



***3**

과제 문제와 해결

UT-NodeJS / 03.17.2023

ut-nodejs.github.io



Assignments / 과제

과제를 완료할 때 해결해야 할 3가지 주요 문제가 있다.



01. Access / 과제 받기

모든 과제를 각 수업이 시작될 때부터 [홈페이지](#)에서 찾을 수 있다.

02. Submit / 과제 제출

과제를 제출하는 가장 쉬운 방법은 VS Code 내에서 직접 커밋 및 푸시를 수행하는 것이다.

03. Check / 테스트 결과 확인

터미널에서 로컬로 코드를 확인하거나 GitHub Actions에서 온라인으로 확인할 수 있다.



01

Access

과제 받기



과제를 어떻게 받을 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. 깃 클래스룸으로 **VS Code**를 직접 열기
2. 명령줄에서 **git clone**
3. 직접 코드 다운로드



과제를 어떻게 받을 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. 깃 클래스룸으로 **VS Code**를 직접 열기

2. 명령줄에서 **git clone**

3. 직접 코드 다운로드

1.

깃 클래스룸으로 VS Code를 열기

1. [홈페이지](#),
2. 과제 참여 링크,
3. 개별 저장소,
4. GitHub 조직

New Tab x +

← → ↻ 🔒 http://

🏠 | 소개 | 일정 | 슬라이드 | **[과제]** | 프로젝트 | 시험 | 성적 | 설문 ↗

Web Programming Application 2023

한국교통대학교, 충주 | KNUT (Korea National University of Transportation)

Practice / 과제

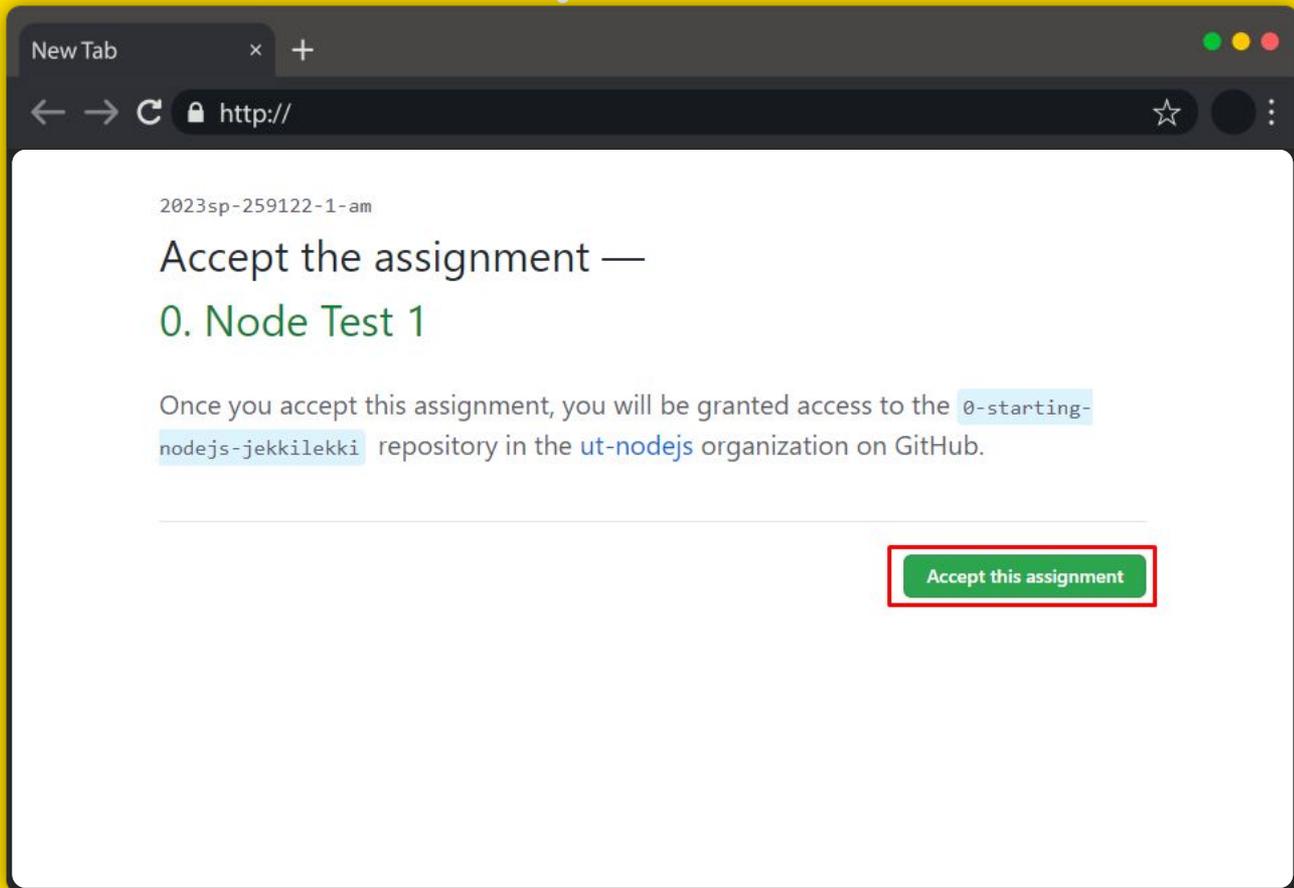
Practice Labs will be assigned and collected in **GitHub** Classroom and accessible there. Follow the instructions below to understand how to access and submit labs.

Week	Date	과제	am	pm
1	3월3일	-	-	-
2	3월10일	0. Starting Node / 노드 시작	오전	오후

1.

깃 클래스룸으로 VS Code를 열기

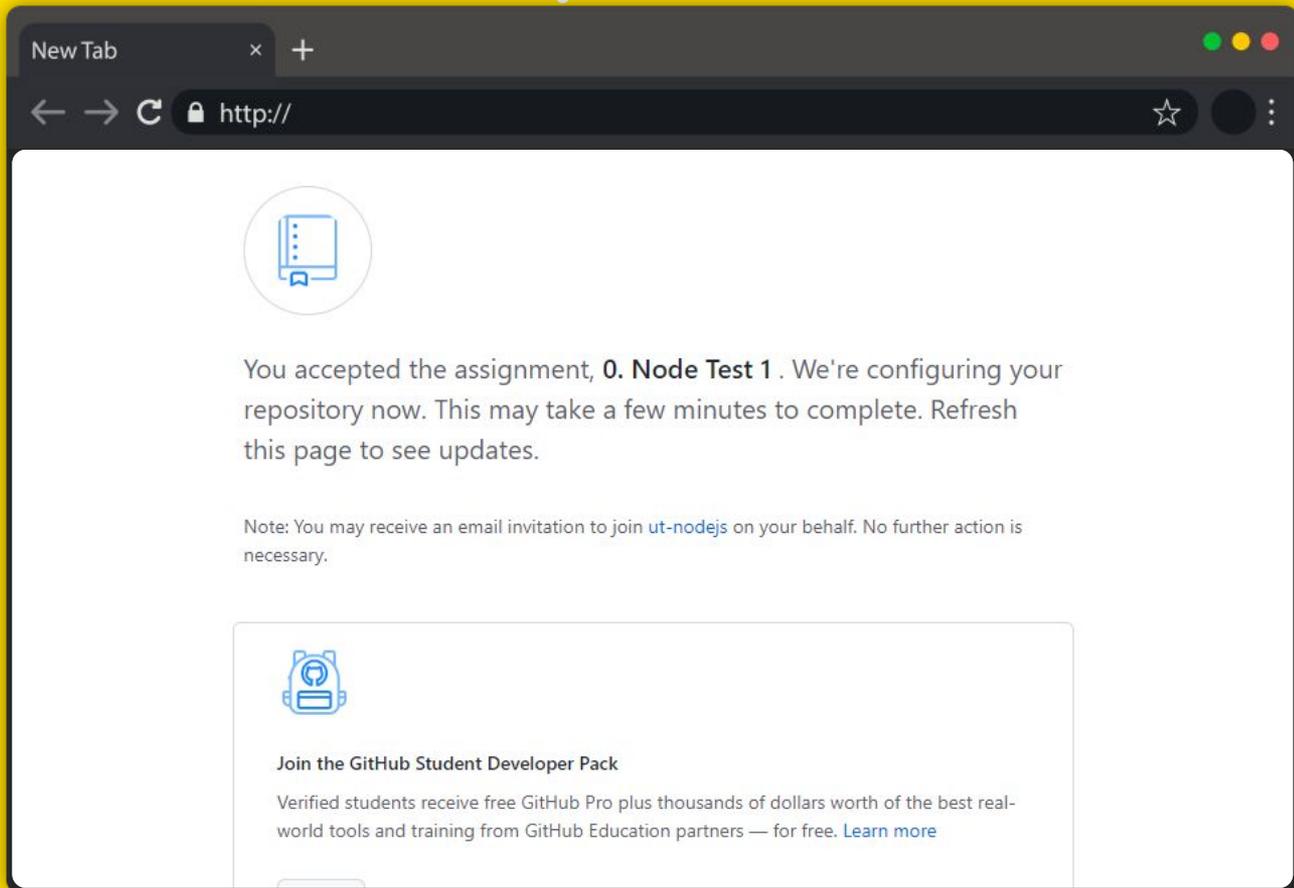
1. 홈페이지,
2. 과제 참여 링크,
3. 개별 저장소,
4. GitHub 조직



1.

