

GitHub Classroom

Your course assignments on GitHub



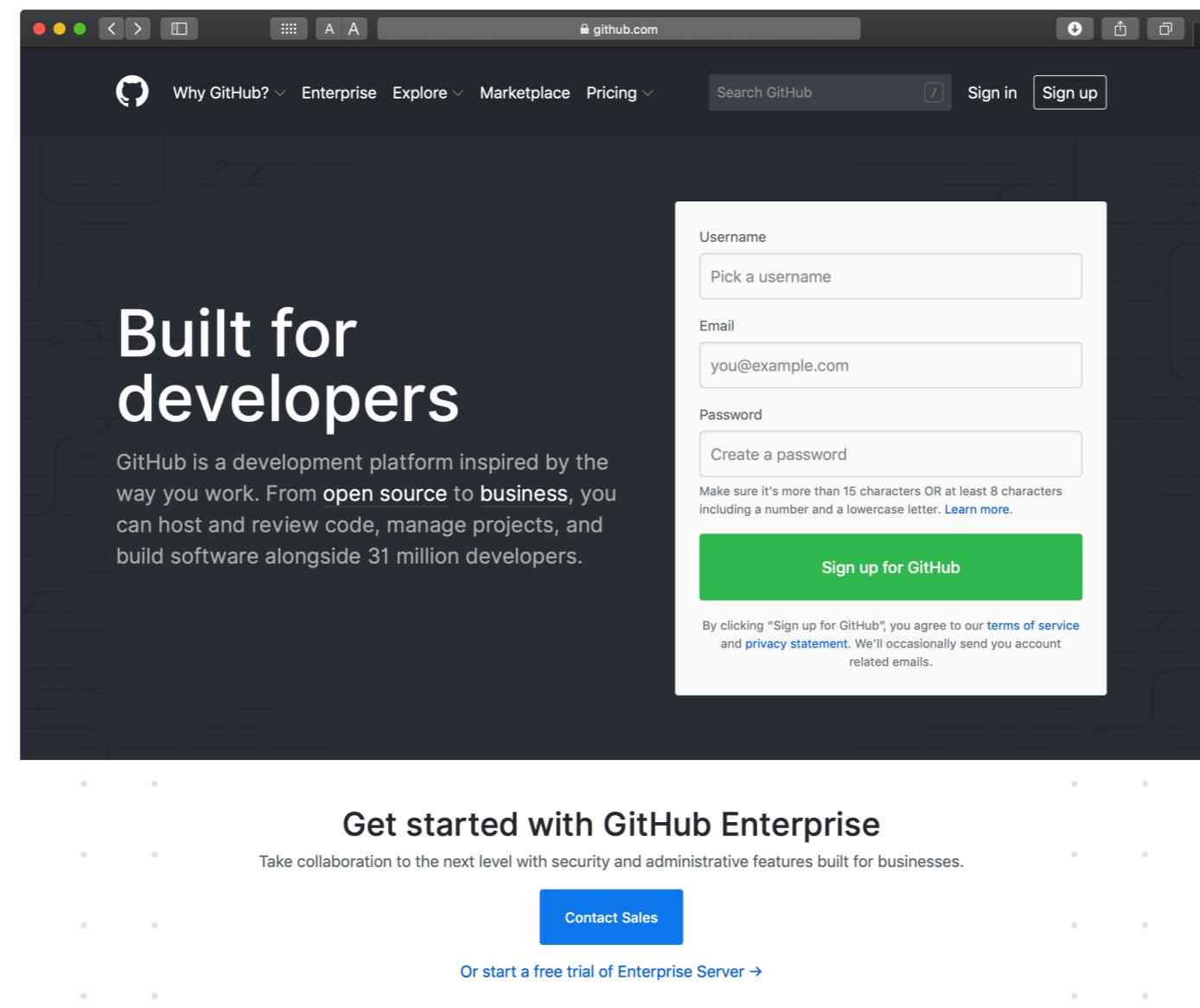
Git과 Github를 이용하여 과제 제출하기

1. Github 가입하고 내 PC에 git 깔기
2. 과제 생성하기
3. 과제 제출하기: 공용 PC에서
4. 과제 제출하기: 자신의 PC에서

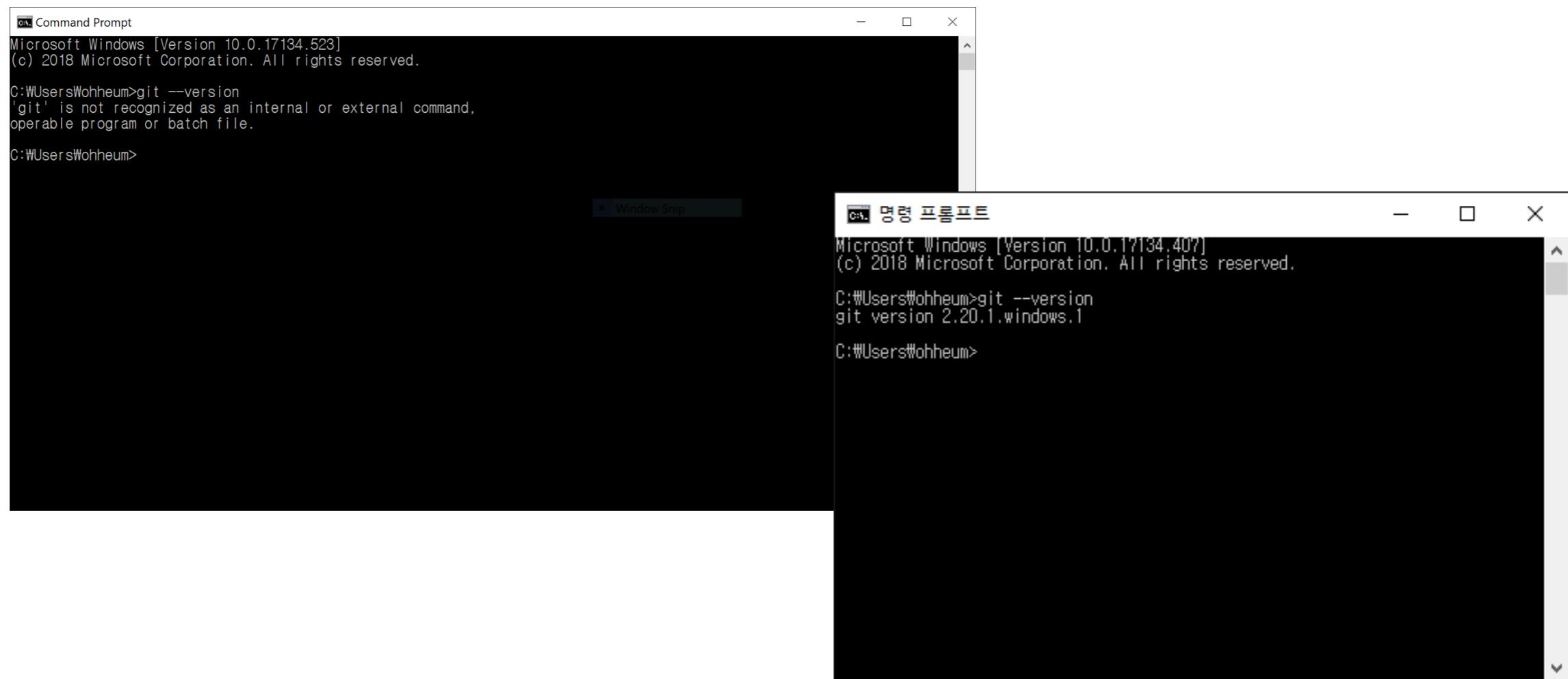
Github 가입하고 git 설치하기

- Github 주소는 <https://github.com> 이다. username과 email주소를 기입하고 Sign up을 한다.

