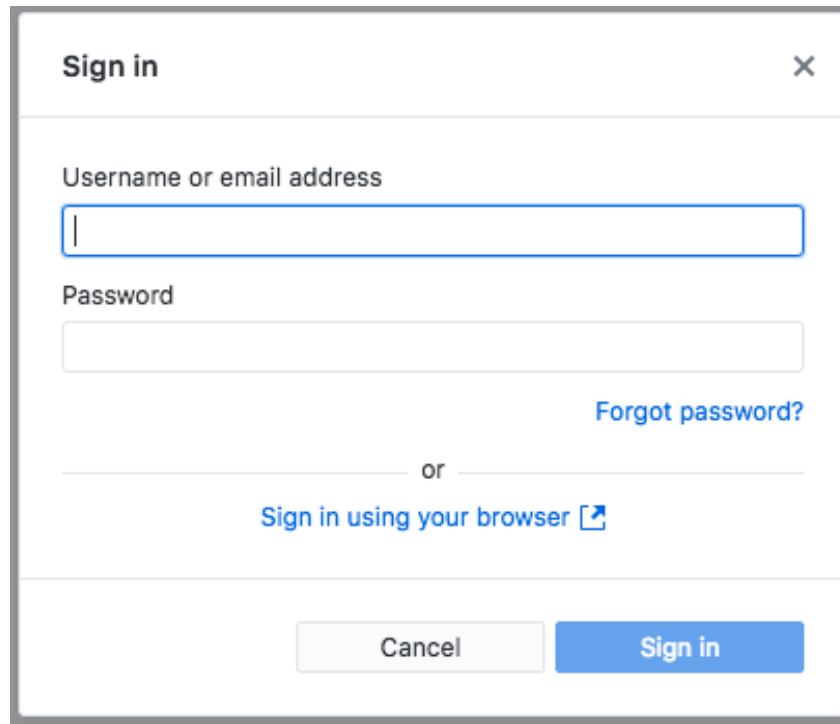


- Github.com 클릭 후, Sign in 버튼 클릭



Git – 원격 저장소에 올리기

- 앞 단계에서 생성한 Github 계정정보 입력 후,
Sign in 버튼 클릭



A screenshot of the Github 'Sign in' dialog box. The dialog has a title bar with 'Sign in' and a close button (X). Below the title bar, there are two input fields: 'Username or email address' and 'Password'. The 'Username or email address' field is currently empty with a cursor. Below the password field, there is a link 'Forgot password?'. Below this, there is a horizontal line with the word 'or' in the center. Below the line, there is a link 'Sign in using your browser' with an external link icon. At the bottom of the dialog, there are two buttons: 'Cancel' and 'Sign in'.

Sign in

Username or email address

Password

[Forgot password?](#)

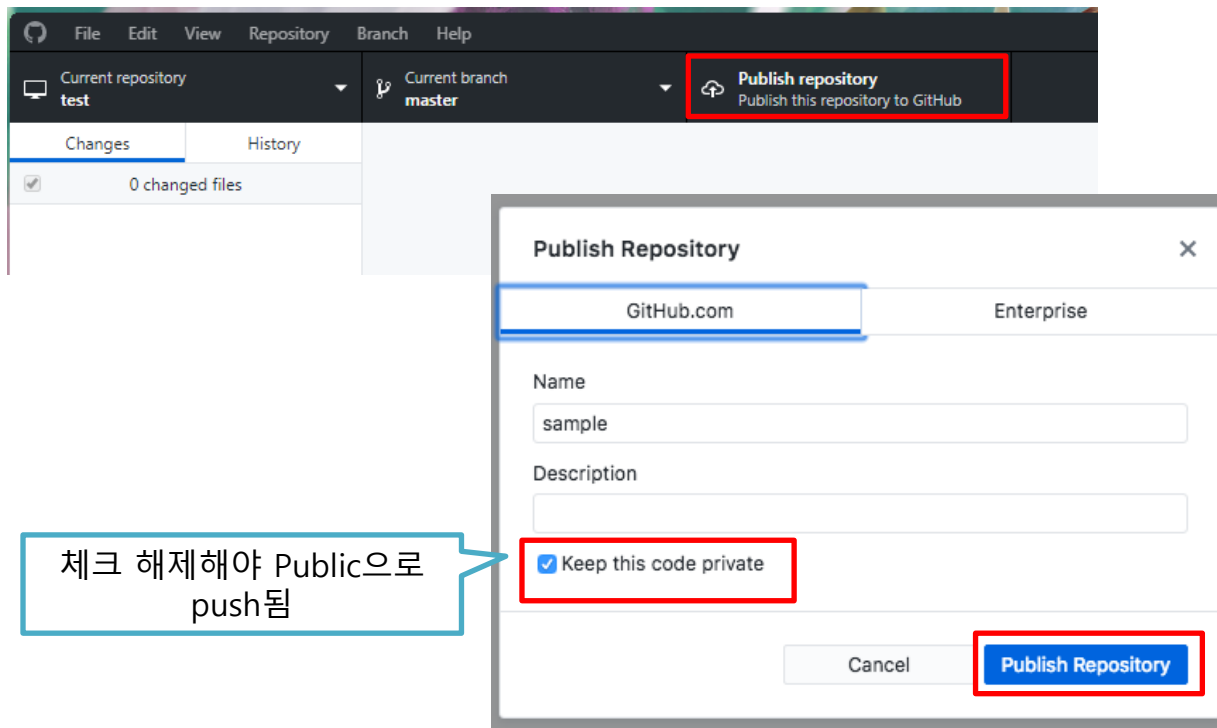
or

[Sign in using your browser](#)