깃 클래스룸으로 VS Code를 열기

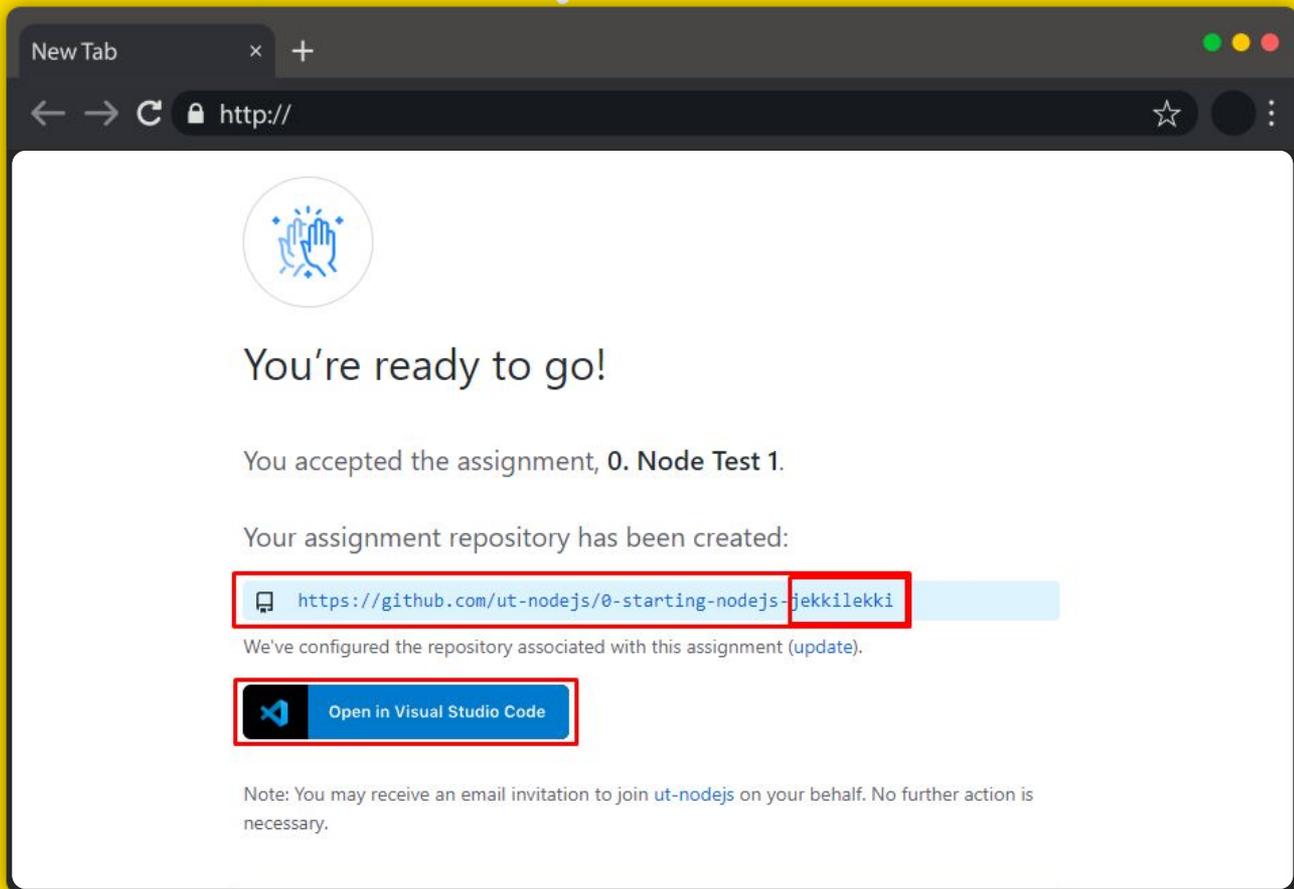
1. 홈페이지,
2. 과제 참여 링크,
3. 개별 저장소,
4. GitHub 조직



1.

깃 클래스룸으로 VS Code를 열기

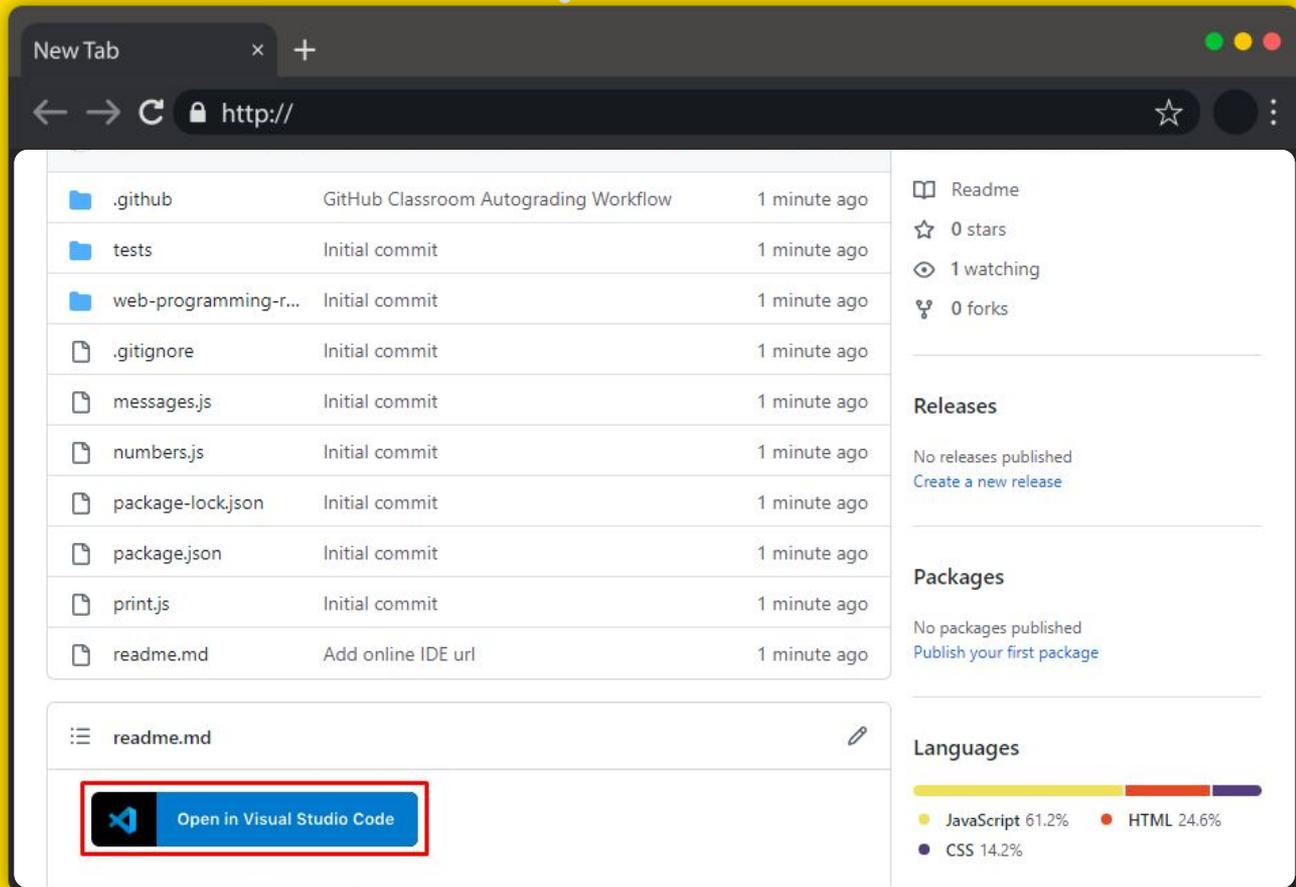
1. 홈페이지,
2. 과제 참여 링크,
3. 개별 저장소,
4. GitHub 조직



1.

깃 클래스룸으로 VS Code를 열기

1. 홈페이지,
2. 과제 참여 링크,
3. 개별 저장소,
4. GitHub 조직



The screenshot shows a web browser window displaying a GitHub repository. The address bar shows 'http://'. The repository name is partially visible as 'web-programming-r...'. The file list includes:

File/Folder	Commit Message	Time
.github	GitHub Classroom Autograding Workflow	1 minute ago
tests	Initial commit	1 minute ago
web-programming-r...	Initial commit	1 minute ago
.gitignore	Initial commit	1 minute ago
messages.js	Initial commit	1 minute ago
numbers.js	Initial commit	1 minute ago
package-lock.json	Initial commit	1 minute ago
package.json	Initial commit	1 minute ago
print.js	Initial commit	1 minute ago
readme.md	Add online IDE url	1 minute ago

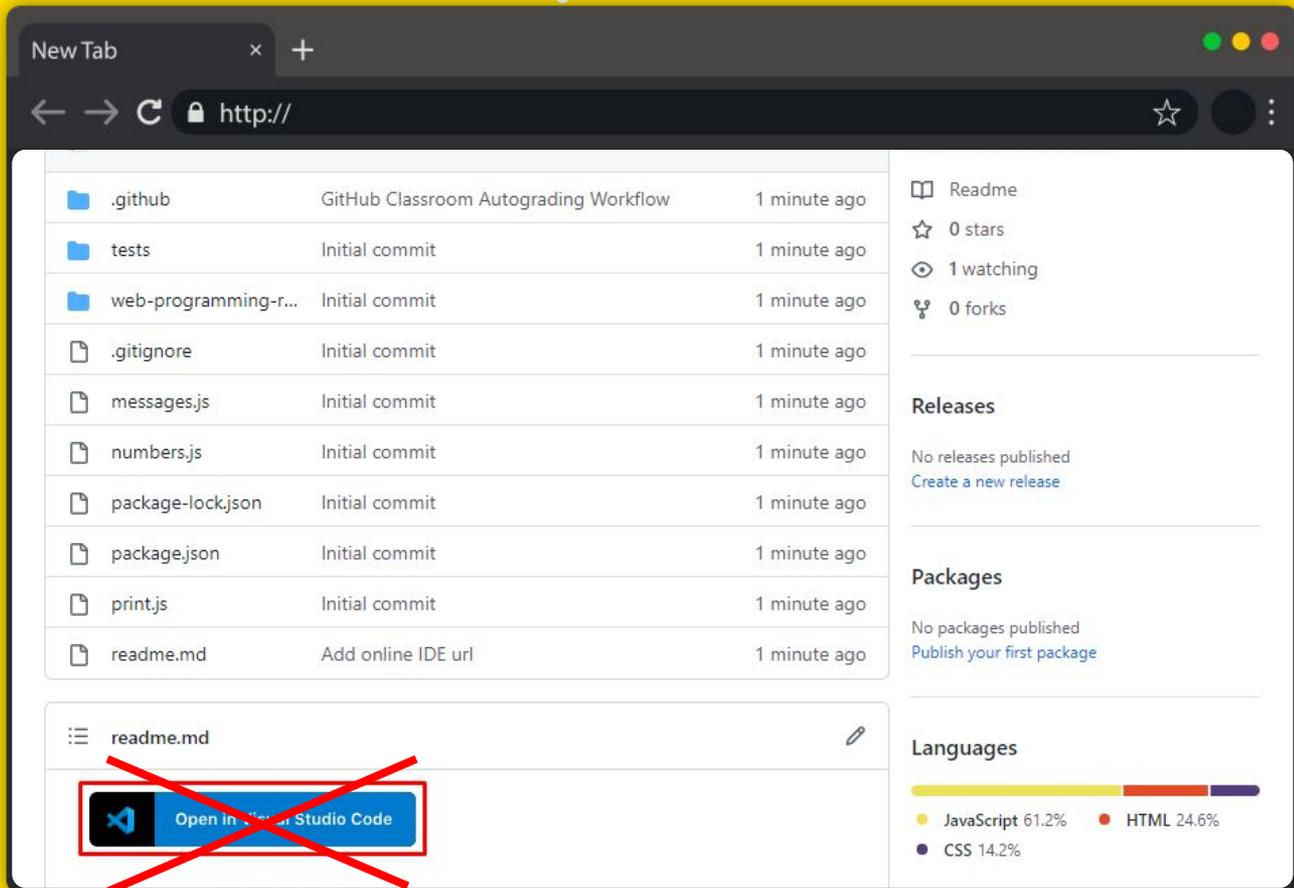
Below the file list, the 'readme.md' file is selected. A button labeled 'Open in Visual Studio Code' is highlighted with a red box. The right sidebar shows repository statistics: Readme, 0 stars, 1 watching, 0 forks, and a language usage chart for JavaScript (61.2%), HTML (24.6%), and CSS (14.2%).

1.

깃 클래스룸으로 VS Code를 열기

1. 홈페이지,
2. 과제 참여 링크,
3. 개별 저장소,
4. GitHub 조직

못 열면?