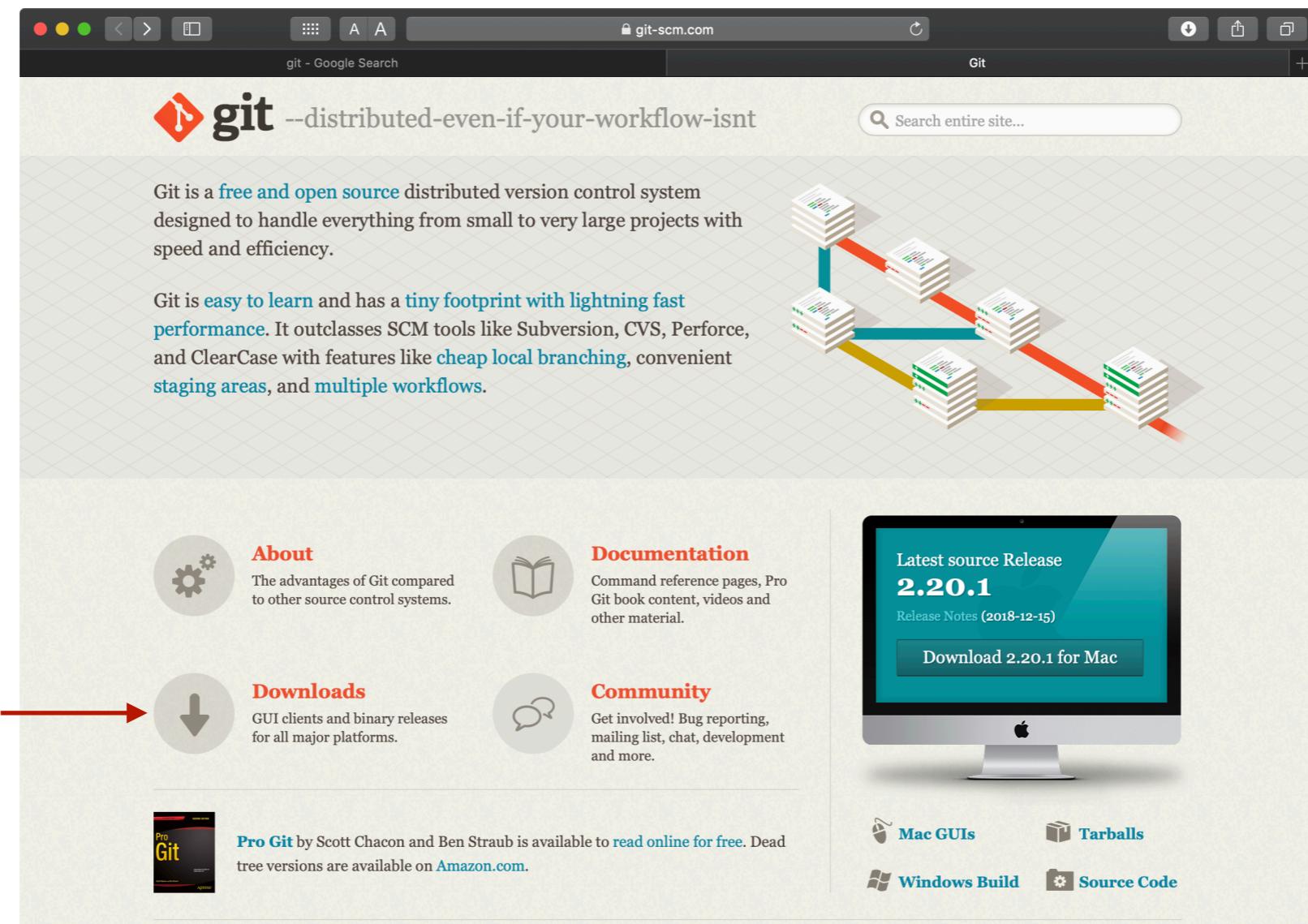
- 기입한 email 주소로 확인(confirm) 메일이 발송될 수 있다. 확인 메일을 열어서 confirm해주면 계정이 활성화된다.
- 자신의 GitHub 저장소를 앞으로도 계속 유지할 것을 염두에 두고 실제로 앞으로 계속 사용할 email 주소를 사용하라.



- Git은 가장 인기있는 version control system이다.
- Git 매뉴얼
- 명령어 창을 열고 git --version 명령을 입력한다. 왼쪽 그림과 같이 나오면 git이 설치되지 않은 상태이고, 오른쪽 그림과 같이 나오면 이미 설치되어 있는 상태이다.

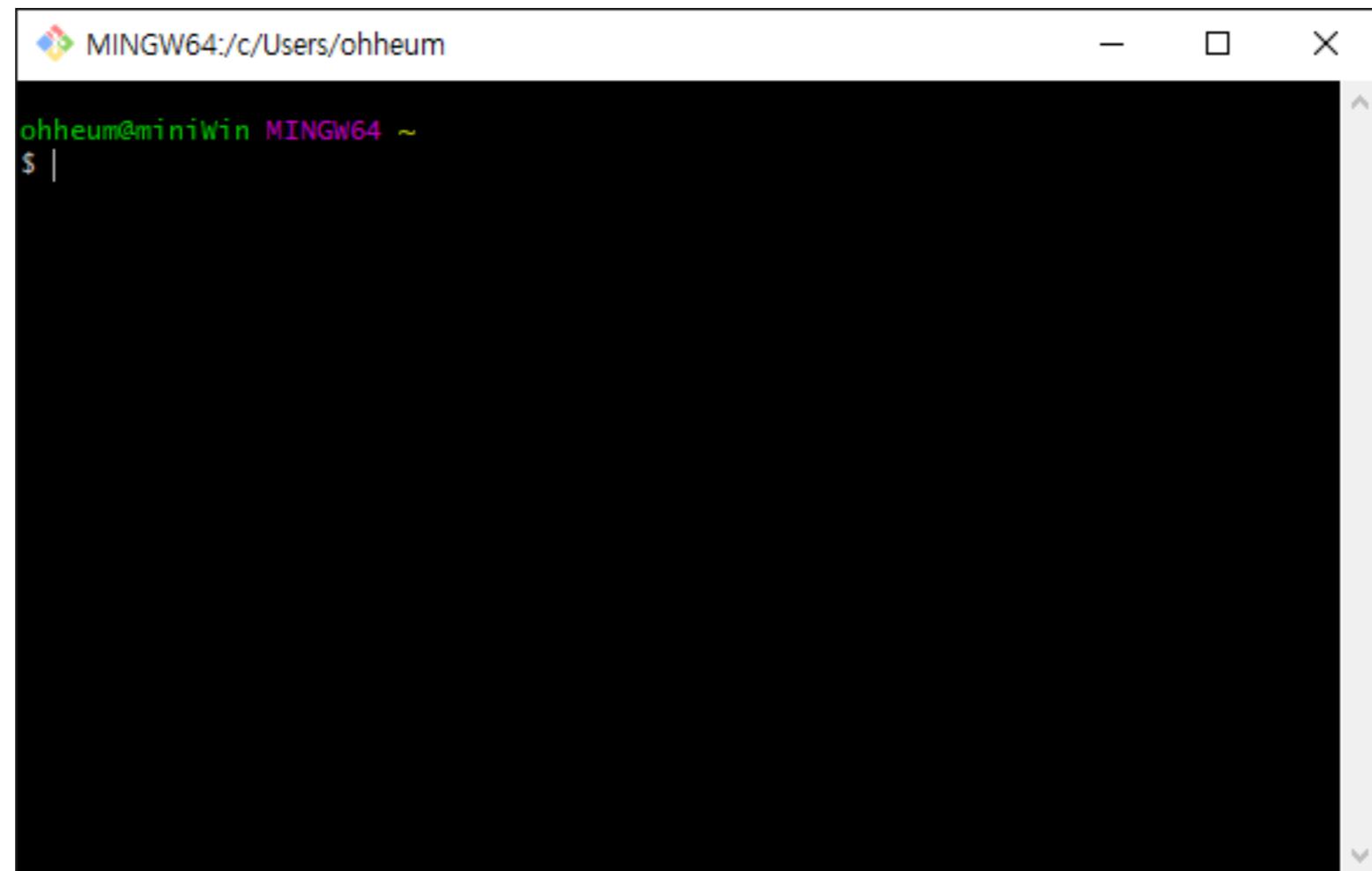


- ❶ git의 홈페이지 주소는 <https://git-scm.com> 이다. Windows용 git을 다운받아 설치한다. 특별히 주의할 사항은 없고 기본 설정을 따르면 된다.