Cancel Sign in

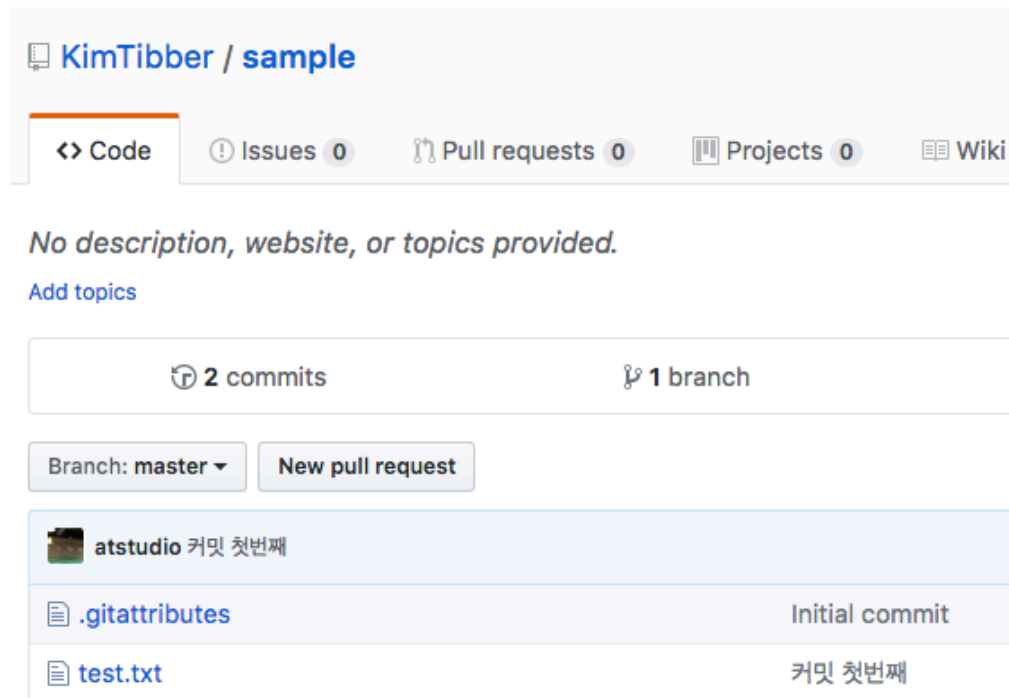
Git – 원격 저장소에 올리기

- 다시 한 번 Publish repository 클릭 후, [Publish Repository] 버튼 클릭



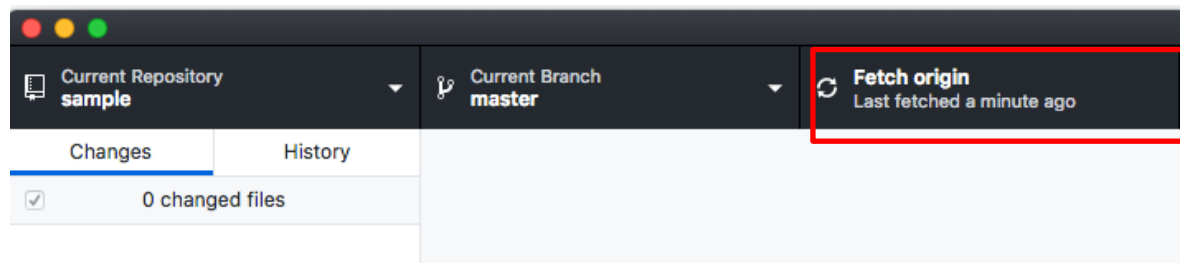
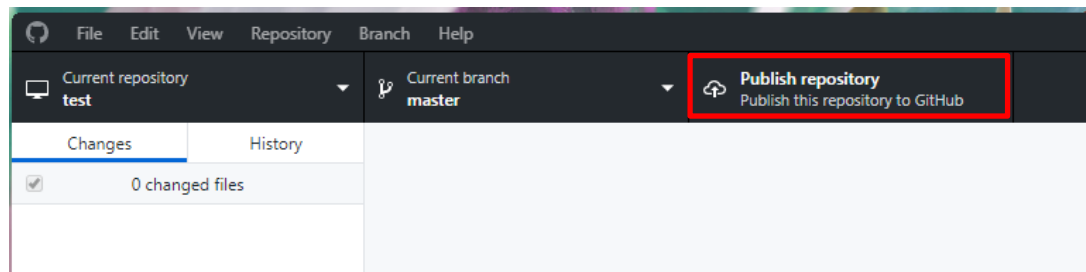
Git – 원격 저장소에 올리기

- `https://github.com/<자신의 Github ID>/리포지토리명` 을 입력하면 방금 Publish한 리포지토리를 확인가능.



Git – 원격 저장소에 올리기

- 한번 Publish한 로컬 리파지토리는 기존의 Publish repository 버튼이 사라지고, Fetch origin 버튼으로 바뀐다



Git – 원격 저장소에서 받아오기

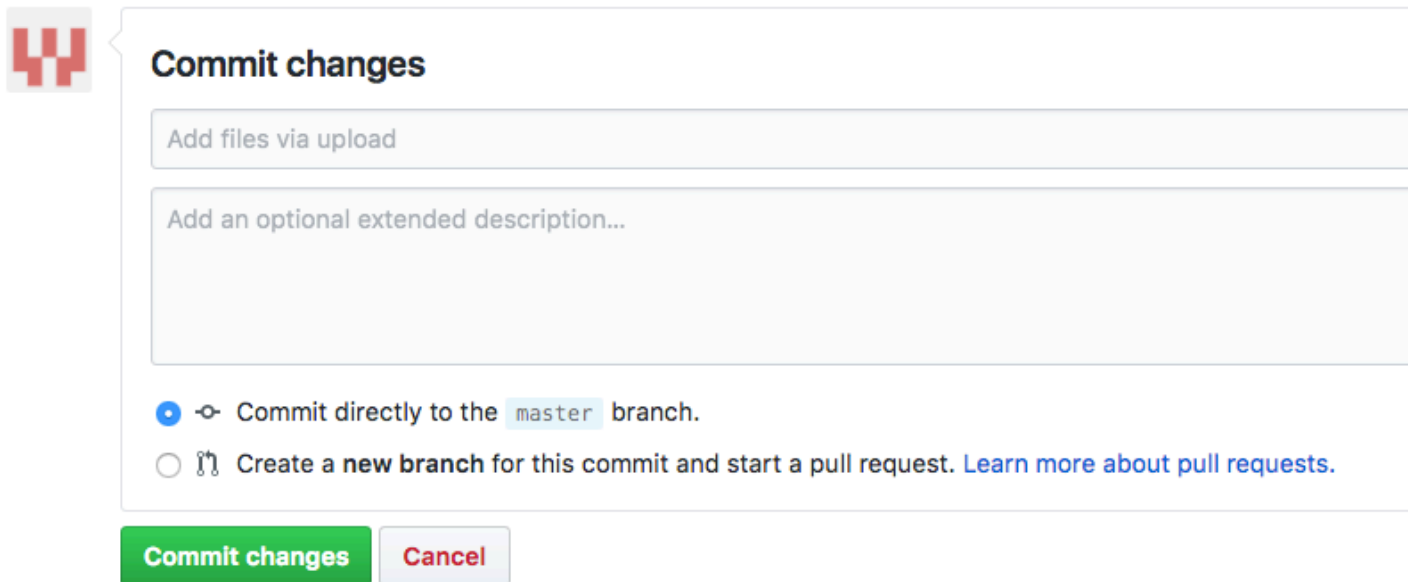
- Fetch origin
 - 원격 저장소에서 내용을 받아오는 명령
 - 원격 저장소와 로컬 저장소간의 변경사항을 동기화할 때 사용.

Git – 원격 저장소에서 받아오기

- 앞서 Publish한 원격 리파지토리에 접속
 - <https://github.com/<자신의 Github ID>/리파지토리명>
- 로그인 후, [Upload file] 클릭.
 - 외부에 공개되어도 상관없는 파일을 하나 임의로 선택
 - 커밋 메시지 입력

Git – 원격 저장소에서 받아오기

- Commit Changes
 - Commit directly to master branch : 마스터 브랜치에 직접 커밋
 - Create a new branch for this commit and start a pull request : 새로운 브랜치를 만들고 그 브랜치에 이 커밋을 포함.



The image shows a Git commit dialog box. On the left is a red icon with white 'W' shapes. The main area is titled 'Commit changes'. It contains a text input field with the placeholder 'Add files via upload'. Below that is a larger text area with the placeholder 'Add an optional extended description...'. At the bottom, there are two radio button options. The first option is selected and has the text 'Commit directly to the master branch.' The second option is unselected and has the text 'Create a new branch for this commit and start a pull request. Learn more about pull requests.' At the very bottom are two buttons: a green 'Commit changes' button and a grey 'Cancel' button.

Commit changes

Add files via upload

Add an optional extended description...