과제를 어떻게 받을 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. 깃 클래스룸으로 **VS Code**를 직접 열기

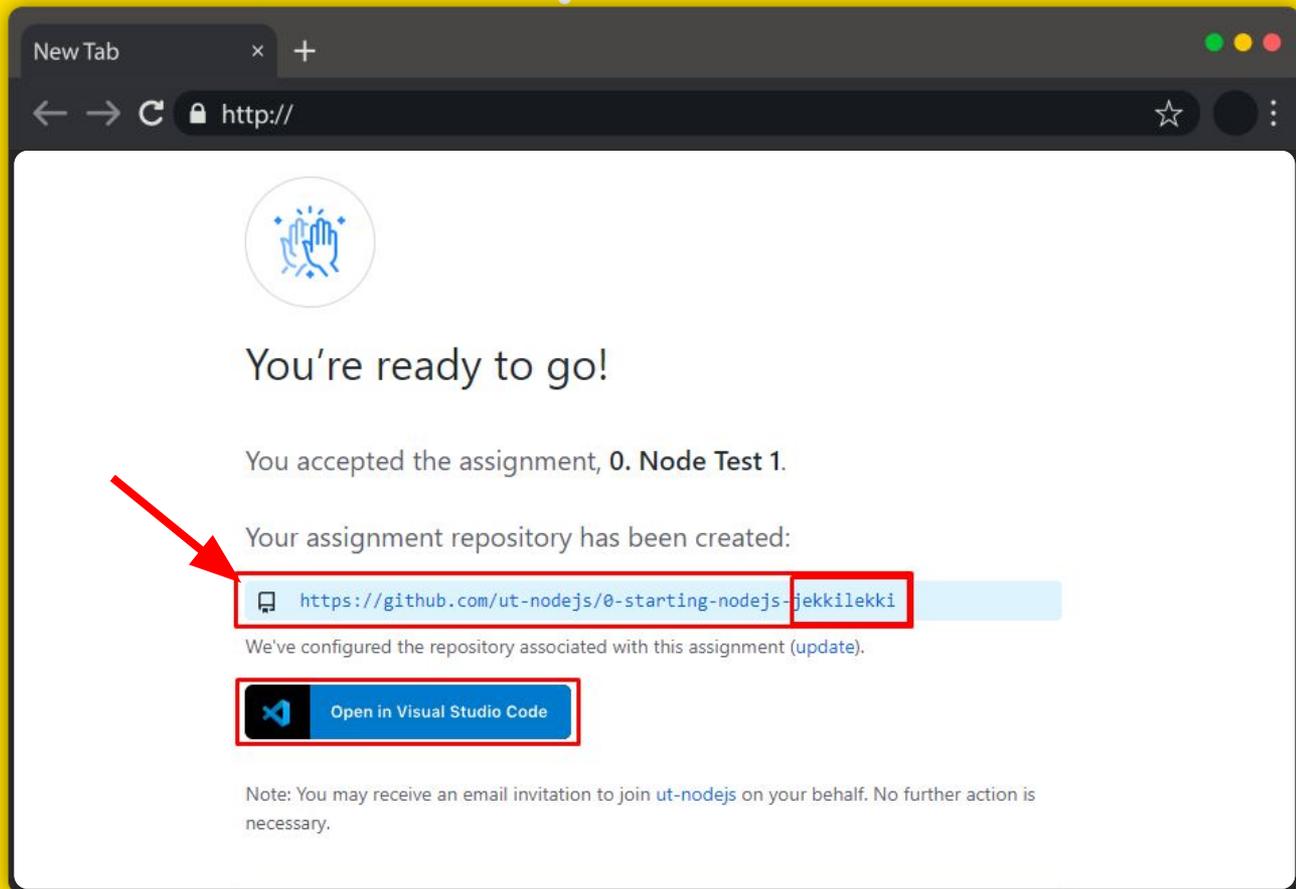
2. 명령줄에서 **git clone**

3. 직접 코드 다운로드

2.

명령줄에서 **git clone**

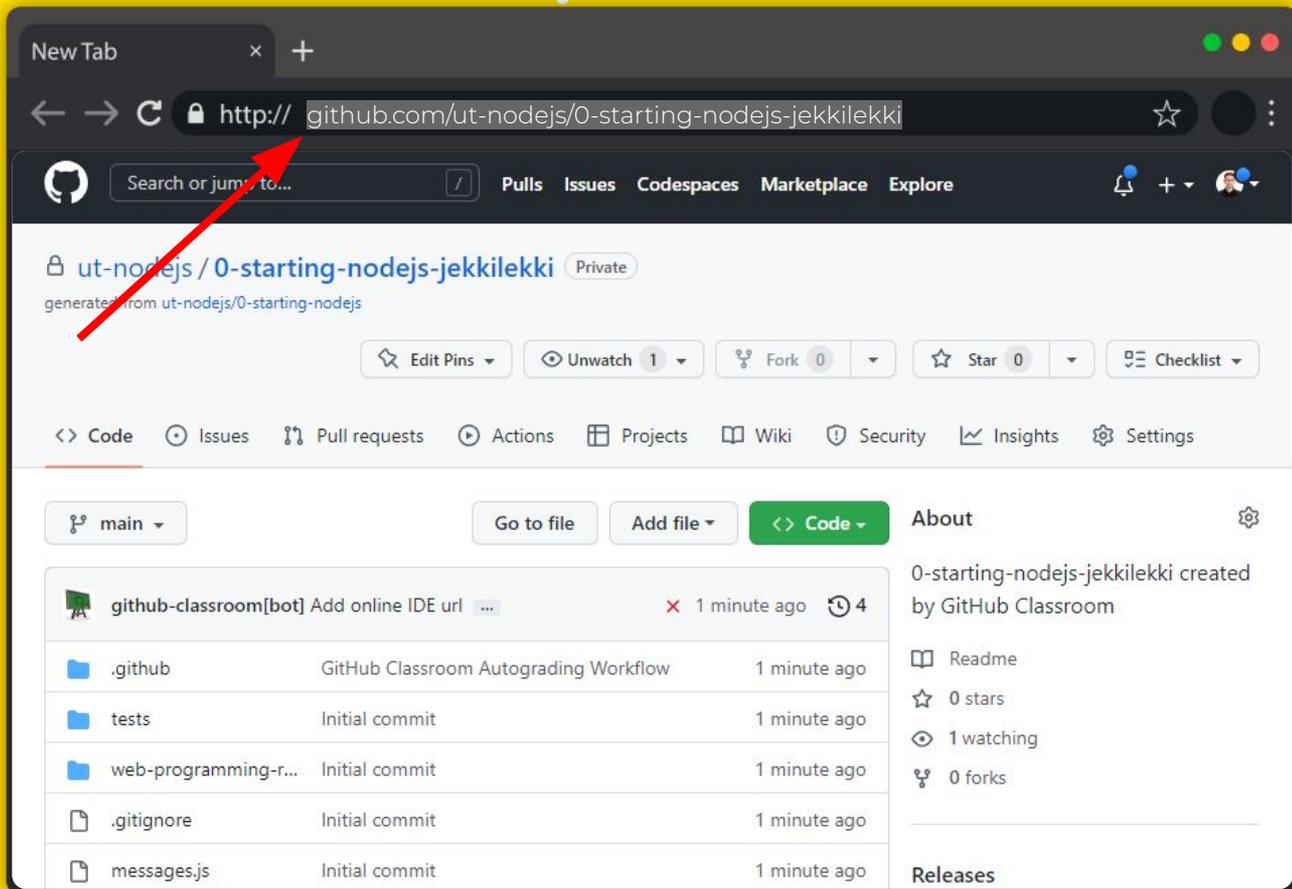
1. 온라인 저장소로 가고
2. **URL 복사**
3. 바탕화면에서 새 폴더
4. Git Bash에서 폴더로 cd
5. git clone URL



2.

명령줄에서 git clone

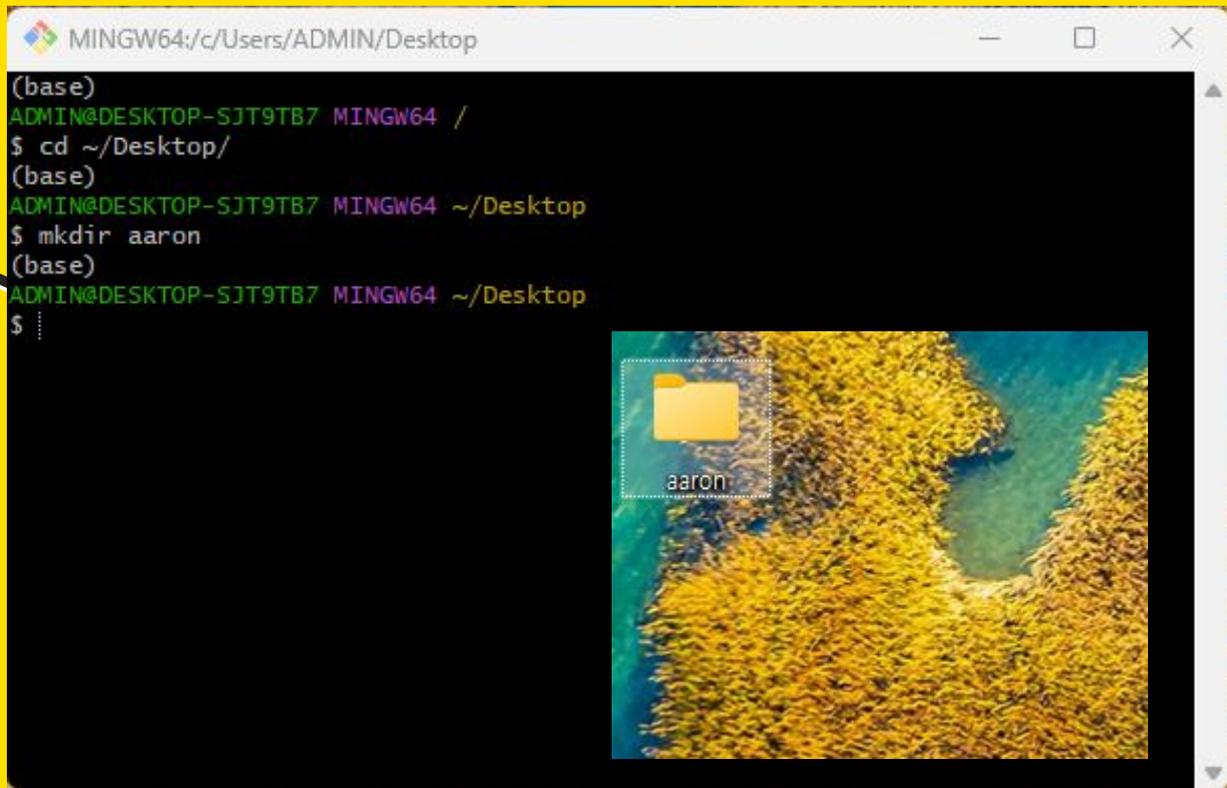
1. 온라인 저장소로 가고
2. **URL** 복사
3. 바탕화면에서 새 폴더
4. Git Bash에서 폴더로 cd
5. git clone URL



2.