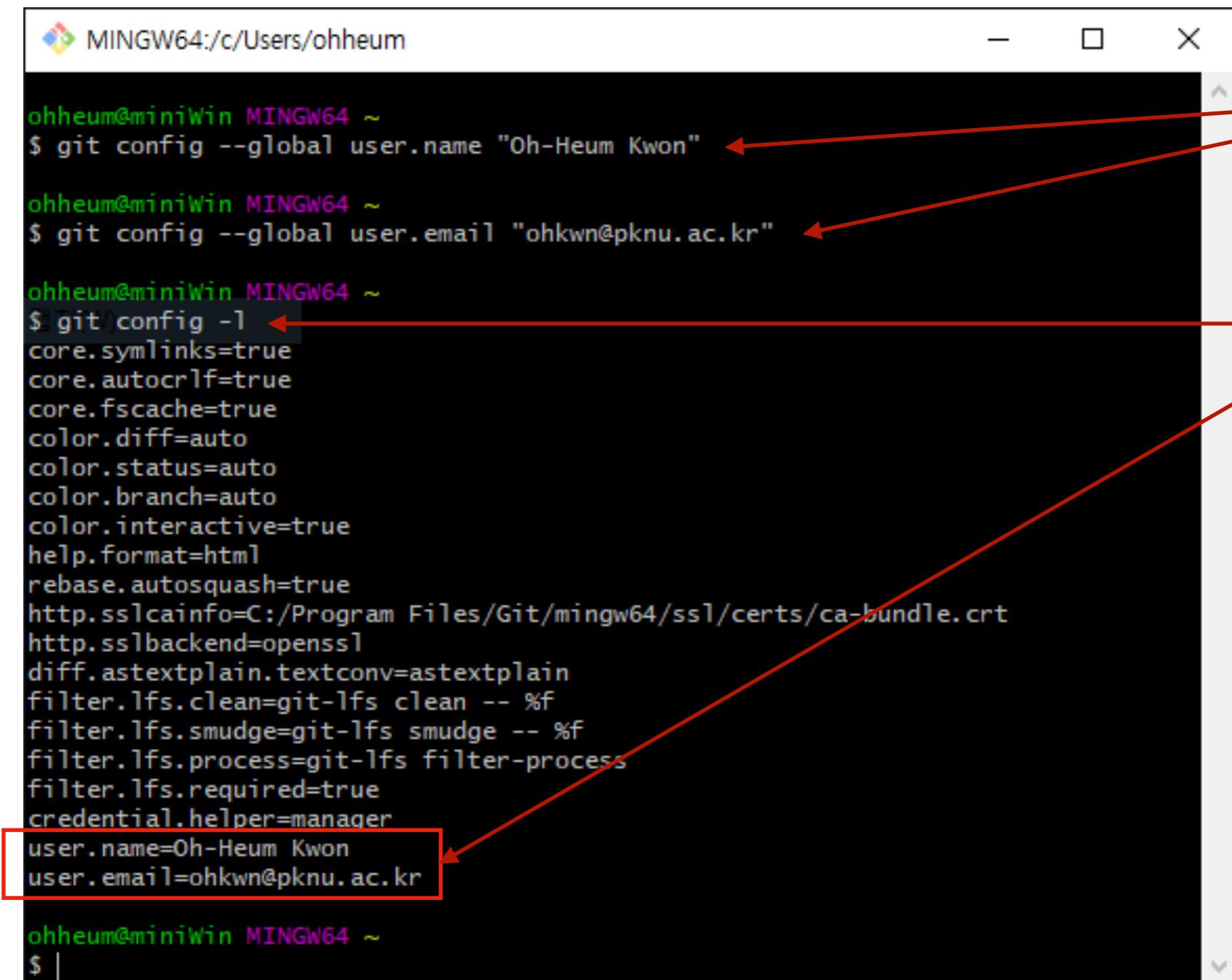


검색창에서 Git Bash를 찾아 실행하면 다음 그림과 같은 콘솔이 실행된다.

- 이 콘솔은 일종의 가상 linux 환경이라고 생각하면 된다. 즉 Windows에서 linux 명령을 실행해 준다.
- **Git Bash는 앞으로도 자주 사용할 것이므로 작업표시줄에 고정해두자.**



내 PC에 git 설치하기



MINGW64:/c/Users/ohheum

```
ohheum@miniWin MINGW64 ~
$ git config --global user.name "Oh-Heum Kwon"
ohheum@miniWin MINGW64 ~
$ git config --global user.email "ohkwn@pknu.ac.kr"
ohheum@miniWin MINGW64 ~
$ git config -l
core.symlinks=true
core.autocrlf=true
core.fscache=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
rebase.autosquash=true
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
http.sslbackend=openssl
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
credential.helper=manager
user.name=Oh-Heum Kwon
user.email=ohkwn@pknu.ac.kr

ohheum@miniWin MINGW64 ~
$ |
```

이렇게 자신의 이름과 email주소를 설정한다.

git config -l 명령으로 설정값들을 볼 수 있다.

```

MINGW64:/c/Users/ohheum/Desktop/Spring2019/Algorithm2019
ohheum@miniWin MINGW64 ~
$ pwd
/c/Users/ohheum
ohheum@miniWin MINGW64 ~
$ cd Desktop/
ohheum@miniWin MINGW64 ~/Desktop
$ ls
'0113 강의 계획서 탐재 '/      HS/
Algorithm2019/                  LtAppRsnRs1tSheet.pdf
desktop.ini                      pa-01-ohheum/
ohheum@miniWin MINGW64 ~/Desktop
$ mkdir Spring2019
ohheum@miniWin MINGW64 ~/Desktop
$ cd Spring2019/
ohheum@miniWin MINGW64 ~/Desktop/Spring2019
$ ls
ohheum@miniWin MINGW64 ~/Desktop/Spring2019
$ mkdir Algorithm2019
ohheum@miniWin MINGW64 ~/Desktop/Spring2019
$ cd Algorithm2019/
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019
$ ls
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019
$ pwd
/c/Users/ohheum/Desktop/Spring2019/Algorithm2019
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019
$ |

```

1. **pwd**는 현재 나의 위치를 보여주는 명령어이다. 나의 현 위치가 **C:\\\\Users\\\\ohheum**임을 보여준다.
2. **cd**는 다른 디렉토리로 이동하는 명령어이다. 여기서는 바탕화면 (Desktop)으로 이동하였다. **슬래쉬(/)**는 입력할 필요 없다. 만약 현재 위치에서 한 칸 위로 올라가고 싶다면 **cd ..** 이라고 하면 된다.
3. **ls**는 현재 디렉토리에 저장된 파일이나 하위 디렉토리의 목록을 보여주는 명령어이다.
4. **mkdir**은 현재 디렉토리 내에 새로운 디렉토리를 생성하는 명령이다. 여기에서는 바탕화면에 **Spring2019**라는 이름의 새로운 디렉토리를 생성하였다.
5. **cd**명령으로 방금 생성한 **Spring2019** 디렉토리로 이동하였다.
6. **ls**를 해보면 방금 새로 만든 디렉토리이므로 당연히 아무것도 없다.
7. 다시 **mkdir** 명령으로 **Spring2019** 디렉토리 내부에 **Algorithm2019**라는 이름의 디렉토리를 만들었다.
8. **cd**명령으로 방금 생성한 **Algorithm2019** 디렉토리로 이동하였다.
9. **ls**를 해보면 역시 새로 만든 디렉토리이므로 아무것도 없다.
10. **pwd**를 해보면 나의 현 위치를 보여준다.

명령어

의미

pwd

자신의 현재 위치(working directory or current directory)
를 알려줌 (print working directory)

ls

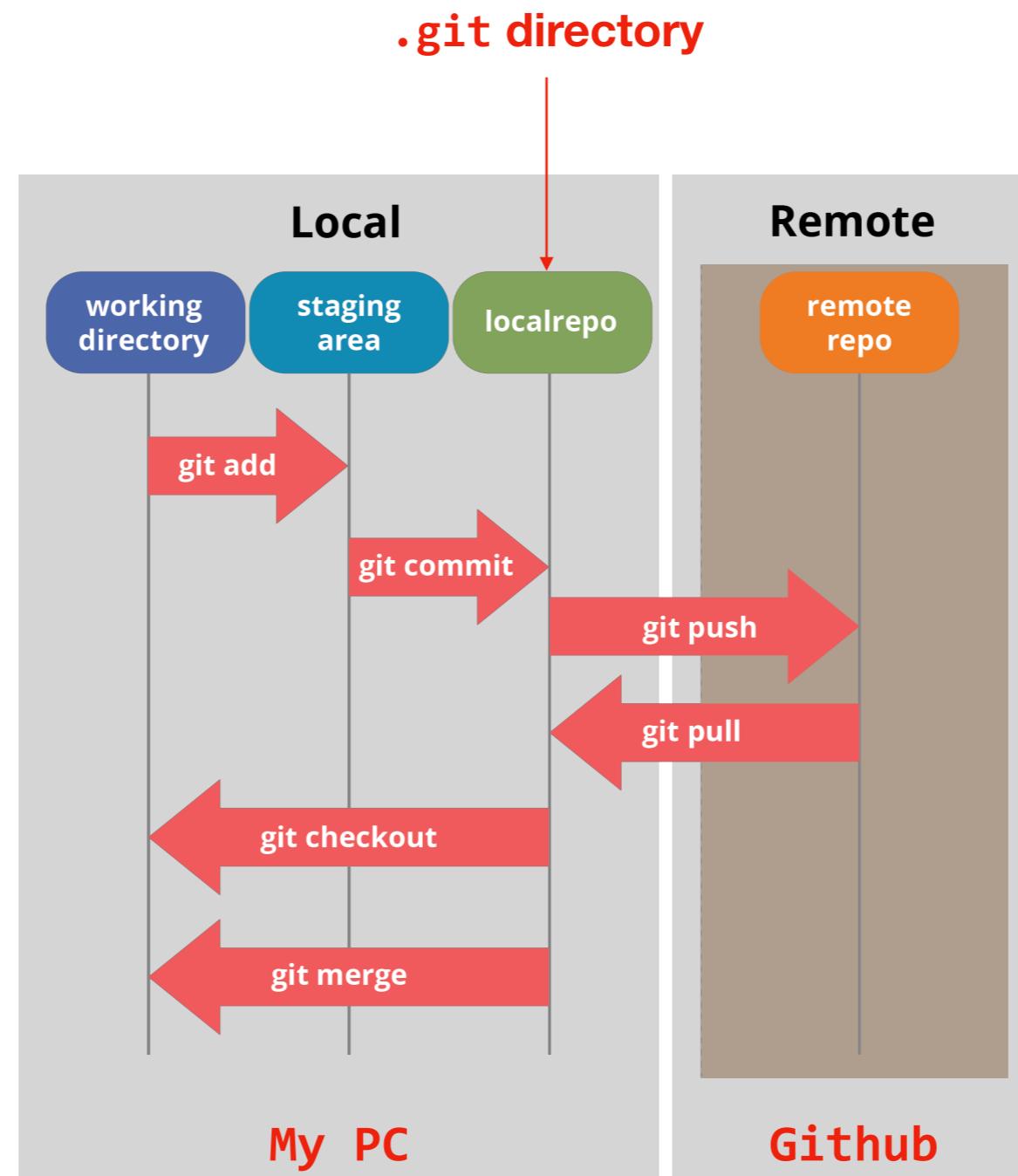
현재 디렉토리에 있는 파일과 서브 디렉토리의 목록을 보여줌
(ls -a는 숨긴 파일까지 보여줌)

cd

다른 디렉토리로 이동(change directory)
상위 디렉토리로 이동할 때는 cd ..

mkdir

디렉토리를 생성



Github에 과제 저장소(repository) 생성하기

④ Smart LMS에서 프로그래밍 과제 링크를 클릭한다. 과제는 “강의자료” 페이지에 게시된다.