☒ Commit directly to the master branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

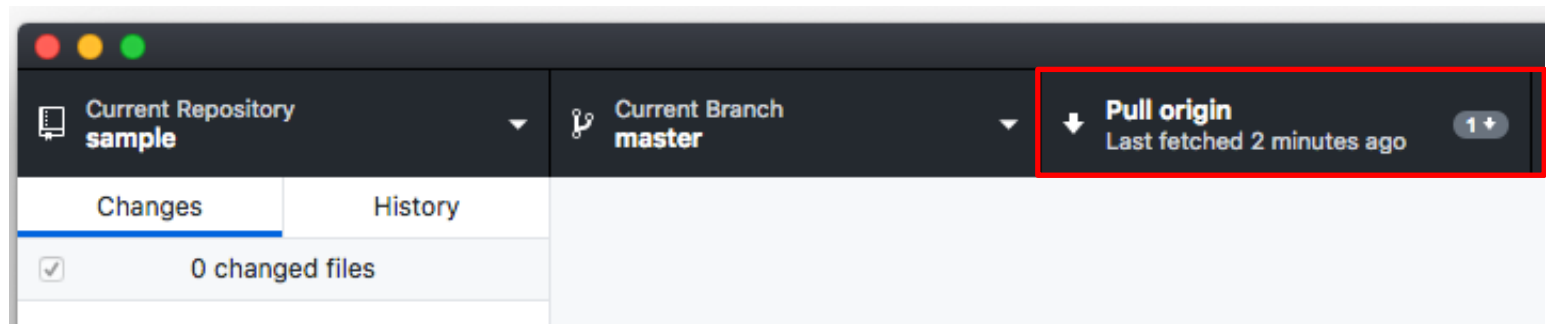
Git – 원격 저장소에서 받아오기

- 업로드한 파일이 원격 저장소에 반영됨.
- 새로운 커밋이 생성됨.

KimTibber Add files via upload		Latest commit 493f046 a minute ago
 .gitattributes	Initial commit	16 minutes ago
 4bd50d61b27f3_kimhuidong.jpg	Add files via upload	a minute ago
 test.txt	커밋 첫번째	15 minutes ago

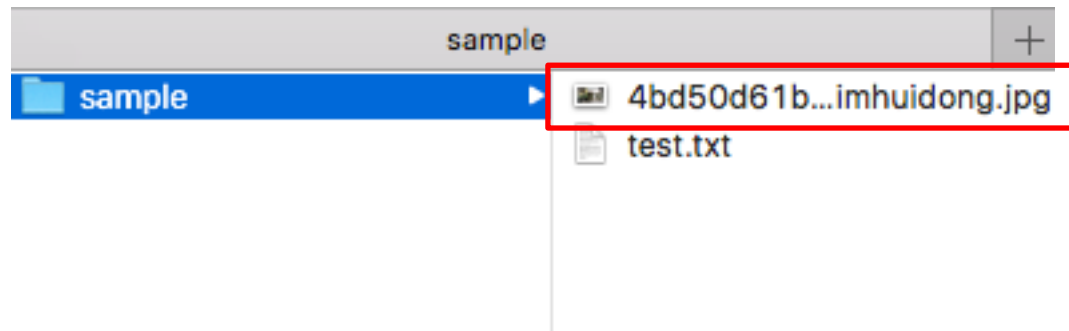
Git – 원격 저장소에 올리기

- 원격 저장소에서 변경사항이 발생하면, Github Desktop가 이를 자동으로 감지하여 Fetch origin 버튼을 Pull origin 버튼으로 바꿔줌
- Pull origin 클릭.



Git – 원격 저장소에 올리기

- 동기화가 완료됨.
- 로컬 작업폴더에 Github.com에서 직접 업로드한 파일이 생성되어있음을 확인.



Single Page Application 맛보기

- Single Page Application?
 - 단일 화면에서 모든 처리가 이루어지는 어플리케이션

Single Page Application 맛보기

- 프로젝트 생성

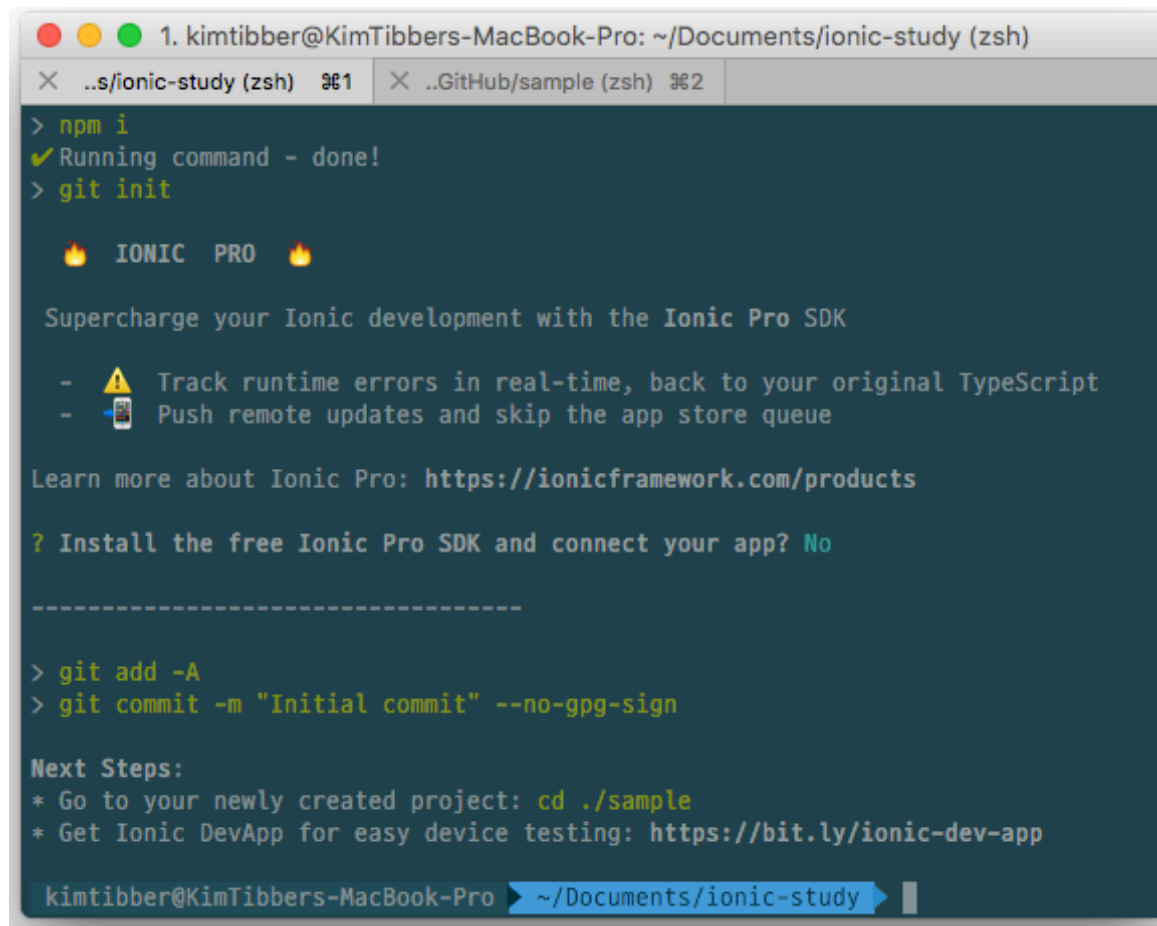
- Ionic 3 CLI는 새로운 프로젝트를 생성하기 위해 start라는 명령어를 지원
 - `ionic start [<프로젝트명>] [<템플릿>] [<옵션>]`
- sample라는 이름의 단일 페이지로 구성된 프로젝트를 생성하려면?
 - `ionic start sample blank`
 - blank는 템플릿의 한 종류. 한 페이지(Single Page)로 구성된 앱(App)을 디폴트로 생성하는 템플릿

Single Page Application 맛보기

- Git 작업 폴더 안에서 새로운 프로젝트를 생성
- ionic start sample blank
 - Would you like to integrate your new app with Cordova to target native iOS and Android? - 새로 만드는 앱을 Cordova를 사용하여 네이티브 iOS와 안드로이드에서 사용하겠는지 묻는 옵션.
 - Install the free Ionic Pro SDK and connect your app? - Ionic Pro SDK와 생성하는 앱을 연동시킬것인지 묻는 옵션.