명령줄에서 git clone

1. 온라인 저장소로 가고
2. URL 복사
3. 바탕화면에서 새 폴더
4. Git Bash에서 폴더로 cd
5. git clone URL



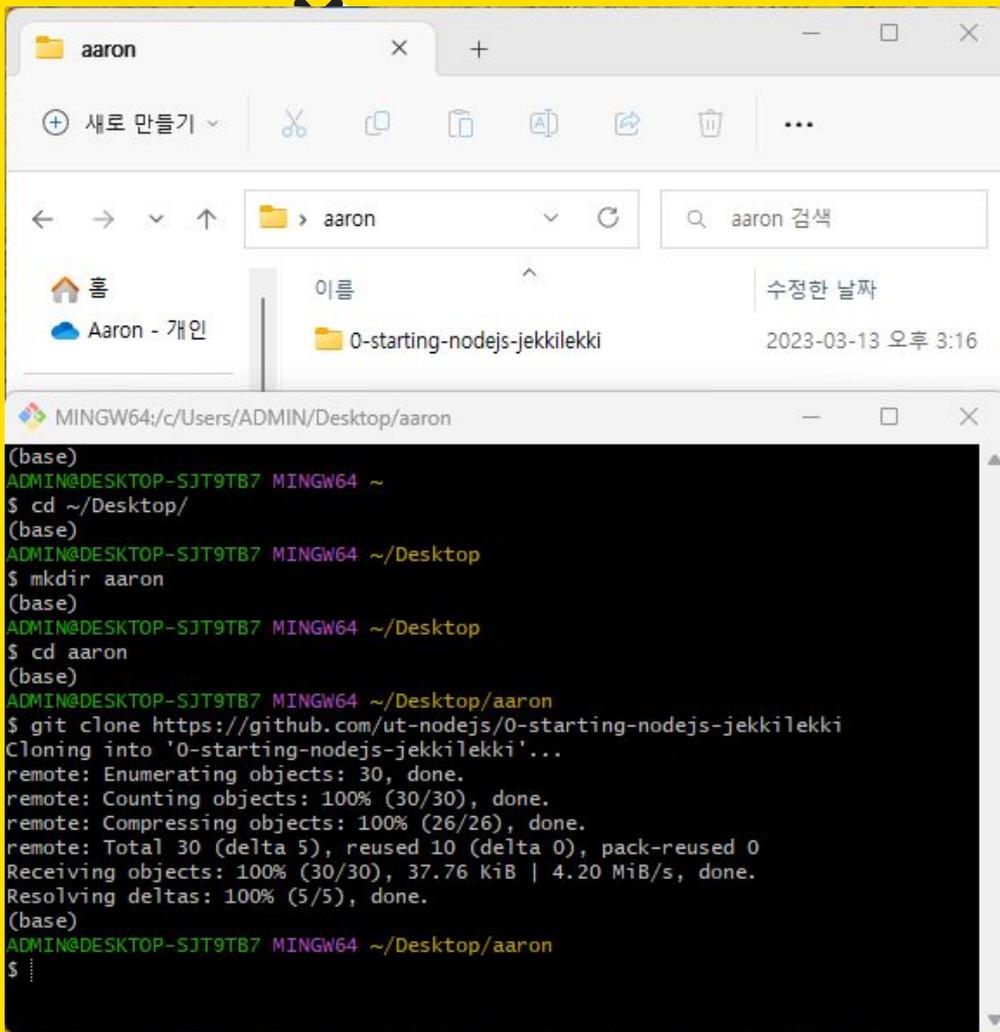
```
MINGW64:/c/Users/ADMIN/Desktop
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 /
$ cd ~/Desktop/
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 ~/Desktop
$ mkdir aaron
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 ~/Desktop
$
```



2.

명령줄에서 git clone

1. 온라인 저장소로 가고
2. URL 복사
3. 바탕화면에서 새 폴더
4. **Git Bash**에서 폴더로 **cd**
5. **git clone** URL



```
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 ~
$ cd ~/Desktop/
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 ~/Desktop
$ mkdir aaron
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 ~/Desktop
$ cd aaron
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 ~/Desktop/aaron
$ git clone https://github.com/ut-nodejs/0-starting-nodejs-jekkilekki
Cloning into '0-starting-nodejs-jekkilekki'...
remote: Enumerating objects: 30, done.
remote: Counting objects: 100% (30/30), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 30 (delta 5), reused 10 (delta 0), pack-reused 0
Receiving objects: 100% (30/30), 37.76 KiB | 4.20 MiB/s, done.
Resolving deltas: 100% (5/5), done.
(base)
ADMIN@DESKTOP-SJT9TB7 MINGW64 ~/Desktop/aaron
$
```



과제를 어떻게 받을 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. 깃 클래스룸으로 VS Code를 직접 열기

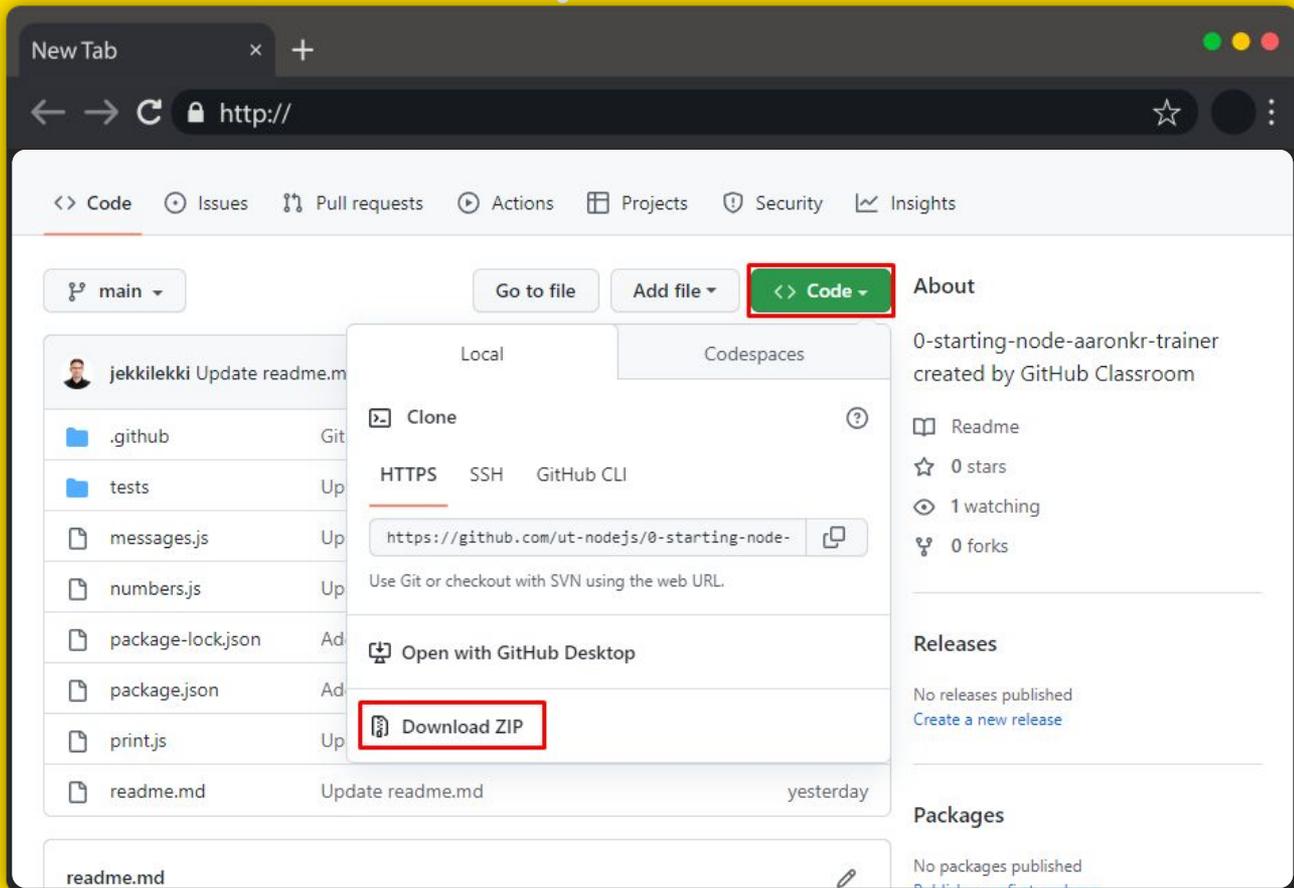
2. 명령줄에서 git clone

3. 직접 코드 다운로드

3.

직접 코드 다운로드

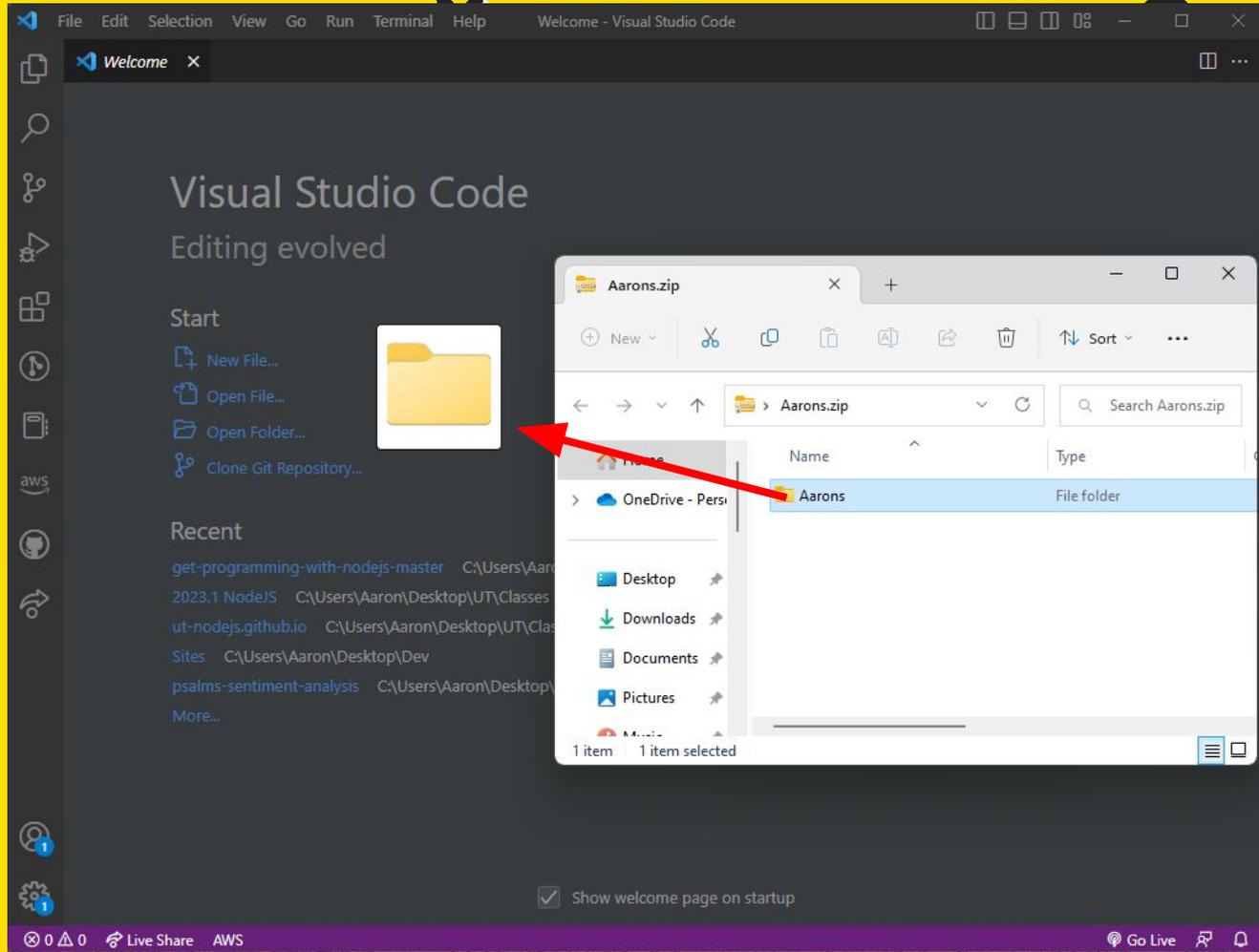
1. 온라인 저장소로 가고
2. 다운로드 ZIP
3. 압축
4. VS Code로 열고
5. VS Code터미널에서
 - a. git remote add URL



3.