한국어

부경대학교 Smart - LMS

권오흠 0 로그아웃

강의과목

2018-1학기

[대연]알고리즘(104) 월67 수2

▶ 강의계획서
▶ 온라인강의
▶ 공지사항
▶ 질의응답
▶ 강의자료
▶ 출석
▶ 과제
▶ 팀프로젝트
▶ 시험
▶ 토론
▶ 투표
▶ 설문
▶ 성적
▶ 열린게시판

강의자료

제목 2019년 알고리즘 강의 자료

공개구분 공개 (수강생 공개)

작성자 권오흠

공개일 2019.02.12 13:15

조회수 0

강의 소개

• 강의소개

제1장 알고리즘의 분석

학습목표

- 시간복잡도의 개념을 이해한다.
- 점근적 시간복잡도 분석 방법과 표현법을 학습한다.

강의자료

• 강의 슬라이드

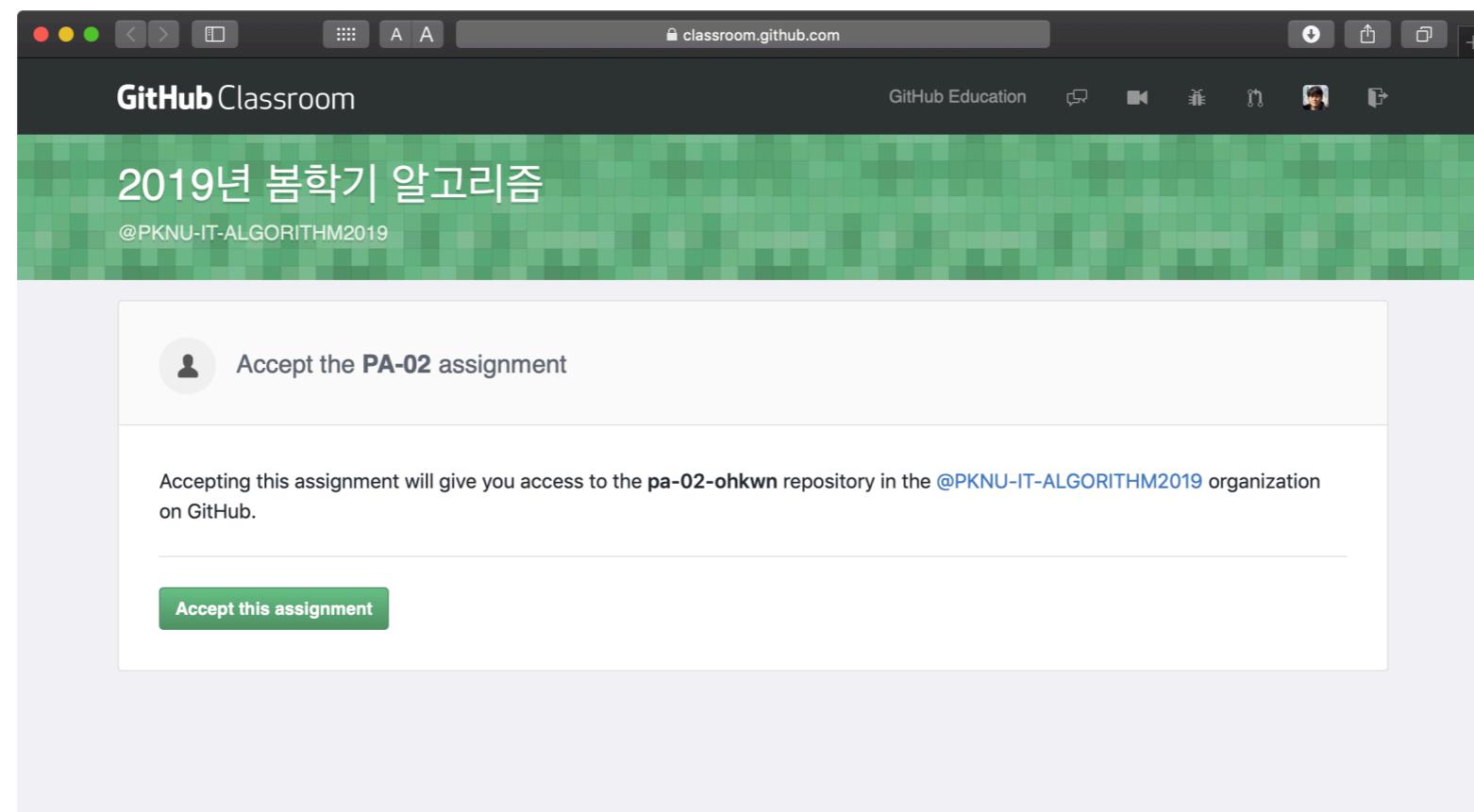
제1주 (3/4, 3/7)

- 동영상 강의1
- 동영상 강의2
- 동영상 강의3
- 동영상 강의4
- 프로그래밍 과제 01

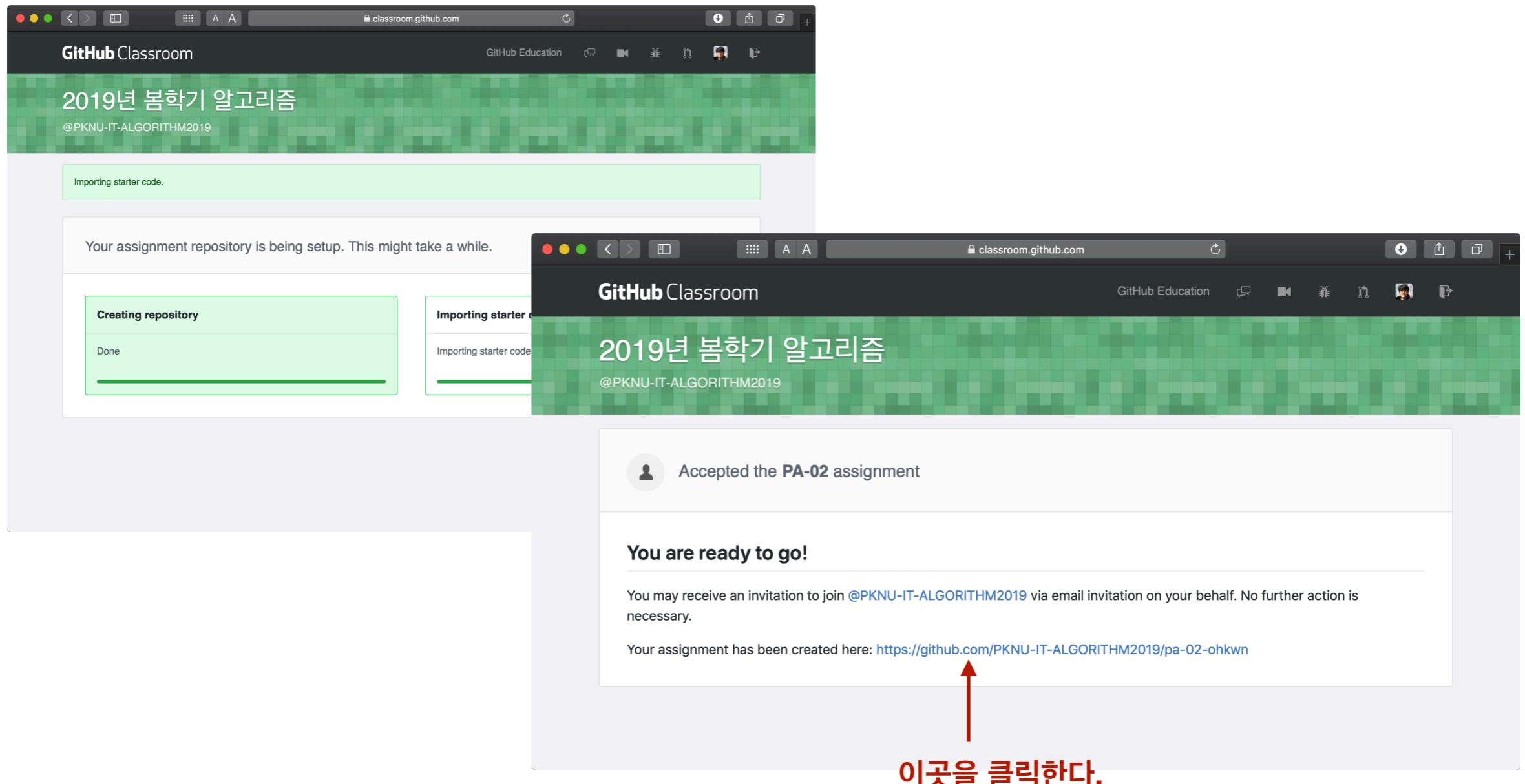
제2장 순환(recursion)

프로그래밍 과제를 클릭한다.