Single Page Application 맛보기

- 자동으로 Git 리파지토리까지 생성해줌.



```
1. kimtibber@KimTibbers-MacBook-Pro: ~/Documents/ionic-study (zsh)
X ..s/ionic-study (zsh) %1 X ..GitHub/sample (zsh) %2
> npm i
✓Running command - done!
> git init

🔥 IONIC PRO 🔥

Supercharge your Ionic development with the Ionic Pro SDK

- ⚠ Track runtime errors in real-time, back to your original TypeScript
- 📱 Push remote updates and skip the app store queue

Learn more about Ionic Pro: https://ionicframework.com/products

? Install the free Ionic Pro SDK and connect your app? No

-----

> git add -A
> git commit -m "Initial commit" --no-gpg-sign

Next Steps:
* Go to your newly created project: cd ./sample
* Get Ionic DevApp for easy device testing: https://bit.ly/ionic-dev-app

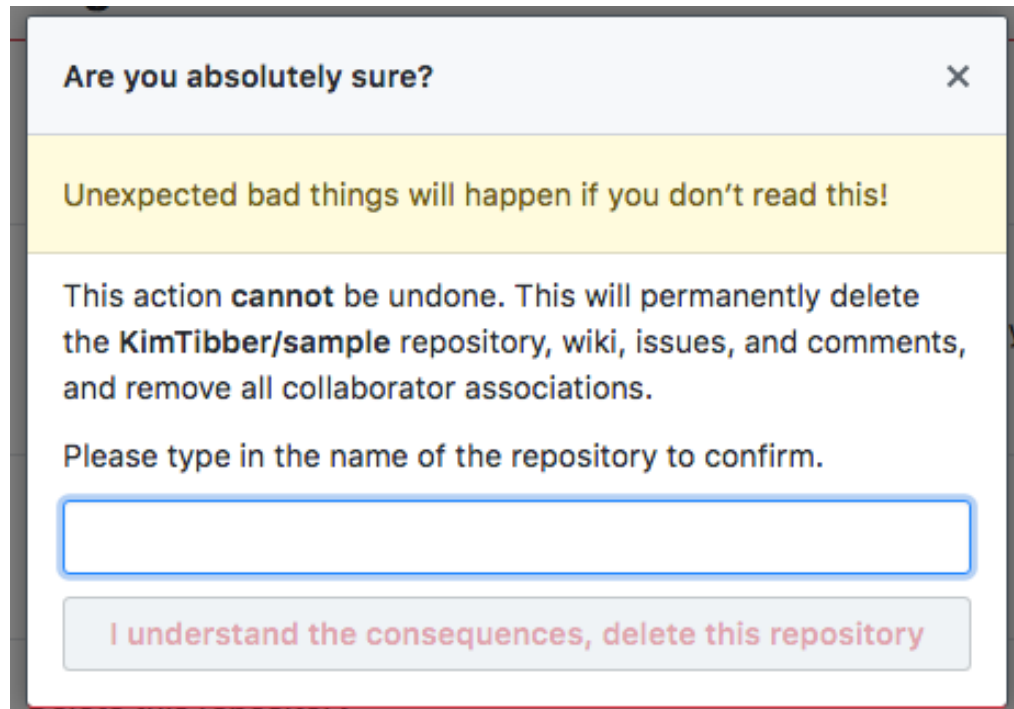
kimtibber@KimTibbers-MacBook-Pro ~/Documents/ionic-study
```

Single Page Application 맛보기

- sample 프로젝트와 Github 연동하기.
 - 기존에 실습을 위해 만든 테스트 리파지토리 삭제하기.
 - <https://github.com/<자신의 Github ID>/리파지토리명>
 - [Settings] 탭 클릭
 - 아래로 쪽 내리면 Danger Zone가 있음. [Delete this repository] 버튼 클릭.

Single Page Application 맛보기

- 테스트 리파지토리 삭제하기
 - 해당 리파지토리명을 다시 한 번 입력하고 아래 버튼 클릭하면 삭제됨.



A confirmation dialog box with a light blue header bar containing the text "Are you absolutely sure?" and a close button (X). Below the header is a yellow warning bar with the text "Unexpected bad things will happen if you don't read this!". The main body of the dialog is white and contains the following text: "This action **cannot** be undone. This will permanently delete the KimTibber/sample repository, wiki, issues, and comments, and remove all collaborator associations." followed by "Please type in the name of the repository to confirm." Below this text is a text input field with a blue border. At the bottom of the dialog is a button with a light blue background and red text that reads "I understand the consequences, delete this repository".

Are you absolutely sure? ×

Unexpected bad things will happen if you don't read this!

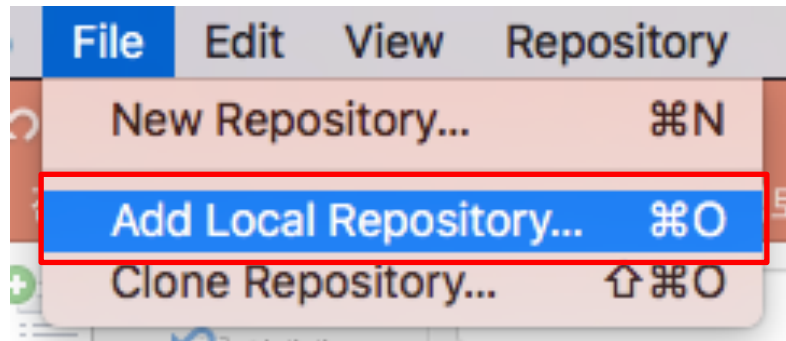
This action **cannot** be undone. This will permanently delete the KimTibber/sample repository, wiki, issues, and comments, and remove all collaborator associations.

Please type in the name of the repository to confirm.

I understand the consequences, delete this repository

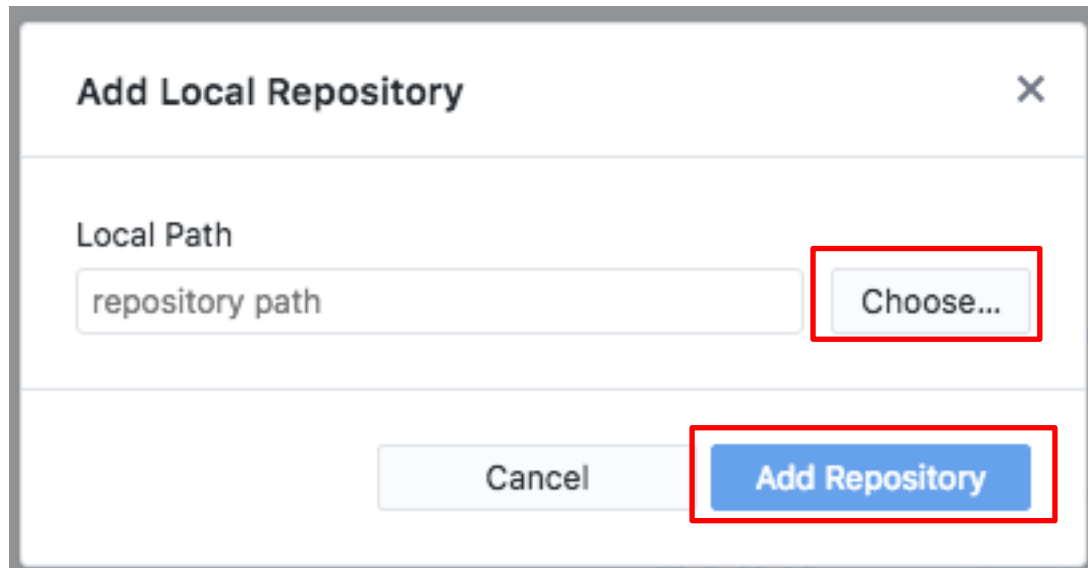
Single Page Application 맛보기

- Github Desktop에서 리파지토리 불러오기
 - [File] - [Add Local Repository...] 클릭



Single Page Application 맛보기

- sample 프로젝트가 위치한 경로 선택
 - [Choose...] 버튼으로 선택 가능.
 - [Add Repository] 클릭



Add Local Repository [X]