직접 코드 다운로드

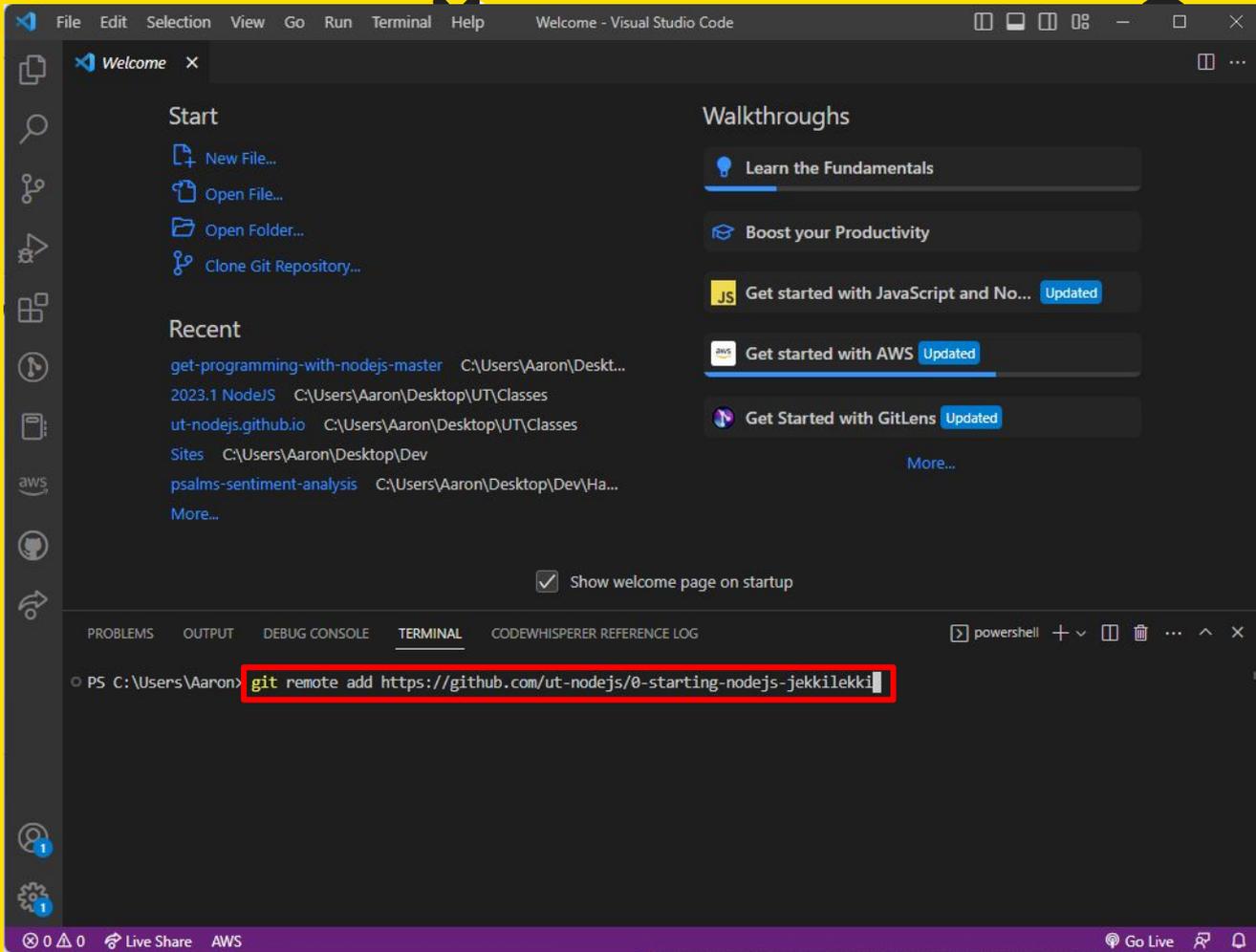
1. 온라인 저장소로 가고
2. 다운로드 ZIP
3. 압축
4. VS Code로 열고
5. VS Code터미널에서
 - a. git remote add URL



3.

직접 코드 다운로드

1. 온라인 저장소로 가고
2. 다운로드 ZIP
3. 압축
4. VS Code로 열고
5. **VS Code** 터미널에서
 - a. **git remote add URL**





02

Submit

과제 제출



과제를 어떻게 제출할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. VS Code에서 직접 **commit & push**