- ④ Github Classroom 사이트로 연결된다. Github에 로그인한 상태가 아니면 login 창이 먼저 뜬다.
- ⑤ 초록색 “Accept this assignment” 버튼을 클릭한다.



- ❶ 그냥 기다리면 첫번째 그림과 같은 상태를 거쳐서 두 번째 그림과 같은 상태가 된다.
- ❷ 그런 다음 생성된 과제 repository 링크를 클릭한다.



자신의 과제 repository가 생성되었다.

repository 이름에는 자신의 id가 포함될 것이다.

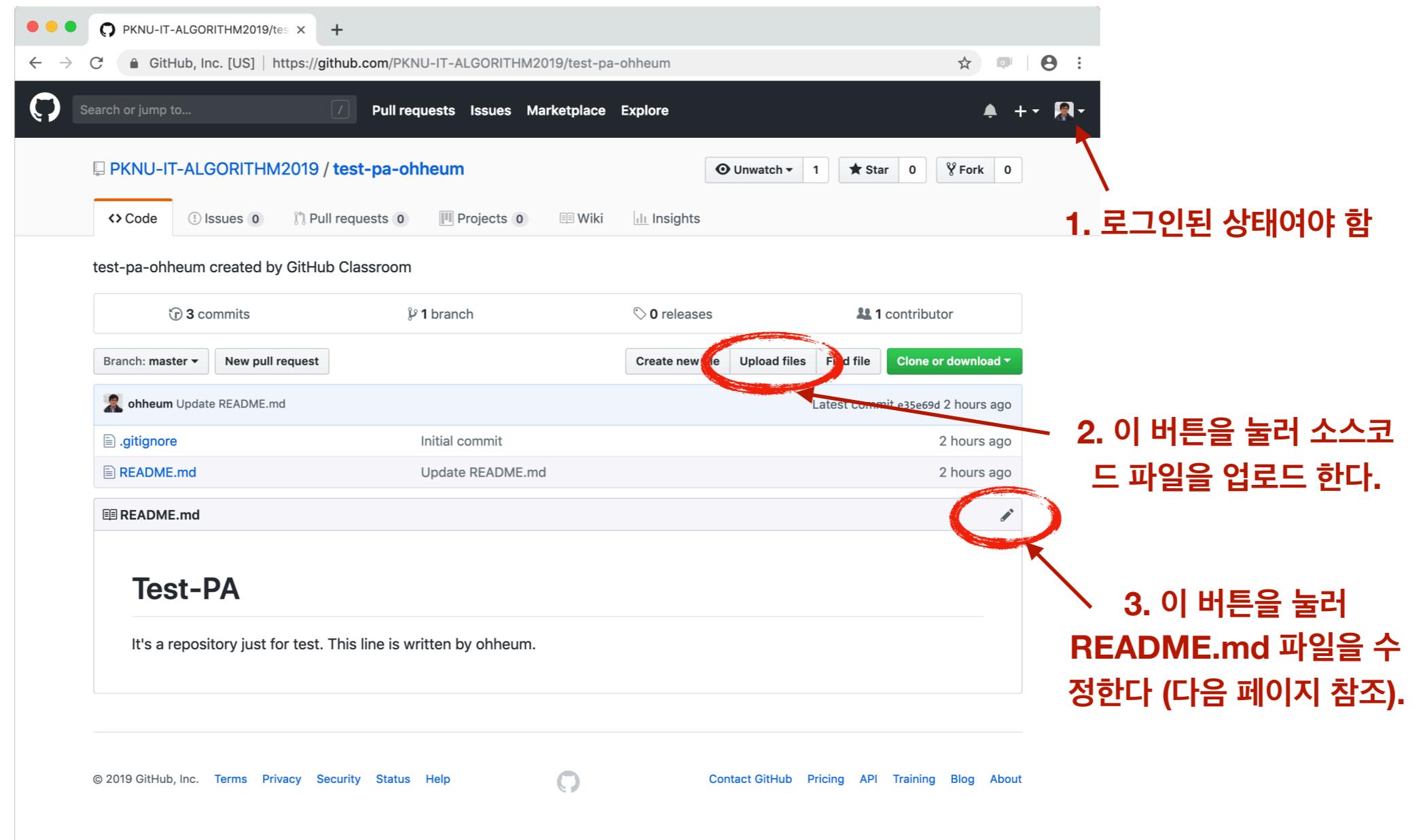
The screenshot shows a GitHub repository page for 'pa-02-ohkwn'. At the top, there's a navigation bar with links like 'Why GitHub?', 'Enterprise', 'Explore', 'Marketplace', 'Pricing', 'Search', 'Sign in', and 'Sign up'. Below the navigation bar, the repository name 'PKNU-IT-ALGORITHM2019 / pa-02-ohkwn' is displayed, followed by a series of zeroed-out metrics: 'Watch 0', 'Star 0', 'Fork 0'. A red arrow points from the text 'repository 이름에는 자신의 id가 포함될 것이다.' to the repository name. The main content area shows a 'Join GitHub today' banner with a 'Sign up' button. Below the banner, it says 'pa-02-ohkwn created by GitHub Classroom'. It lists '3 commits', '1 branch', '0 releases', and '1 contributor'. A red box highlights the 'Find file' and 'Clone or download' buttons. Another red arrow points from the text 'Github에 로그인한 상태라면 이 부분에 "Create new file", "Upload files" 등의 버튼이 있을 것이다.' to these buttons. The repository's commit history is shown, with the most recent commit being 'ohkwn Update README.md' made 7 hours ago. A red arrow points from the text '제출기한이나 주의사항 등의 사항들이 있을 것이다.' to the commit message. At the bottom of the page, there's footer information including copyright details and links to 'Contact GitHub', 'Pricing', 'API', 'Training', 'Blog', and 'About'.

실제 과제에는 이 부분에 과제의 내용을 기술한 pdf 파일이 있을 것이다.

제출기한이나 주의사항 등의 사항들이 있을 것이다.

공용 PC에서 과제 제출하기

- 실습실 PC 등의 공용 PC에서 과제를 제출할 때는 아래 그림에서 "Upload files" 버튼을 클릭하여 소스코드 파일을 업로드한다. 제일 간단한 방법이기는 하지만 부득이한 경우가 아니면 권장하지 않는다.



공용 PC에서 과제 제출하기

The screenshot shows a GitHub repository page for 'PKNU-IT-ALGORITHM2019/test-pa-ohheum'. The user is editing the 'README.md' file. The code editor contains the following text:

```
1 # 알고리즘 프로그래밍 과제 02|  
2  
3 이름: 홍길동  
4  
5 학번: 20001234  
6
```

A red arrow points from the text '홍길동' to the heading '4. README.md 파일에 자신의 이름과 학번 등을 기입한다.'