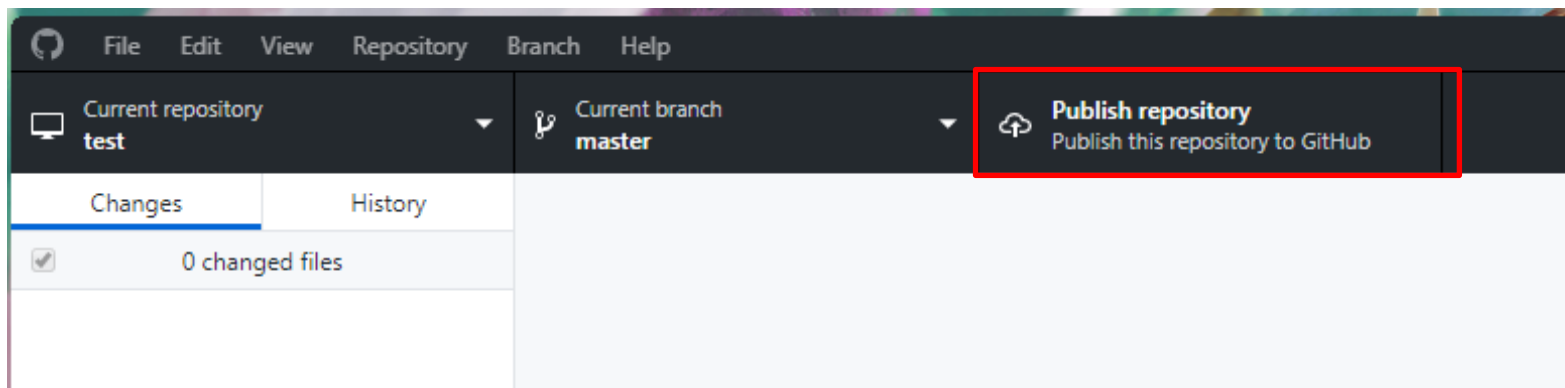
Local Path

repository path [Choose...]

[Cancel] [Add Repository]

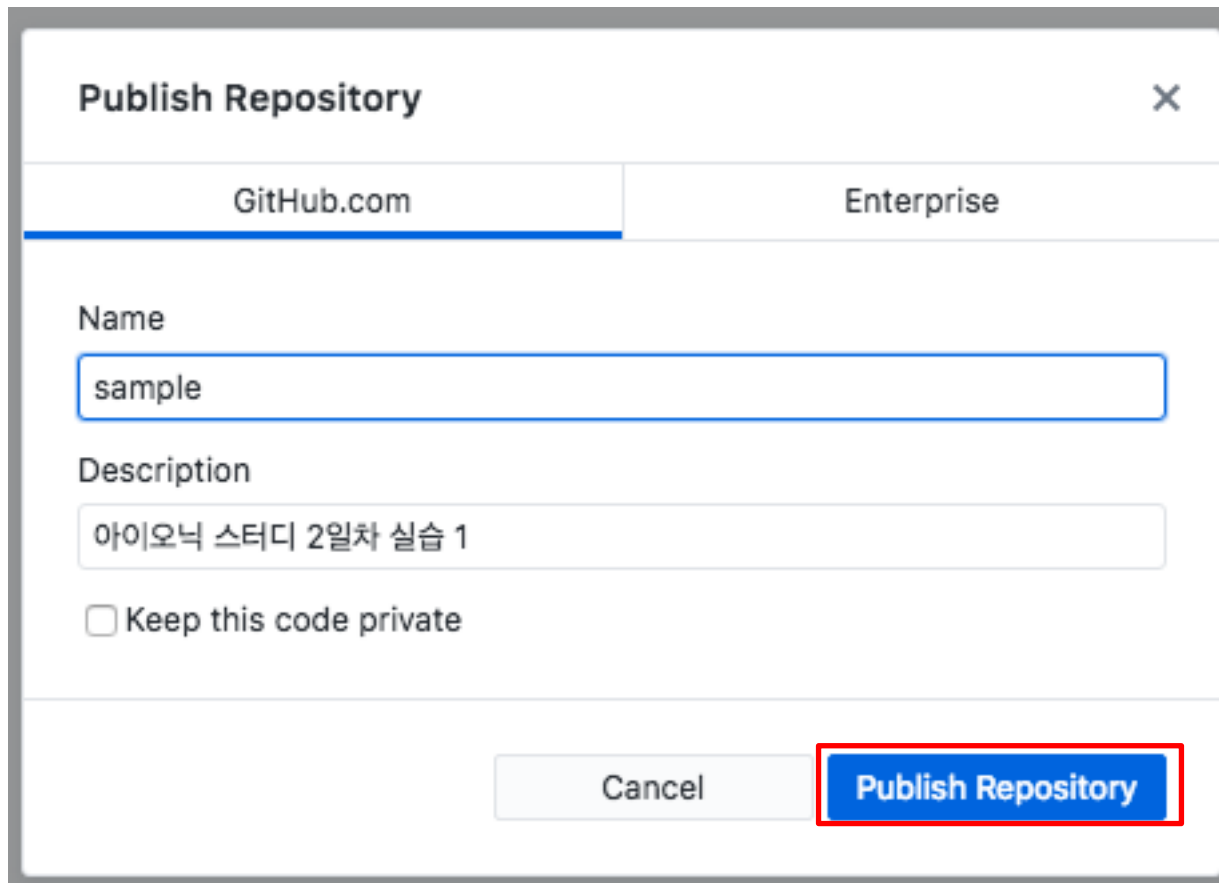
Single Page Application 맛보기

- 상단의 Publish repository 클릭



Single Page Application 맛보기

- [Publish Repository] 클릭



The screenshot shows a 'Publish Repository' dialog box with a close button (X) in the top right corner. It features two tabs: 'GitHub.com' (selected) and 'Enterprise'. Below the tabs, there are two text input fields: 'Name' with the value 'sample' and 'Description' with the value '아이오닉 스터디 2일차 실습 1'. A checkbox labeled 'Keep this code private' is unchecked. At the bottom, there are two buttons: 'Cancel' and 'Publish Repository', with the latter highlighted by a red rectangular border.