2. 명령줄에서 **git commit & push**

3. 직접 코드 업로드



과제를 어떻게 제출할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



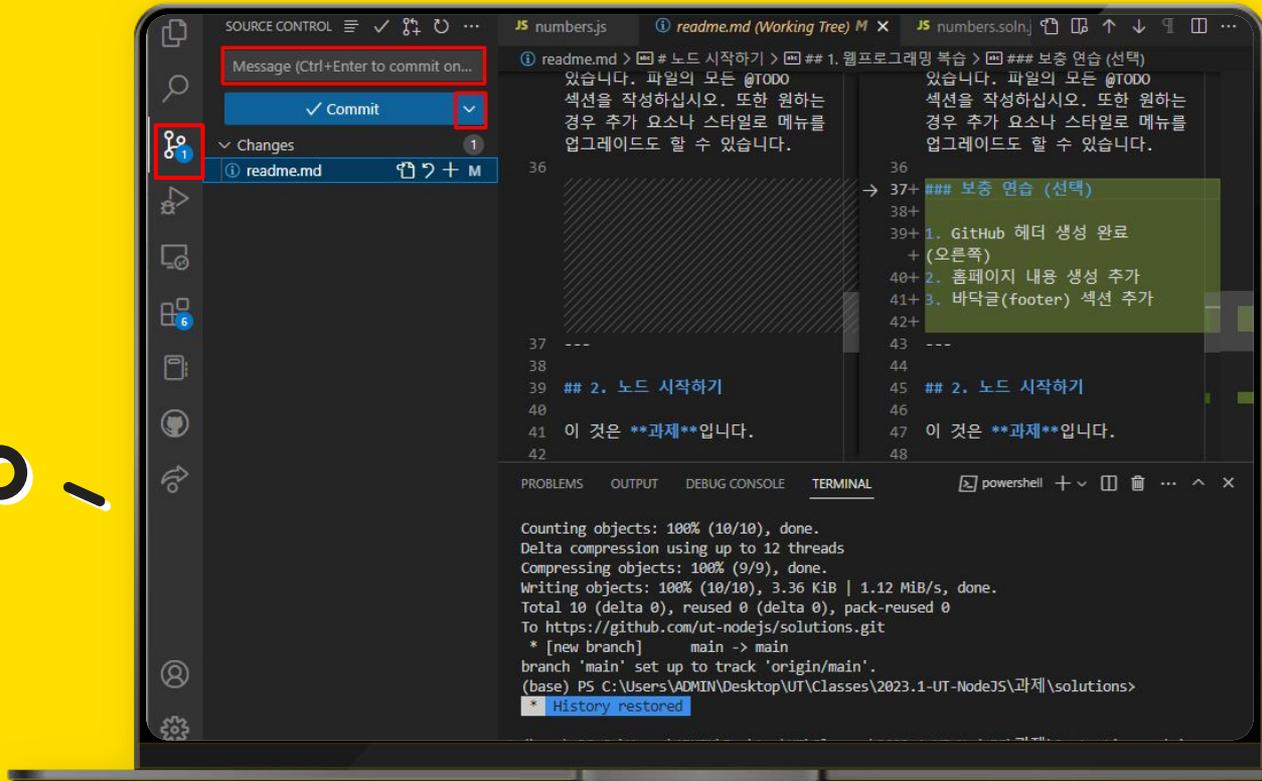
1. VS Code에서 직접 **commit & push**

2. 명령줄에서 **git commit & push**

3. 직접 코드 업로드

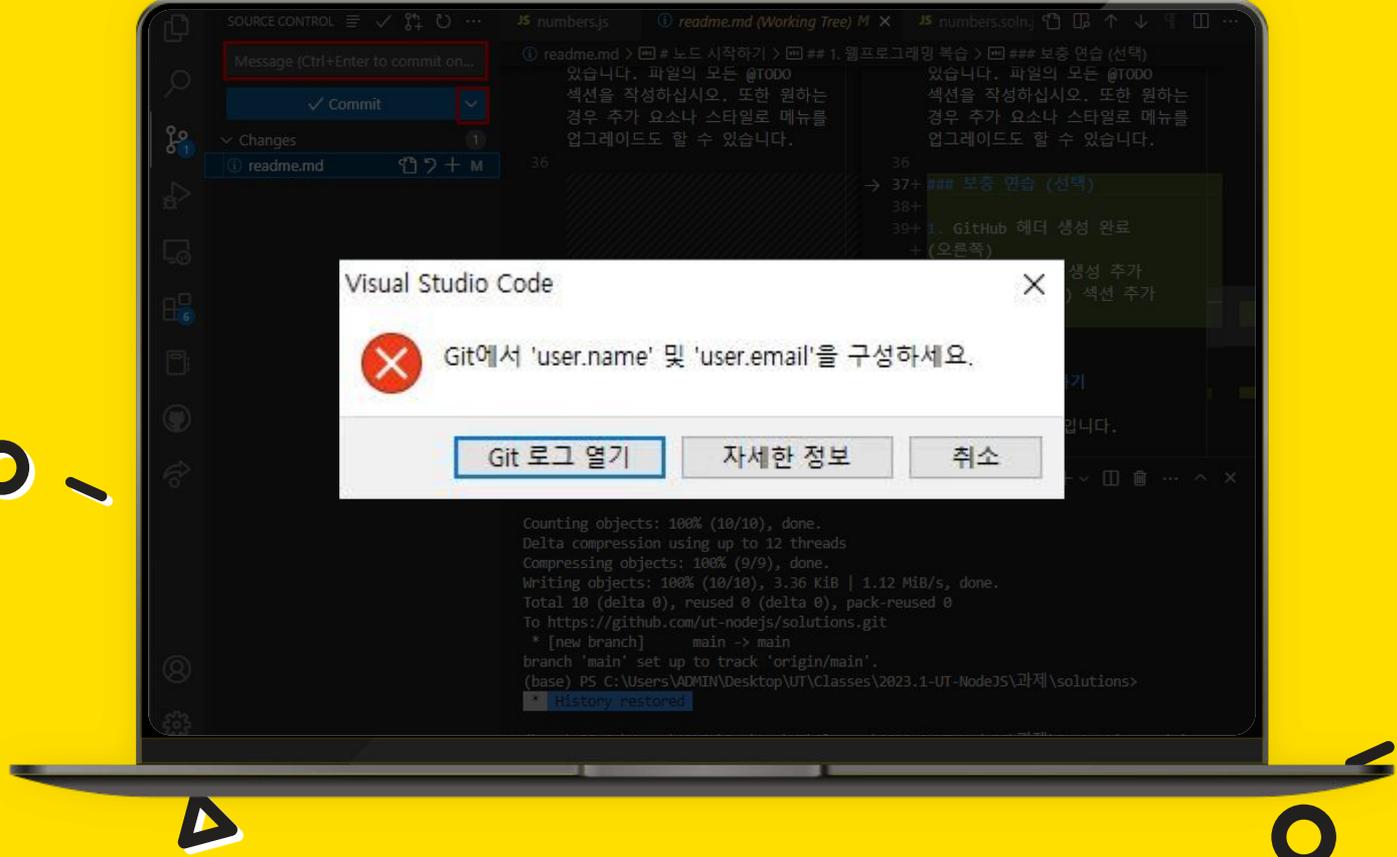
1. VS Code에서 git commit git push

1. 과제 참여 링크,
2. 코드 받았다
3. 코딩,
4. **commit & push**



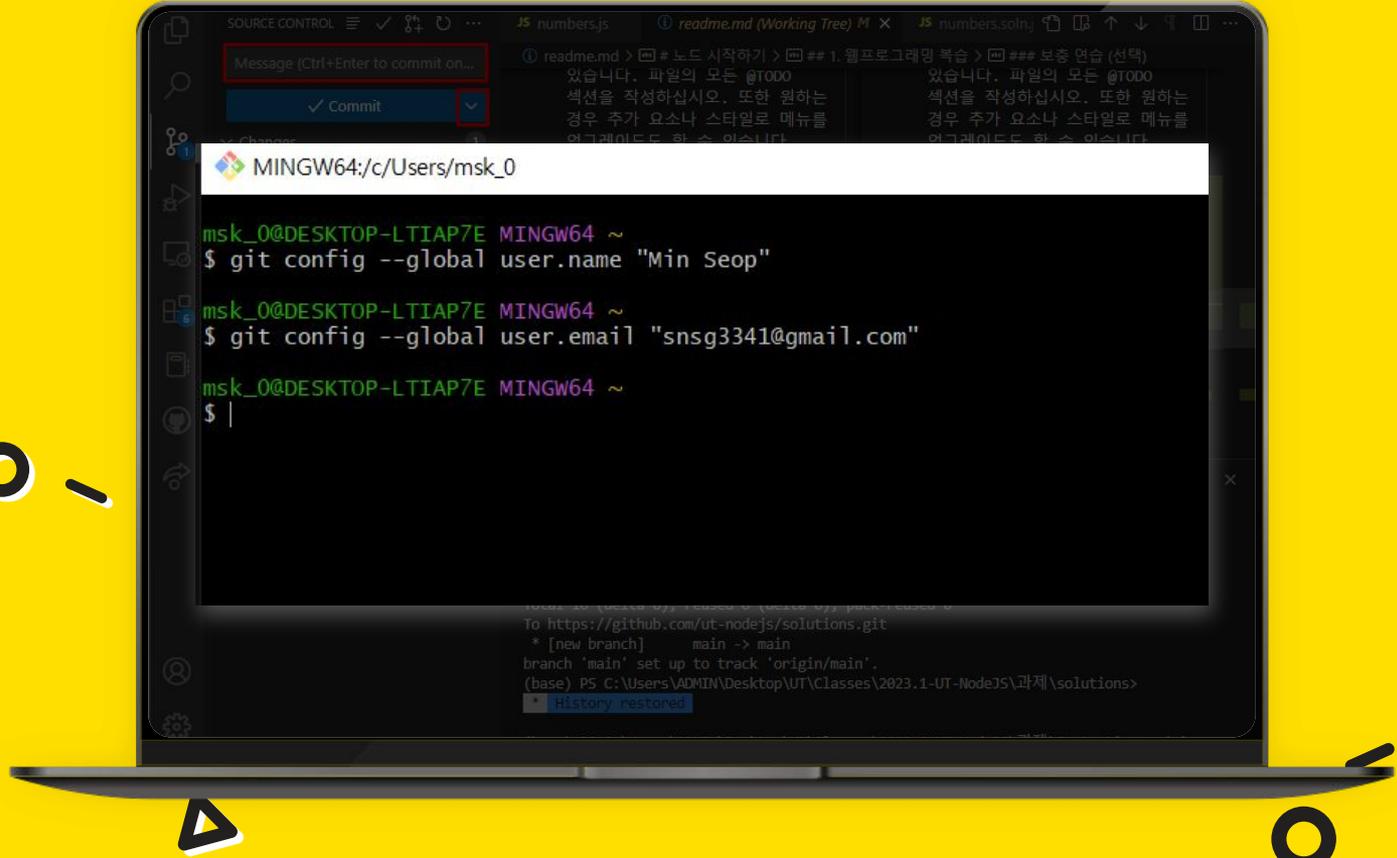
1. VS Code에서 git commit git push

1. 과제 참여 링크,
2. 코드 받았다
3. 코딩,
4. commit & push
5. 오류 생기면



1. VS Code에서 git commit git push

1. 과제 참여 링크,
2. 코드 받았다
3. 코딩,
4. commit & push
5. 오류 생기면
user.name입력
user.email입력





과제를 어떻게 제출할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. VS Code에서 직접 **commit & push**

2. 명령줄에서 **git commit & push**

3. 직접 코드 업로드



2.

명령줄에서

git commit git push

1. **git init**
2. git remote add origin <URL>
3. git add .
4. git commit -m "MSG"
5. git push -u origin master

```
MINGW64/c:/Users/Aaron/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main
Aaron@DESKTOP-PTD9GI3 MINGW64 ~
$ cd ~/Desktop/UT/Classes/2023.1\ NodeJS/0-web-programming-review-node-start-jekkilekki-main
Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/c/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main
$ git init
Initialized empty Git repository in C:/Users/Aaron/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main/.git/
Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/c/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$
```





2.

명령줄에서

git commit git push

1. git init
2. **git remote add origin <URL>**
3. git add .
4. git commit -m "MSG"
5. git push -u origin master

```
MINGW64:/c:/Users/Aaron/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main
Aaron@DESKTOP-PTD9GI3 MINGW64 ~
$ cd ~/Desktop/UT/Classes/2023.1\ NodeJS/0-web-programming-review-node-start-jekkilekki-main

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/c/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main
$ git init
Initialized empty Git repository in C:/Users/Aaron/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main/.git/

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/c/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$ git remote add origin https://github.com/ut-nodejs/0-web-programming-review-node-start-jekkilekki-main

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/c/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$ |
```





2.

명령줄에서

git commit git push

1. git init
2. git remote add origin <URL>
3. **git add .**
4. git commit -m "MSG"
5. git push -u origin master



```

MINGW64:~/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main
$ git init
Initialized empty Git repository in C:/Users/Aaron/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main/.git/

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$ git remote add origin https://github.com/ut-nodejs/0-web-programming-review-node-start-jekkilekki-main

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$ git add .
warning: in the working copy of '.github/classroom/autograding.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.github/workflows/classroom.yml', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'messages.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'numbers.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'print.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'readme.md', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tests/messages.test.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tests/numbers.test.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tests/print.test.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'web-programming-review/web-programming-review.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'web-programming-review/web-programming-review.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'web-programming-review/web-programming-review.js', LF will be replaced by CRLF the next time Git touches it

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$ |

```





2.

명령줄에서

git commit git push

1. git init
2. git remote add origin <URL>
3. git add .
4. **git commit -m "MSG"**
5. git push -u origin master

```
MINGW64:/c/Users/Aaron/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main
warning: in the working copy of 'print.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'readme.md', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tests/messages.test.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tests/numbers.test.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tests/print.test.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'web-programming-review/web-programming-review.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'web-programming-review/web-programming-review.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'web-programming-review/web-programming-review.js', LF will be replaced by CRLF the next time Git touches it

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/c/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$ git commit -m "Demo"
[master (root-commit) c76aa50] Demo
15 files changed, 3716 insertions(+)
create mode 100644 .github/classroom/autograding.json
create mode 100644 .github/workflows/classroom.yml
create mode 100644 .gitignore
create mode 100644 messages.js
create mode 100644 numbers.js
create mode 100644 package-lock.json
create mode 100644 package.json
create mode 100644 print.js
create mode 100644 readme.md
create mode 100644 tests/messages.test.js
create mode 100644 tests/numbers.test.js
create mode 100644 tests/print.test.js
create mode 100644 web-programming-review/web-programming-review.css
create mode 100644 web-programming-review/web-programming-review.html
create mode 100644 web-programming-review/web-programming-review.js

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/c/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$
```



2.

명령줄에서

git commit git push

1. git init
2. git remote add origin <URL>
3. git add .
4. git commit -m "MSG"
5. **git push -u origin master**

```
MINGW64~/c/Users/Aaron/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main
[master (root-commit) c76aa50] Demo
15 files changed, 3716 insertions(+)
create mode 100644 .github/classroom/autograding.json
create mode 100644 .github/workflows/classroom.yml
create mode 100644 .gitignore
create mode 100644 messages.js
create mode 100644 numbers.js
create mode 100644 package-lock.json
create mode 100644 package.json
create mode 100644 print.js
create mode 100644 readme.md
create mode 100644 tests/messages.test.js
create mode 100644 tests/numbers.test.js
create mode 100644 tests/print.test.js
create mode 100644 web-programming-review/web-programming-review.css
create mode 100644 web-programming-review/web-programming-review.html
create mode 100644 web-programming-review/web-programming-review.js

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$ git push -u origin master
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 16 threads
Compressing objects: 100% (19/19), done.
Writing objects: 100% (22/22), 35.65 KiB | 2.38 MiB/s, done.
Total 22 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/ut-nodejs/0-web-programming-review-node-start-jekkilekki/pull/new/master
remote:
To https://github.com/ut-nodejs/0-web-programming-review-node-start-jekkilekki
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

Aaron@DESKTOP-PTD9GI3 MINGW64 ~/Desktop/UT/Classes/2023.1 NodeJS/0-web-programming-review-node-start-jekkilekki-main (master)
$
```





과제를 어떻게 제출할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. VS Code에서 직접 **commit & push**