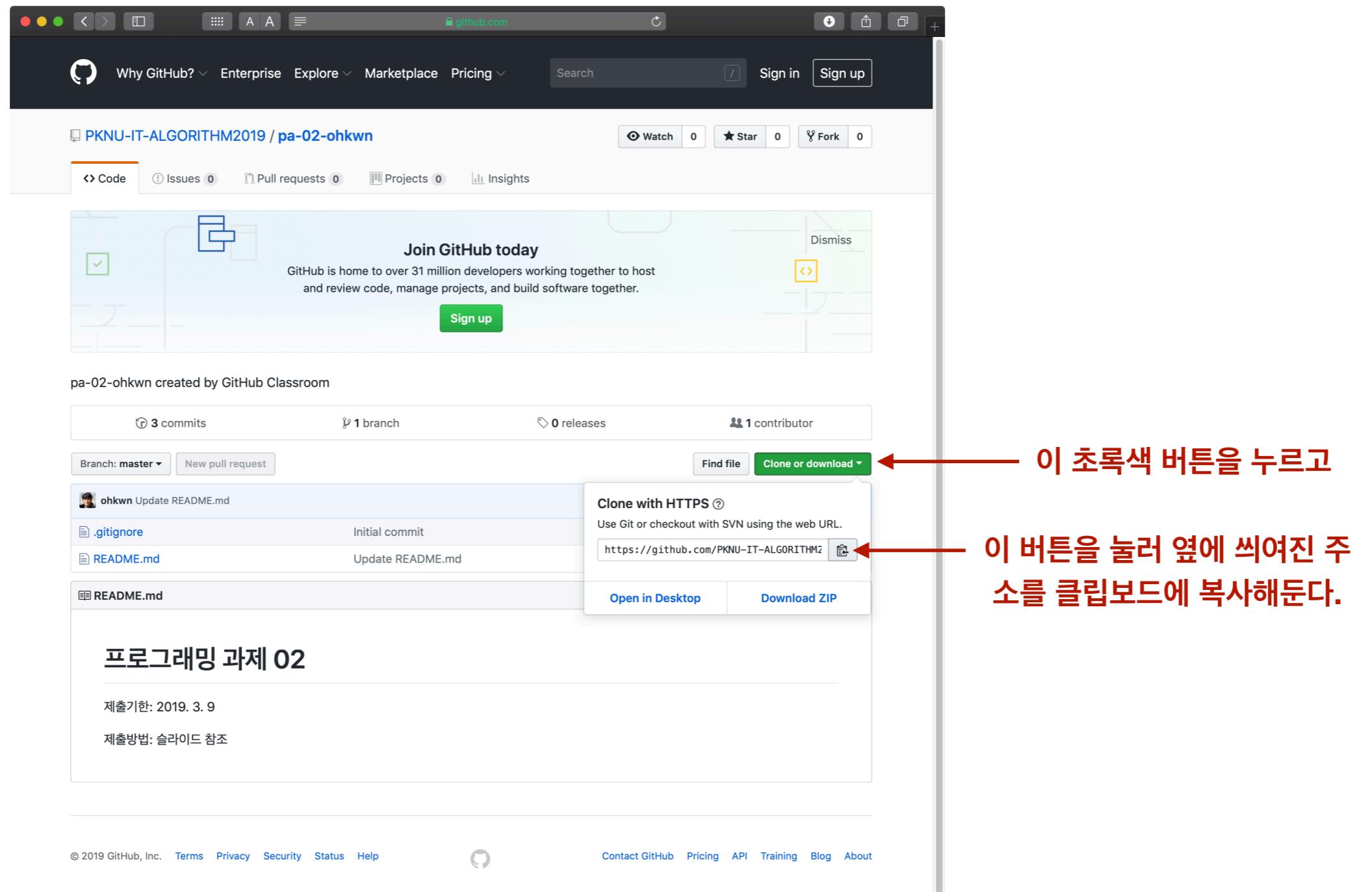
Below the editor is a 'Commit changes' dialog box. It includes fields for 'Update README.md' and 'Add an optional extended description...', and two radio button options: 'Commit directly to the master branch.' (selected) and 'Create a new branch for this commit and start a pull request.' A red arrow points from the 'Commit changes' button to the heading '5. 이 버튼을 눌러 저장한다.'

At the bottom of the page, there are copyright notices for GitHub, Inc. and links to Contact GitHub, Pricing, API, Training, Blog, and About.

자신의 PC에서 과제 수행하고 제출하기

과제의 local repository 생성하기

Repository 주소를 복사한다.



과제의 local repository 생성하기

The screenshot shows a Git Bash terminal window with the following command history:

```
MINGW64:/c/Users/ohheum/Desktop/Spring2019/Algorithm2019/pa-02-...
ohheum@miniWin MINGW64 ~
$ pwd
/c/Users/ohheum
ohheum@miniWin MINGW64 ~
$ cd Desktop/Spring2019/Algorithm2019/
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019
$ pwd
/c/Users/ohheum/Desktop/Spring2019/Algorithm2019
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019
$ git clone https://github.com/PKNU-IT-ALGORITHM2019/pa-02-ohkwn.git
Cloning into 'pa-02-ohkwn'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 10 (delta 0), reused 10 (delta 0), pack-reused 0
Unpacking objects: 100% (10/10), done.

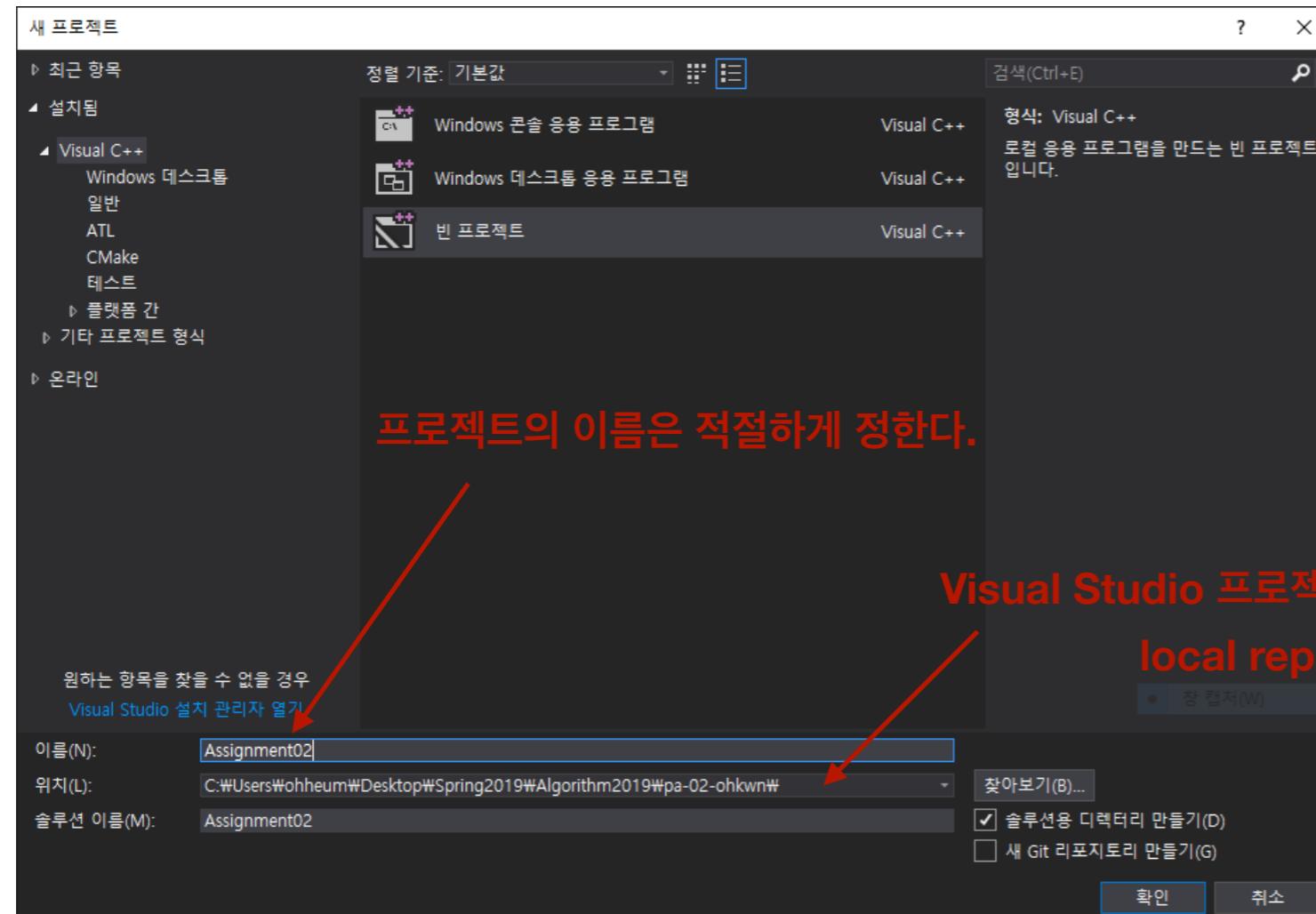
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019
$ ls
pa-02-ohkwn/
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019
$ cd pa-02-ohkwn/
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ ls
README.md
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ ls -a
./ ../ .git/ .gitignore README.md
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ |
```

Red arrows point from the numbered steps on the right to specific parts of the terminal output. A red circle highlights the directory name in the final command line.

1. Git Bash 콘솔을 실행한다.
2. cd 명령을 사용하여 과제를 수행할 디렉토리로 이동한다.
3. git clone 명령으로 Github repository를 복사한다. 명령의 형식은 다음과 같다.
\$ git clone “클립보드에 복사한 repository 주소”
4. Github repository의 내용이 복사되어진다.
5. Github repository의 이름과 동일한 이름의 디렉토리가 생성된다.
6. 생성된 디렉토리에 들어가 본다.
7. README.md 등 GitHub repository에 있는 파일들이 복사되었다. 실제의 과제에서는 과제의 내용을 기술하는 pdf 파일도 복사될 것이다.
8. ls -a 명령은 숨긴파일까지 보여주는 명령이다. .git 디렉토리와 .gitignore 등 숨긴 디렉토리와 파일도 보인다.

이 디렉토리가 과제를 수행할 디렉토리이다. 잊어버리지 말자.