Publish Repository [X]

GitHub.com Enterprise

Name
sample

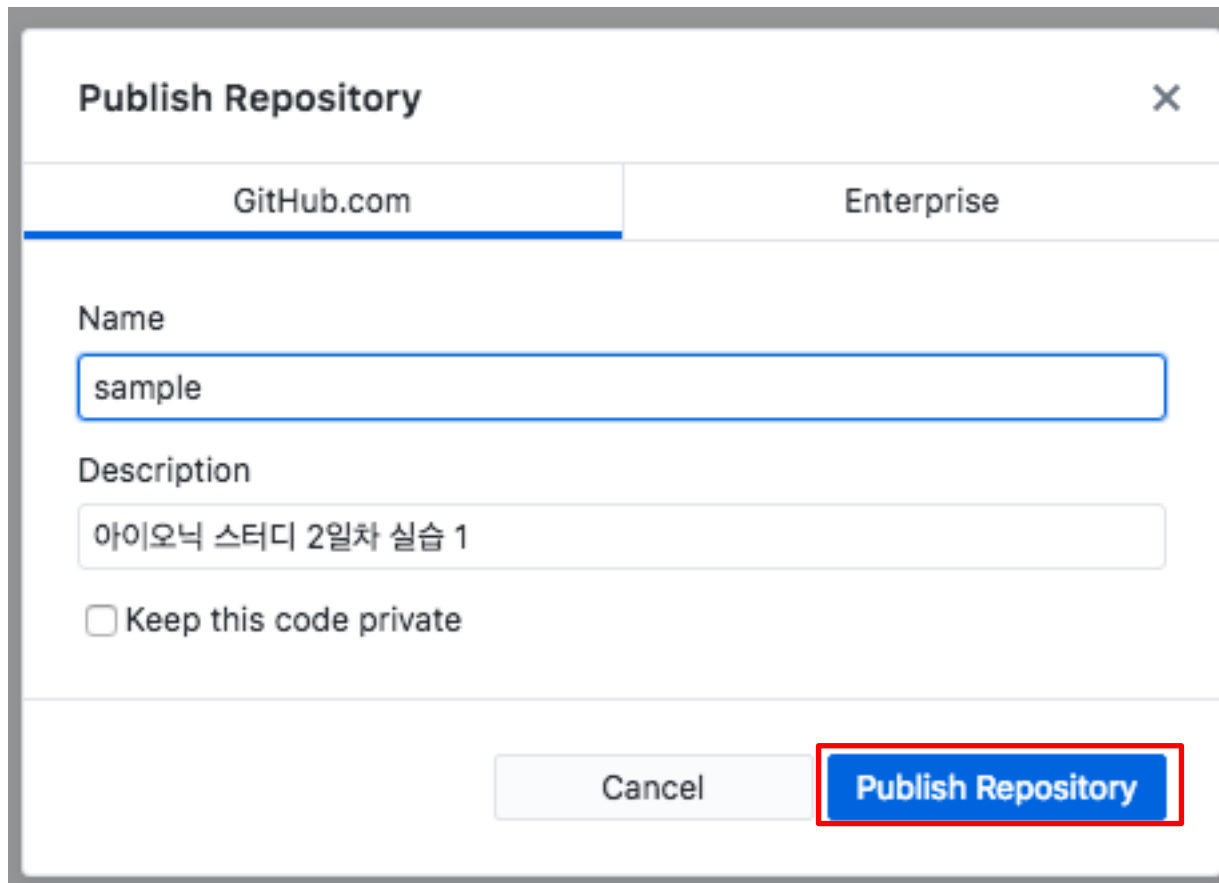
Description
아이오닉 스터디 2일차 실습 1

☐ Keep this code private

Cancel **Publish Repository**

Single Page Application 맛보기

- [Publish Repository] 클릭



The screenshot shows a 'Publish Repository' dialog box with a close button (X) in the top right corner. It features two tabs: 'GitHub.com' (selected) and 'Enterprise'. Below the tabs, there are two text input fields: 'Name' with the value 'sample' and 'Description' with the value '아이오닉 스터디 2일차 실습 1'. A checkbox labeled 'Keep this code private' is unchecked. At the bottom, there are two buttons: 'Cancel' and 'Publish Repository', with the latter highlighted by a red rectangular border.

Publish Repository [X]

GitHub.com Enterprise

Name
sample

Description
아이오닉 스터디 2일차 실습 1

☐ Keep this code private

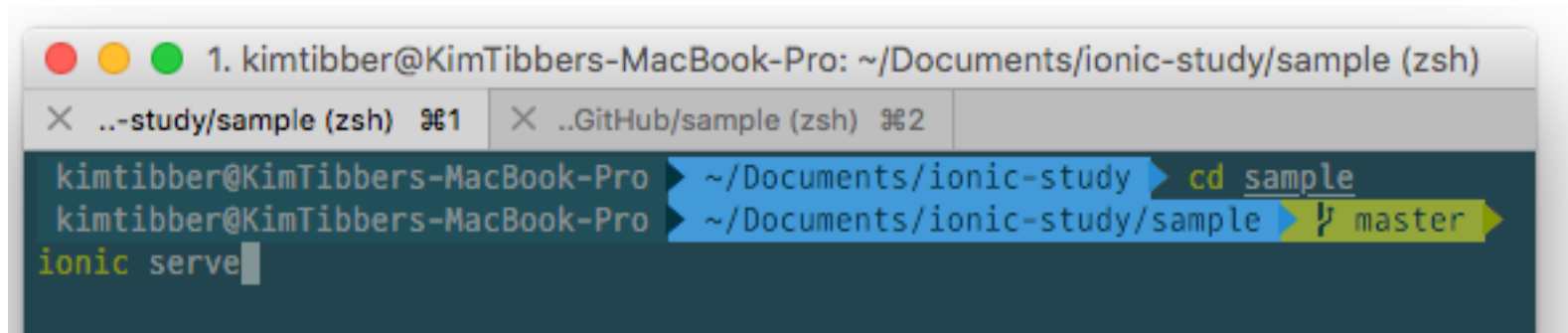
Cancel **Publish Repository**

Single Page Application 맛보기

- sample 프로젝트 브라우저로 미리보기.
 - 웹 브라우저 (구글 크롬등)로 작업중인 프로젝트를 미리볼 수 있음.
- sample 폴더 들어가기.
 - `cd sample` (Win/Mac 상관없음)

Single Page Application 맛보기

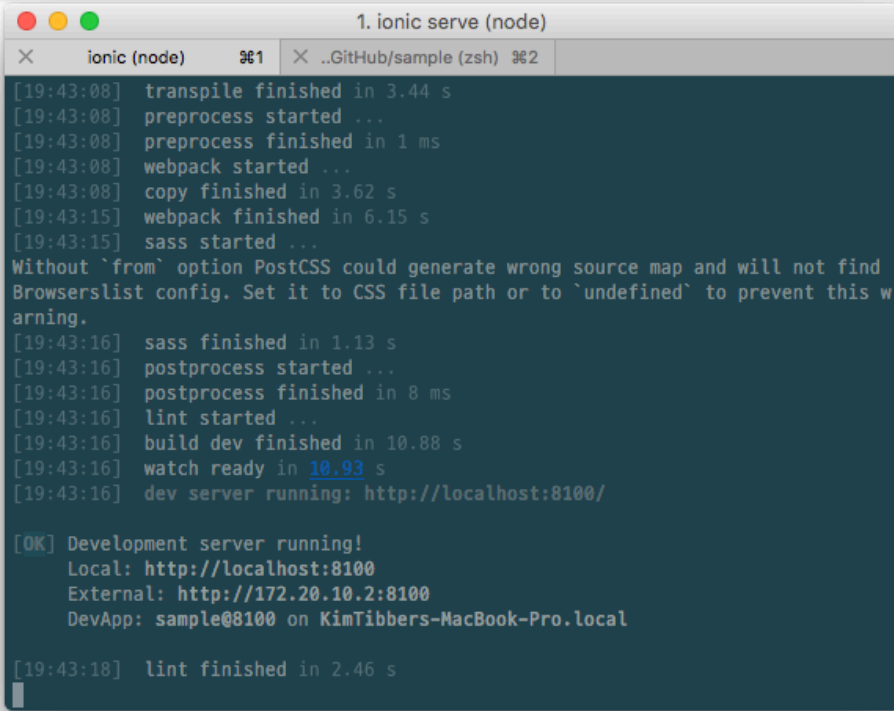
- ionic serve
 - Cordova 플러그인을 사용하여 스마트폰의 기능을 사용하지 않는 경우 웹 브라우저로 실행하는 옵션



```
1. kimtibber@KimTibbers-MacBook-Pro: ~/Documents/ionic-study/sample (zsh)
X ..-study/sample (zsh) 961 X ..GitHub/sample (zsh) 962
kimtibber@KimTibbers-MacBook-Pro ~/Documents/ionic-study > cd sample
kimtibber@KimTibbers-MacBook-Pro ~/Documents/ionic-study/sample > ? master
ionic serve
```

Single Page Application 맛보기

- 자동으로 배포과정 진행 후, 브라우저가 열리고 새로운 탭에 앱의 첫 화면이 나타남.