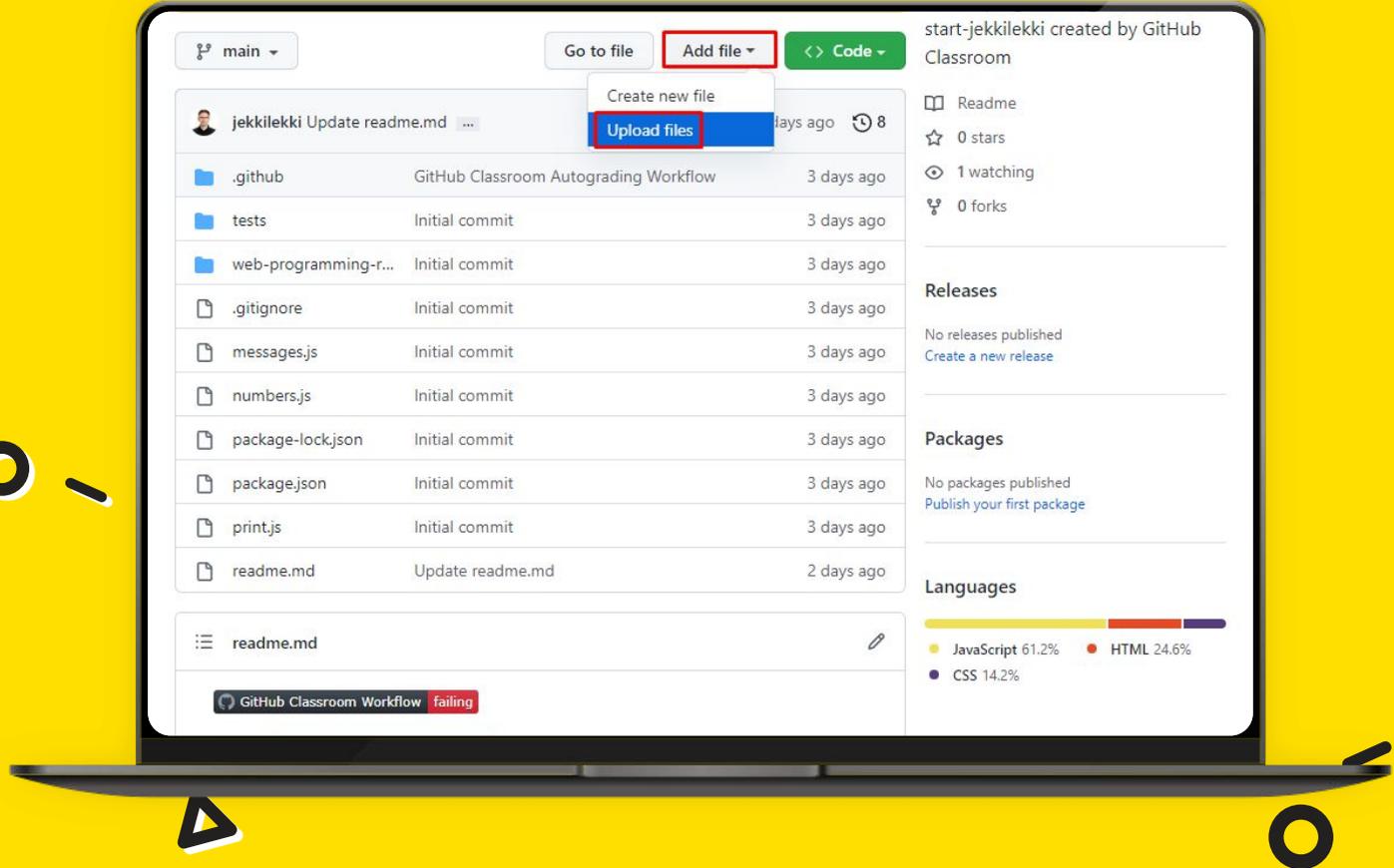
2. 명령줄에서 **git push**

3. 직접 코드 업로드

3.

직접 코드 업로드

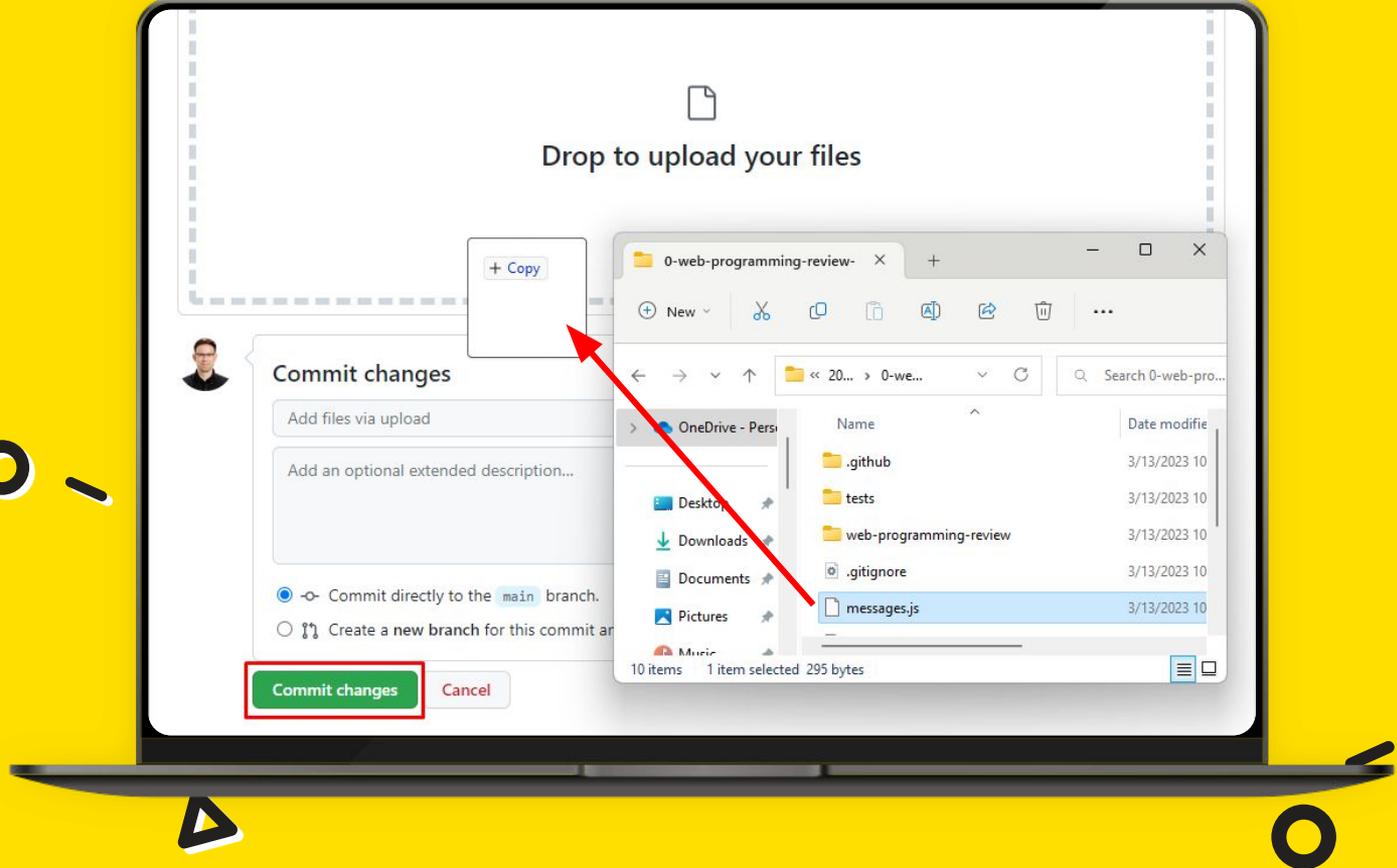
1. 저장소에서
2. **“Add files -> Upload files”**
3. Commit changes



3.

직접 코드 업로드

1. 저장소에서
2. "Add files -> Upload files"
3. **Commit changes**





03

Check

테스트 결과 확인



과제를 어떻게 확인할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. 로컬 컴퓨터 명령줄에서 코드 실행
2. 명령줄에서 **npm install + npm test**
3. 온라인 **GitHub Actions**에서
 - * **PASS/FAIL** 배지 추가 하면?



과제를 어떻게 확인할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. 로컬 컴퓨터 명령줄에서 코드 실행

2. 명령줄에서 **npm install + npm test**

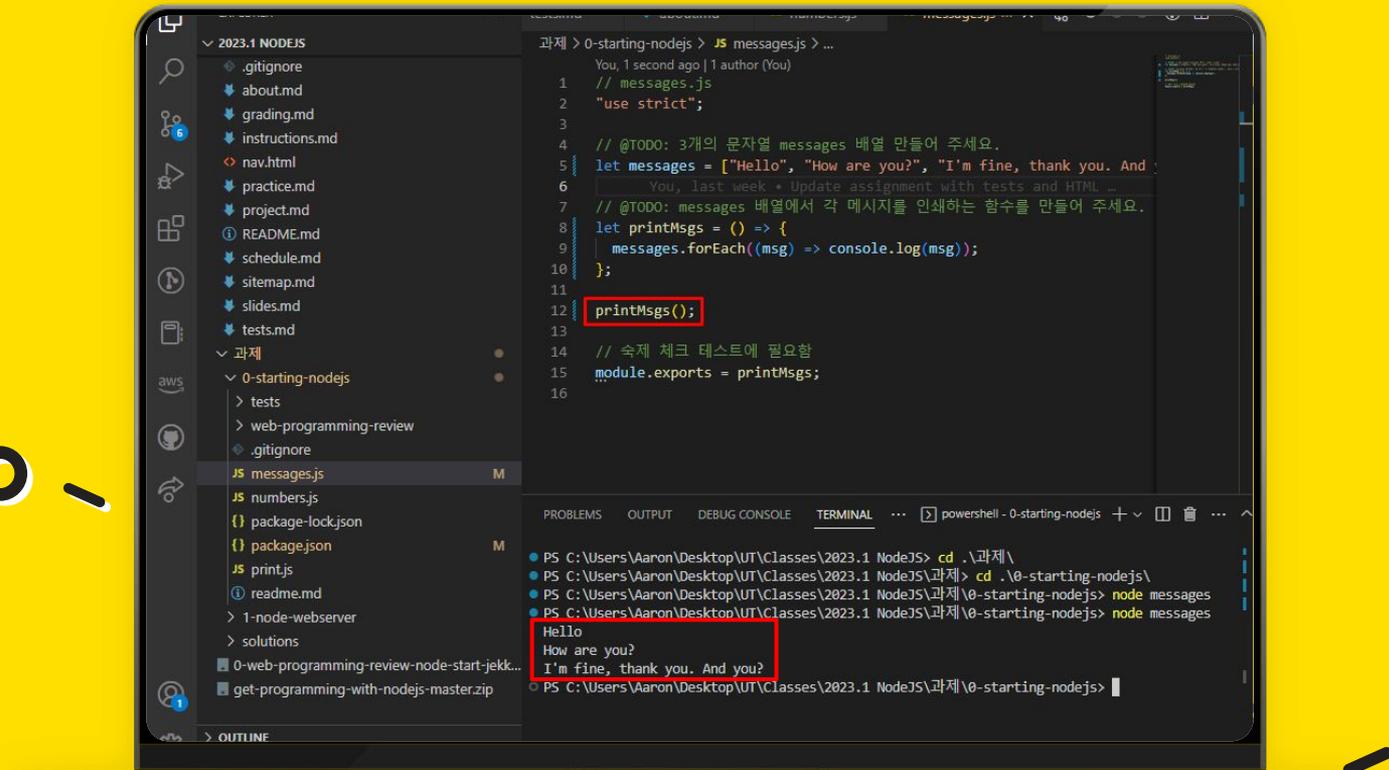
3. 온라인 **GitHub Actions**에서

* **PASS/FAIL** 배지 추가 하면?