Visual Studio 프로젝트 생성하기

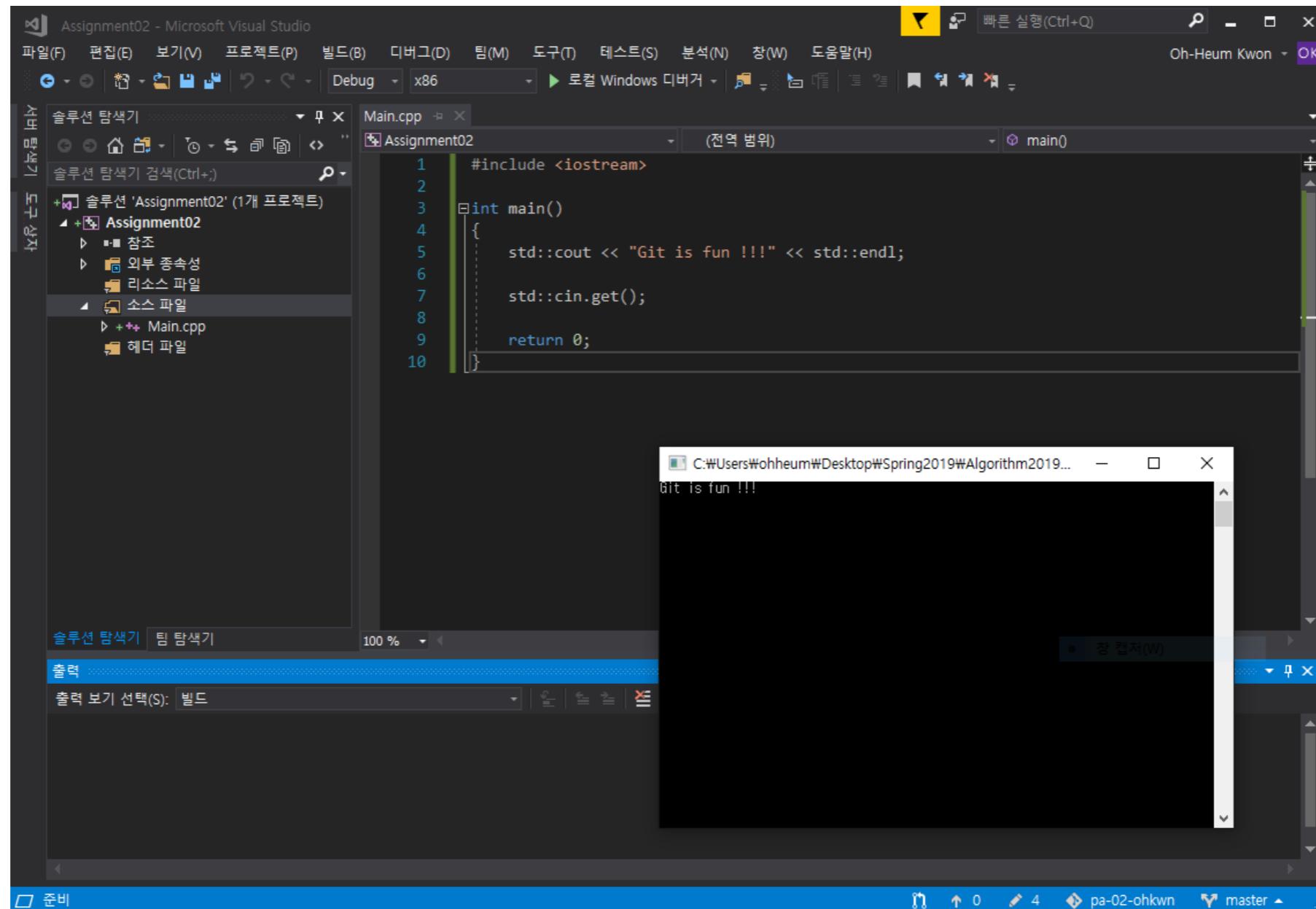


Visual Studio 프로젝트를 만들 때 위치를 반드시 복사된 local repository의 위치로 지정한다.

이미 다른 위치에 프로젝트가 생성/저장되어 있다면
그냥 이 위치로 프로젝트(솔루션) 디렉토리 전체를 복사해오면 된다.

Visual Studio 프로젝트 생성하기

❶ 열심히 과제를 한다 !!!



2. README.md 파일을 열고 이름과 학번을 기입한다.
그냥 텍스트 파일이므로 아무 텍스트 에디터로 열어도 된다.

```

MINGW64:/c/Users/ohheum/Desktop/Spring2019/Algorithm2019/pa-02-...
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ ls
Assignment02/ README.md

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ git add --all

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ git commit -m 'First Commit'
[master bd024a3] First Commit
 4 files changed, 186 insertions(+)
 create mode 100644 Assignment02/Assignment02.sln
 create mode 100644 Assignment02/Assignment02.Assignment02.vcxproj
 create mode 100644 Assignment02/Assignment02.Assignment02.vcxproj.filters
 create mode 100644 Assignment02/Assignment02/Main.cpp

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ git push
remote: Permission to PKNU-IT-ALGORITHM2019/pa-02-ohkwn.git denied to ohheum.
fatal: unable to access 'https://github.com/PKNU-IT-ALGORITHM2019/pa-02-ohkwn.git': The requested URL returned error: 403

```

1. Git Bash 콘솔을 실행하고 과제 repository를 찾아간다. 여기서 Assignment02가 Visual Studio 프로젝트이다. 아래의 작업을 프로젝트 디렉토리 내부에 들어가서 해서는 안된다.

3. git add --all 명령을 실행한다.

4. git commit -m 'Some Message' 명령을 실행한다.

자신이 알아볼 수 있는 아무 메시지나 상관없다.

5. git push 명령을 실행한다.

6. 뭔가 문제가 생겼다. 이 문제는 하나의 PC에서 두 개 이상의 github 계정에 접속할 때 생기는 문제이다.

만약 이 문제가 생겼다면 다음 다음 페이지에 설명한 방법으로 해결한다.

7. 오류가 생기지 않고 git push 명령이 이 그림처럼 진행되면 잘 된 것이다.

```

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ git push
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 2.50 KiB | 511.00 KiB/s, done.
Total 8 (delta 0), reused 0 (delta 0)
To https://github.com/PKNU-IT-ALGORITHM2019/pa-02-ohkwn.git
  43d35e1..bd024a3 master -> master

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-02-ohkwn (master)
$ |

```

The screenshot shows a GitHub repository page for 'PKNU-IT-ALGORITHM2019 / pa-02-ohkwn'. The page displays 4 commits, 1 branch, and 0 releases. A red arrow points from the text '앞 페이지에서 오류가 생기지 않고 git push 명령이 수행되었다면 github repository에 이렇게 Visual Studio 프로젝트가 저장된다.' to the commit list. Another red arrow points from the text '앞의 두 페이지의 과정은 반복해도 된다. 즉 과제를 제출한 후에 Visual Studio에서 코드를 수정하거나, 파일을 추가 혹은 삭제한 후 앞 페이지의 3개의 명령들' to the README.md file content.

Join GitHub today
GitHub is home to over 31 million developers working together to host and review code, manage projects, and build software together.
[Sign up](#)

pa-02-ohkwn created by GitHub Classroom

4 commits	1 branch	0 releases	1 contributor
Branch: master	New pull request	Find file	Clone or download
Oh-Heum Kwon First Commit	Assignment02	First Commit	23 minutes ago
	.gitignore	Initial commit	8 hours ago
	README.md	Update README.md	8 hours ago

프로그래밍 과제 02

제출기한: 2019. 3. 9
제출방법: 슬라이드 참조

README.md 파일을 수정했다면 여기에 이름과 학번 등이 표시될 것이다. 만약 깜빡했다면 18, 19페이지에 설명한 방법으로 수정해주어도 상관없다.

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

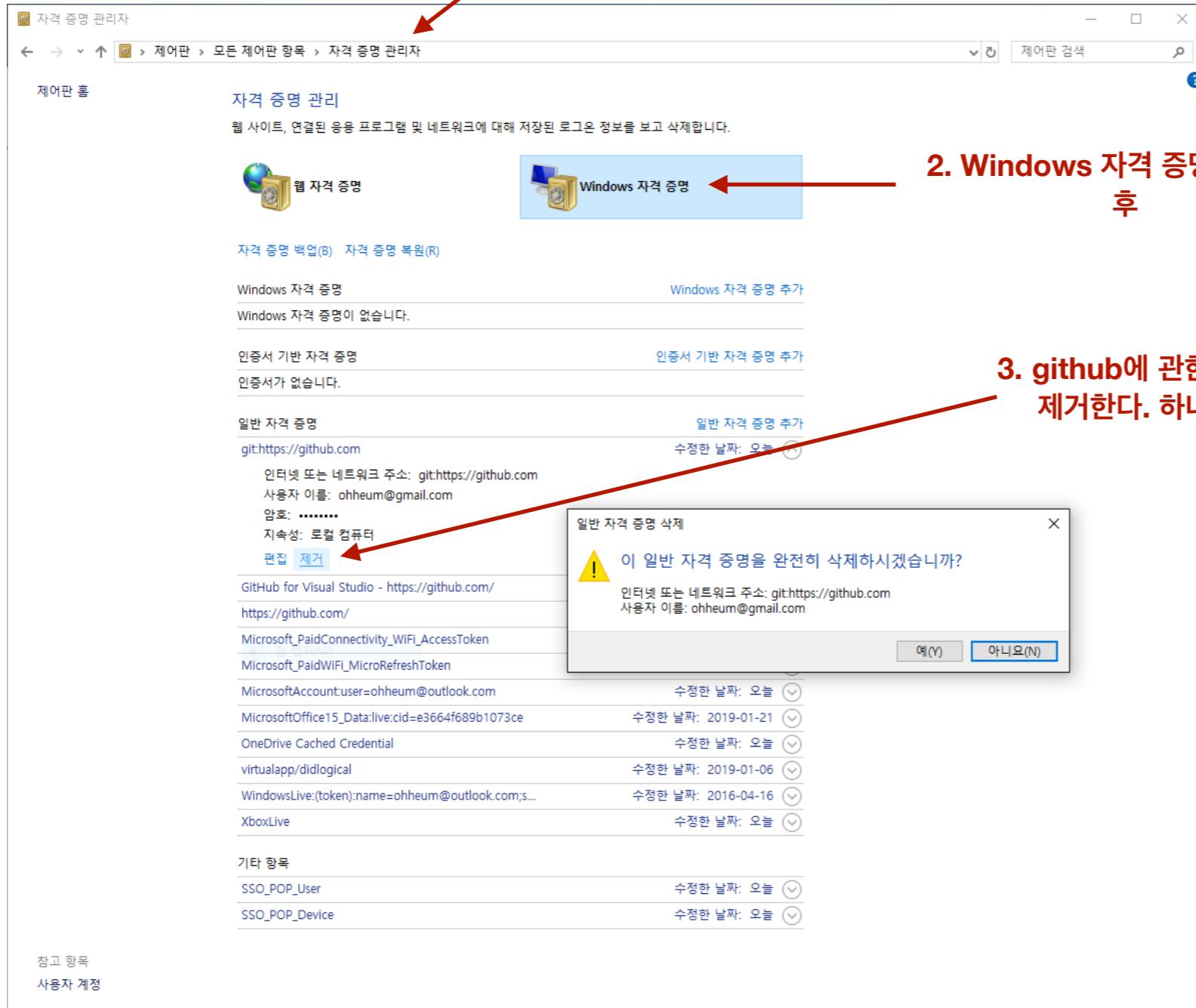
앞 페이지에서 오류가 생기지 않고 git push 명령이 수행되었다면 github repository에 이렇게 Visual Studio 프로젝트가 저장된다.