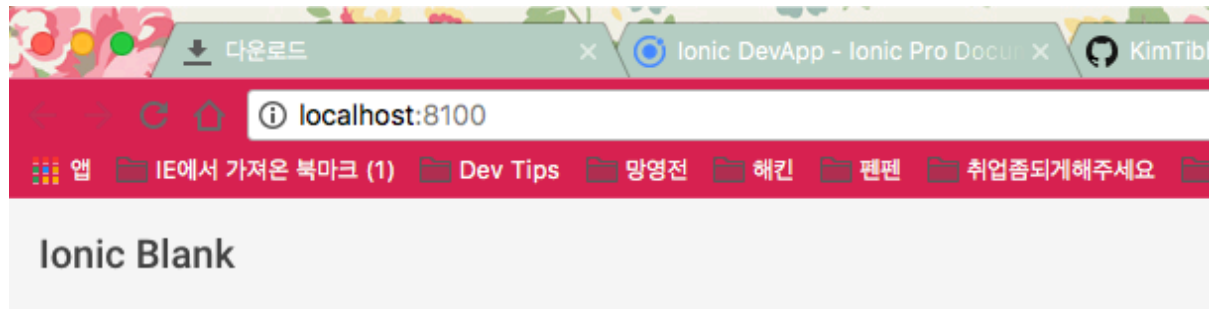
```
1. ionic serve (node)
ionic (node)  1  ..GitHub/sample (zsh)  2
[19:43:08] transpile finished in 3.44 s
[19:43:08] preprocess started ...
[19:43:08] preprocess finished in 1 ms
[19:43:08] webpack started ...
[19:43:08] copy finished in 3.62 s
[19:43:15] webpack finished in 6.15 s
[19:43:15] sass started ...
Without `from` option PostCSS could generate wrong source map and will not find
Browserslist config. Set it to CSS file path or to `undefined` to prevent this w
arning.
[19:43:16] sass finished in 1.13 s
[19:43:16] postprocess started ...
[19:43:16] postprocess finished in 8 ms
[19:43:16] lint started ...
[19:43:16] build dev finished in 10.88 s
[19:43:16] watch ready in 10.93 s
[19:43:16] dev server running: http://localhost:8100/

[OK] Development server running!
Local: http://localhost:8100
External: http://172.20.10.2:8100
DevApp: sample@8100 on KimTibbers-MacBook-Pro.local

[19:43:18] lint finished in 2.46 s
```

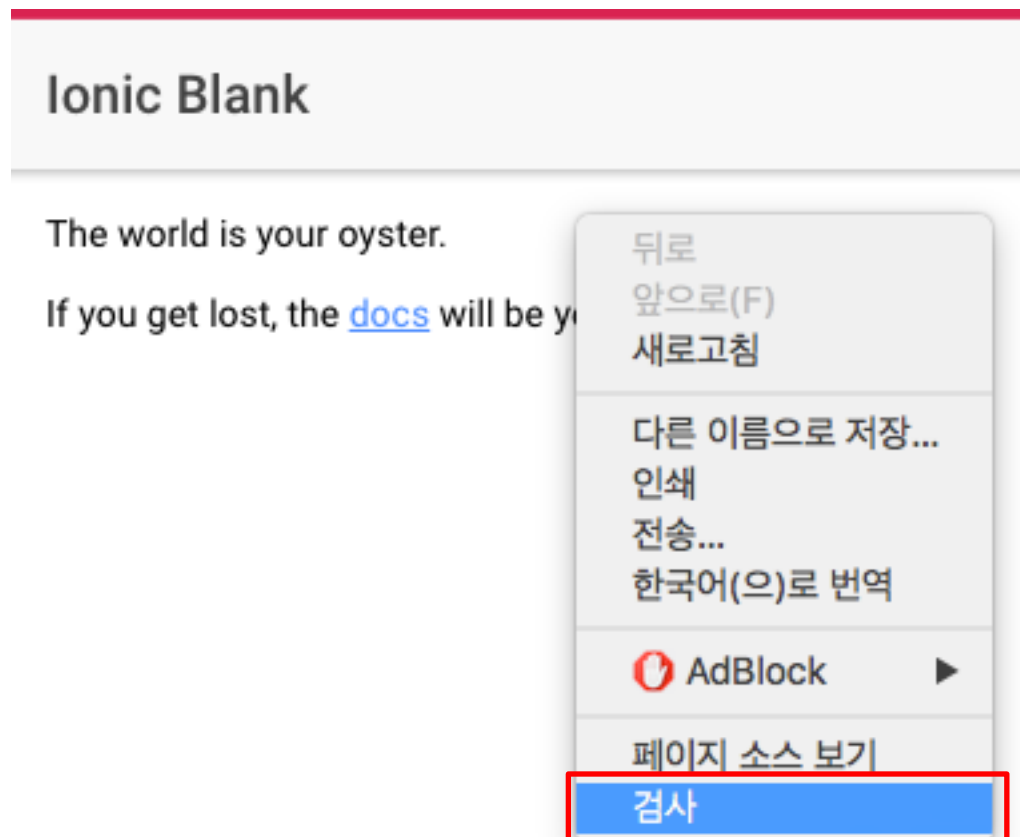
Single Page Application 맛보기

- 자동으로 배포과정 진행 후, 브라우저가 열리고 새로운 탭에 앱의 첫 화면이 나타남.



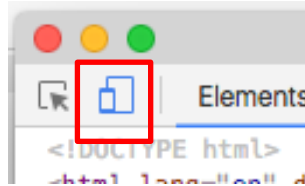
Single Page Application 맛보기

- 브라우저 화면에서 마우스 우클릭 후, [검사] 클릭

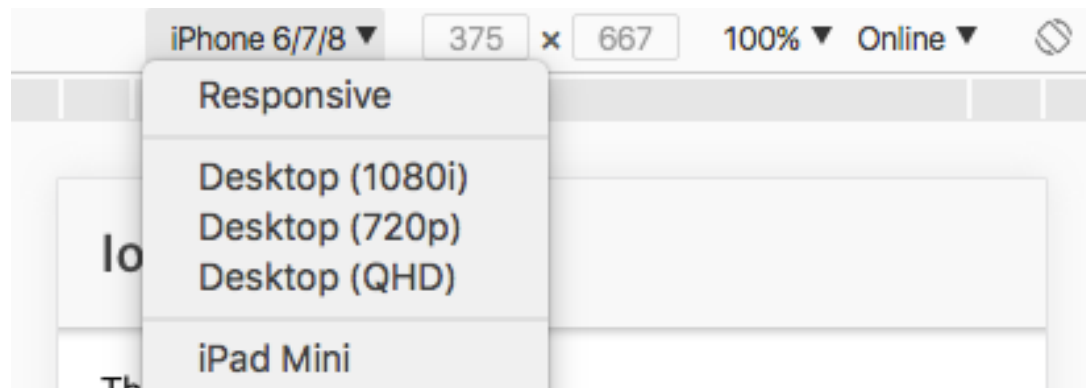


Single Page Application 맛보기

- 이 버튼을 클릭하면 디바이스 툴바가 나타나요.