앞의 두 페이지의 과정은 반복해도 된다. 즉 과제를 제출한 후에 Visual Studio에서 코드를 수정하거나, 파일을 추가 혹은 삭제한 후 앞 페이지의 3개의 명령들

```
$ git add --all
$ git commit -m "New Message"
$ git push
```

를 다시 실행하면 수정된 사항이 반영된다.

1. git push 명령에서 24페이지에서와 같은 오류가 생기면 “제어판>모든 제어판 항목>자격증명 관리자”를 찾아 간다.

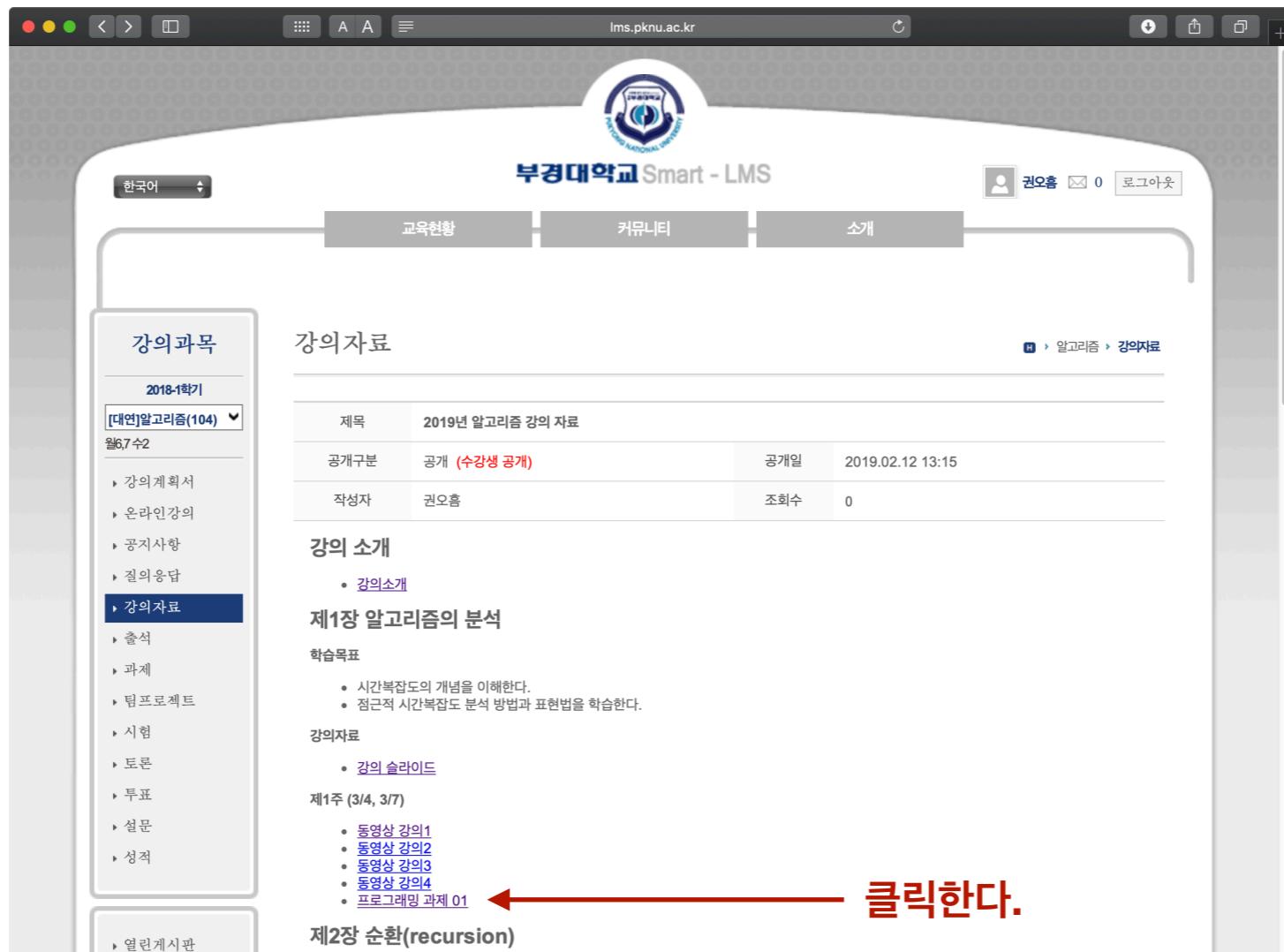


3. github에 관한 항목들을 찾아서 모두 제거한다. 하나 이상 있을 수 있다.

4. 그런 다음 git push 명령을 다시 실행하고 이렇게 Login 창이 뜨면 로그인 한다.



④ Smart LMS에서 프로그래밍 과제 링크를 클릭하면 언제든지 github에 제출한 자신의 과제에 다시 접근할 수 있다.



- ⦿ git push를 이용해서 과제를 제출한 후 Github Web 상에서 소스코드나 README.md 파일 등을 수정했다고 하자.
- ⦿ 그러면 내 PC의 local repository와 Github에 저장된 repository가 일치하지 않게 될 것이다.
- ⦿ 그럴 경우 PC에서 git pull 명령으로 동기화할 수 있다.

The screenshot shows a GitHub pull request interface and a terminal window side-by-side.

Github Web Interface (Left):

- Branch: master
- New pull request
- Create new file, Upload files, Find file, Clone or download
- ohkwn Update README.md (Latest commit eea5c73 13 seconds ago)
- README.md (Update README.md, 12 seconds ago)
- prog_assign01.pdf (Add files via upload, 2 days ago)
- README.md (diff view)
- Content: 알고리즘 프로그래밍 과제 01
이름: 홍길동
학번: 20171234

Terminal Session (Right):

```

MINGW64:/c/Users/ohheum/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn ->
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ pwd
/c/Users/ohheum/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ cat README.md
# 프로그래밍 과제 01
see the pdf file.

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ |

```

A red arrow points from the GitHub interface to the terminal output, indicating the state of the README.md file after a push.

Annotations:

1. git push로 과제를 제출한 후에 18, 19페이지의 방법으로 Web 상에서 README.md 파일을 수정했다고 가정하자.
다른 소스코드 파일을 수정한 경우에도 마찬가지이다.
2. 그러면 내 PC에 있는 README.md 파일은 이렇게 수정되기 이전 상태로 있을 것이다.
(cat 명령은 텍스트 파일을 화면에 보여준다.)

```
MINGW64:/c/Users/ohheum/Desktop/Spring2019/Algorithm2019/pa-01-... - X
ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ pwd
/c/Users/ohheum/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ cat README.md
# 프로그래밍 과제 01

see the pdf file.

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ git pull ←
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/PKNU-IT-ALGORITHM2019/pa-01-ohkwn
  cd209f0..eea5c73 master      -> origin/master
Updating cd209f0..eea5c73
Fast-forward
 README.md | 5 +---+
 1 file changed, 3 insertions(+), 2 deletions(-)

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ cat README.md ←
# 알고리즘 프로그래밍 과제 01
이름 : 흥길동

학번 : 20171234

ohheum@miniWin MINGW64 ~/Desktop/Spring2019/Algorithm2019/pa-01-ohkwn (master)
$ |
```

3. git pull 명령으로 내 PC의 local repository를 Github에 있는 최신 상태로 동기화할 수 있다.

4. cat 명령으로 README.md 파일을 보면 최신 상태로 변경되었음을 알 수 있다.