- 브라우저에서 알맞은 스마트폰 기종을 선택하면 화면 크기가 자동으로 변경됨.

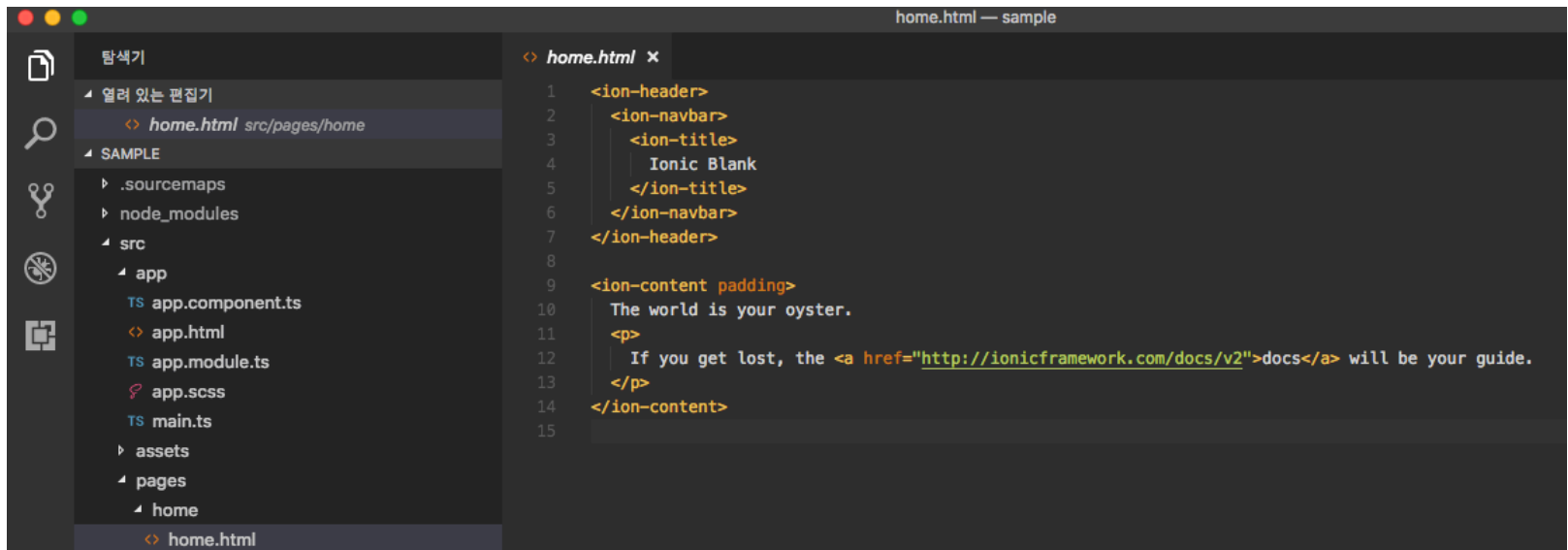


Single Page Application 맛보기

- 무작정 내용 바꿔보기.
 - Visual Studio Code로 해당 프로젝트 열기
 - `src/app/pages/home.html` 편집하기

Single Page Application 맛보기

- home.html



```
home.html — sample

<> home.html x
1  <ion-header>
2    <ion-navbar>
3      <ion-title>
4        Ionic Blank
5      </ion-title>
6    </ion-navbar>
7  </ion-header>
8
9  <ion-content padding>
10   The world is your oyster.
11   <p>
12     If you get lost, the <a href="http://ionicframework.com/docs/v2">docs</a> will be your guide.
13   </p>
14 </ion-content>
15
```

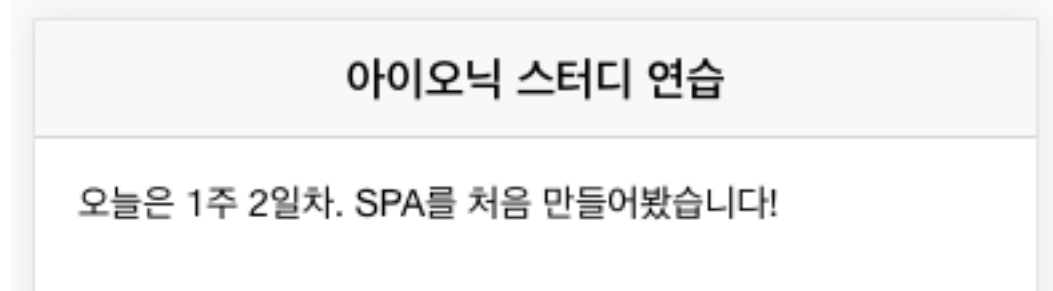
Single Page Application 맛보기

- 뭐가 뭔진 몰라도...
 - `<ion-title>` 이랑 `<ion-content>` 안에 무언가가 있다.
 - 이 태그 안의 내용을 바꿔보자.

```
<> home.html ●  
1   <ion-header>  
2     <ion-navbar>  
3       <ion-title>  
4         아이오닉 스터디 연습  
5       </ion-title>  
6     </ion-navbar>  
7   </ion-header>  
8  
9   <ion-content padding>  
10    오늘은 1주 2일차. SPA를 처음 만들어봤습니다!  
11  </ion-content>  
12
```

Single Page Application 맛보기

- Ctrl+S (Mac: Command+S)를 눌러 저장
 - 자동으로 브라우저에 표시되는 내용이 변경됨.

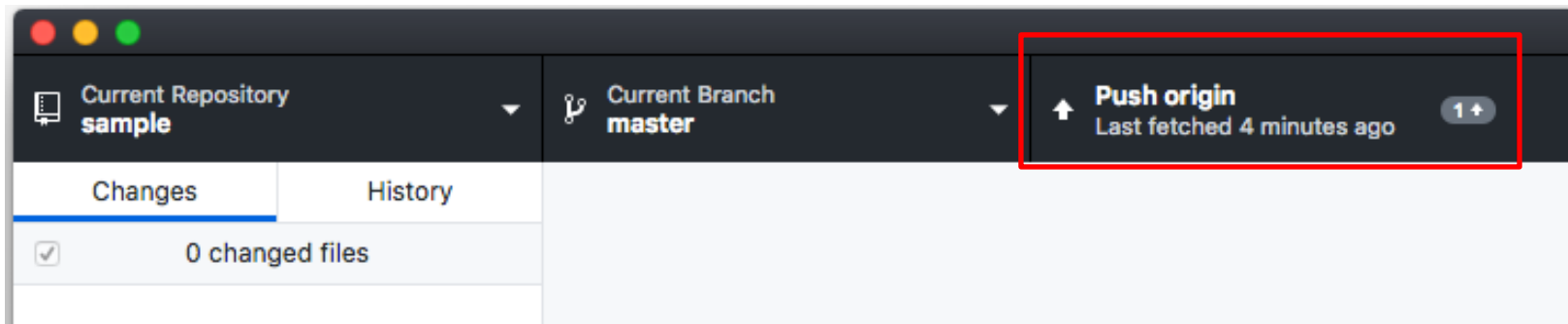


Single Page Application 맛보기

- ionic serve 종료
 - ionic serve를 실행한 명령 프롬프트 화면에서 Ctrl+C 입력.
 - 바로 종료됨.

Single Page Application 맛보기

- 원격 저장소로 발행하기
 - 우선 커밋을 생성해야 함.
 - Github Desktop에서 커밋 실행.
 - 이후 상단에서 [Push origin] 클릭



오늘은 여기까지~

See you next week!