1.

로컬 컴퓨터
명령줄에서
코드 실행

1. 함수를 생성하기
2. 함수를 돌리기





과제를 어떻게 확인할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



1. 로컬 컴퓨터 명령줄에서 코드 실행

2. 명령줄에서 **npm install + npm test**

3. 온라인 **GitHub Actions**에서

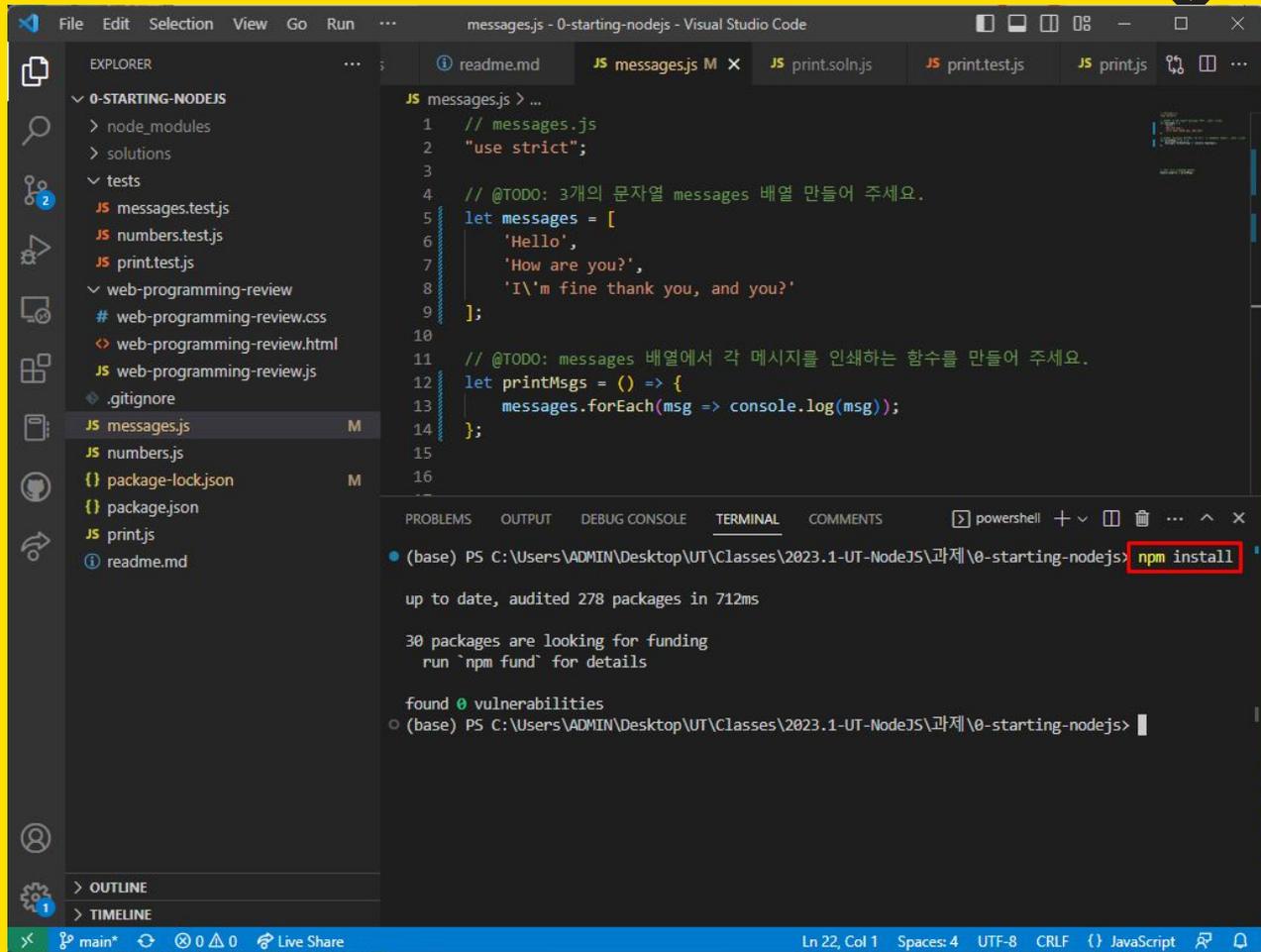
*** PASS/FAIL** 배지 추가 하면?

2.

명령줄에서

npm install + npm test

1. npm install는 Jest 테스트 패키지 설치
2. npm test는 Jest 테스트 패키지 실행



```
File Edit Selection View Go Run ... messages.js - 0-starting-nodejs - Visual Studio Code
EXPLORER
0-STARTING-NODEJS
  > node_modules
  > solutions
  tests
    JS messages.test.js
    JS numbers.test.js
    JS print.test.js
  web-programming-review
    # web-programming-review.css
    <> web-programming-review.html
    JS web-programming-review.js
    .gitignore
    JS messages.js M
    JS numbers.js
    {} package-lock.json M
    {} package.json
    JS print.js
    readme.md

messages.js > ...
1 // messages.js
2 "use strict";
3
4 // @TODO: 3개의 문자열 messages 배열 만들어 주세요.
5 let messages = [
6   'Hello',
7   'How are you?',
8   'I\'m fine thank you, and you?'
9 ];
10
11 // @TODO: messages 배열에서 각 메시지를 인쇄하는 함수를 만들어 주세요.
12 let printMsgs = () => {
13   messages.forEach(msg => console.log(msg));
14 };
15
16

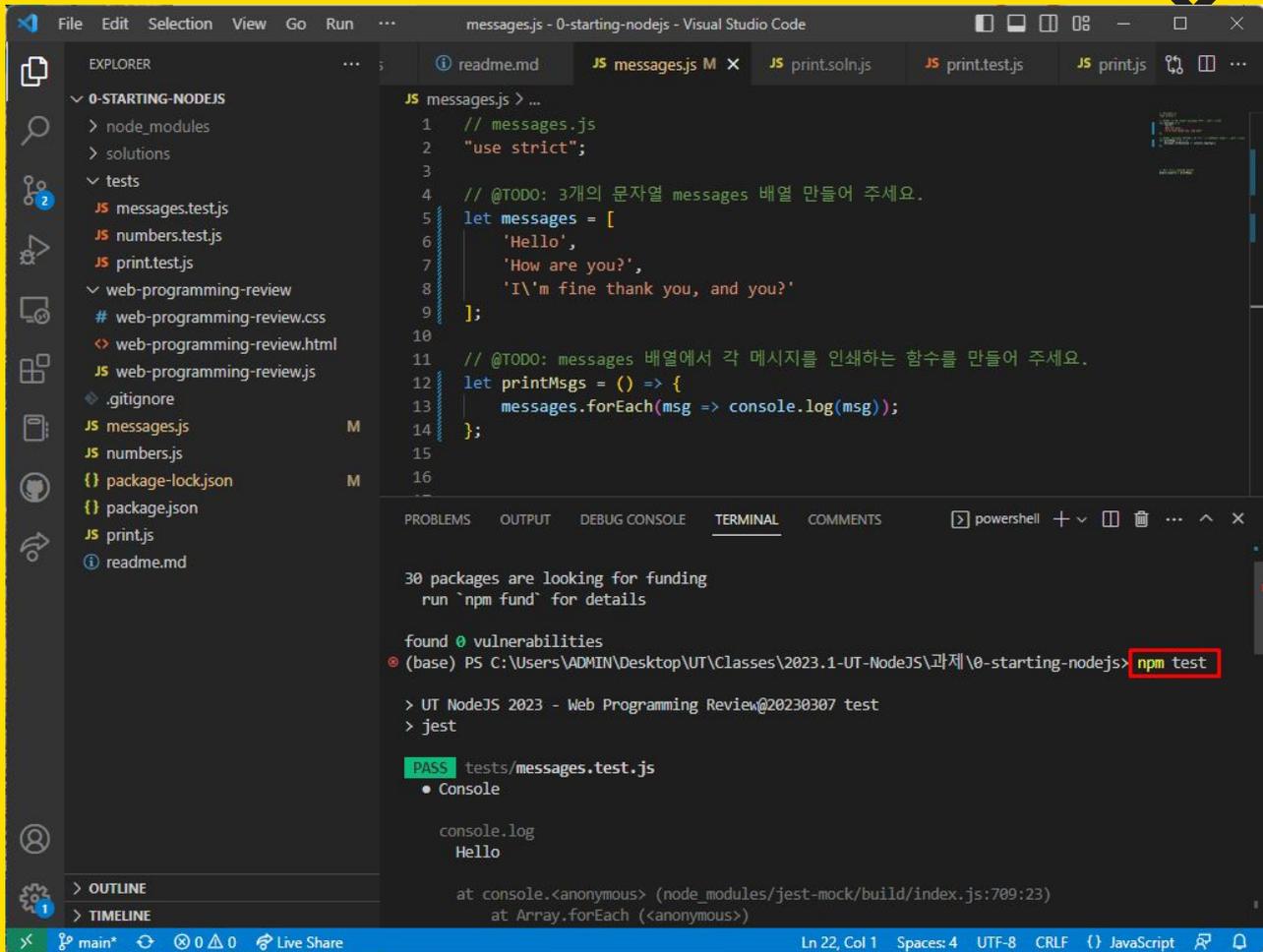
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
(base) PS C:\Users\ADMIN\Desktop\UT\Classes\2023.1-UT-NodeJS\과제\0-starting-nodejs> npm install
up to date, audited 278 packages in 712ms
30 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities
(base) PS C:\Users\ADMIN\Desktop\UT\Classes\2023.1-UT-NodeJS\과제\0-starting-nodejs>
```

2.

명령줄에서

npm install + npm test

1. npm install는 Jest 테스트 패키지 설치
2. **npm test**는 Jest 테스트 패키지 실행



```
File Edit Selection View Go Run ... messages.js - 0-starting-nodejs - Visual Studio Code
EXPLORER
0-STARTING-NODEJS
  > node_modules
  > solutions
  > tests
    JS messages.test.js
    JS numbers.test.js
    JS print.test.js
  > web-programming-review
    # web-programming-review.css
    <> web-programming-review.html
    JS web-programming-review.js
  > .gitignore
  JS messages.js M
  JS numbers.js
  {} package-lock.json M
  {} package.json
  JS print.js
  @ readme.md

JS messages.js > ...
1 // messages.js
2 "use strict";
3
4 // @TODO: 3개의 문자열 messages 배열 만들어 주세요.
5 let messages = [
6   'Hello',
7   'How are you?',
8   'I\'m fine thank you, and you?'
9 ];
10
11 // @TODO: messages 배열에서 각 메시지를 인쇄하는 함수를 만들어 주세요.
12 let printMsgs = () => {
13   messages.forEach(msg => console.log(msg));
14 };
15
16

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS powershell + v [] [] ... ^ x
30 packages are looking for funding
run `npm fund` for details

found 0 vulnerabilities
(base) PS C:\Users\ADMIN\Desktop\UT\Classes\2023.1-UT-NodeJS\과제\0-starting-nodejs> npm test

> UT NodeJS 2023 - Web Programming Review@20230307 test
> jest

PASS tests/messages.test.js
  • Console

  console.log
    Hello

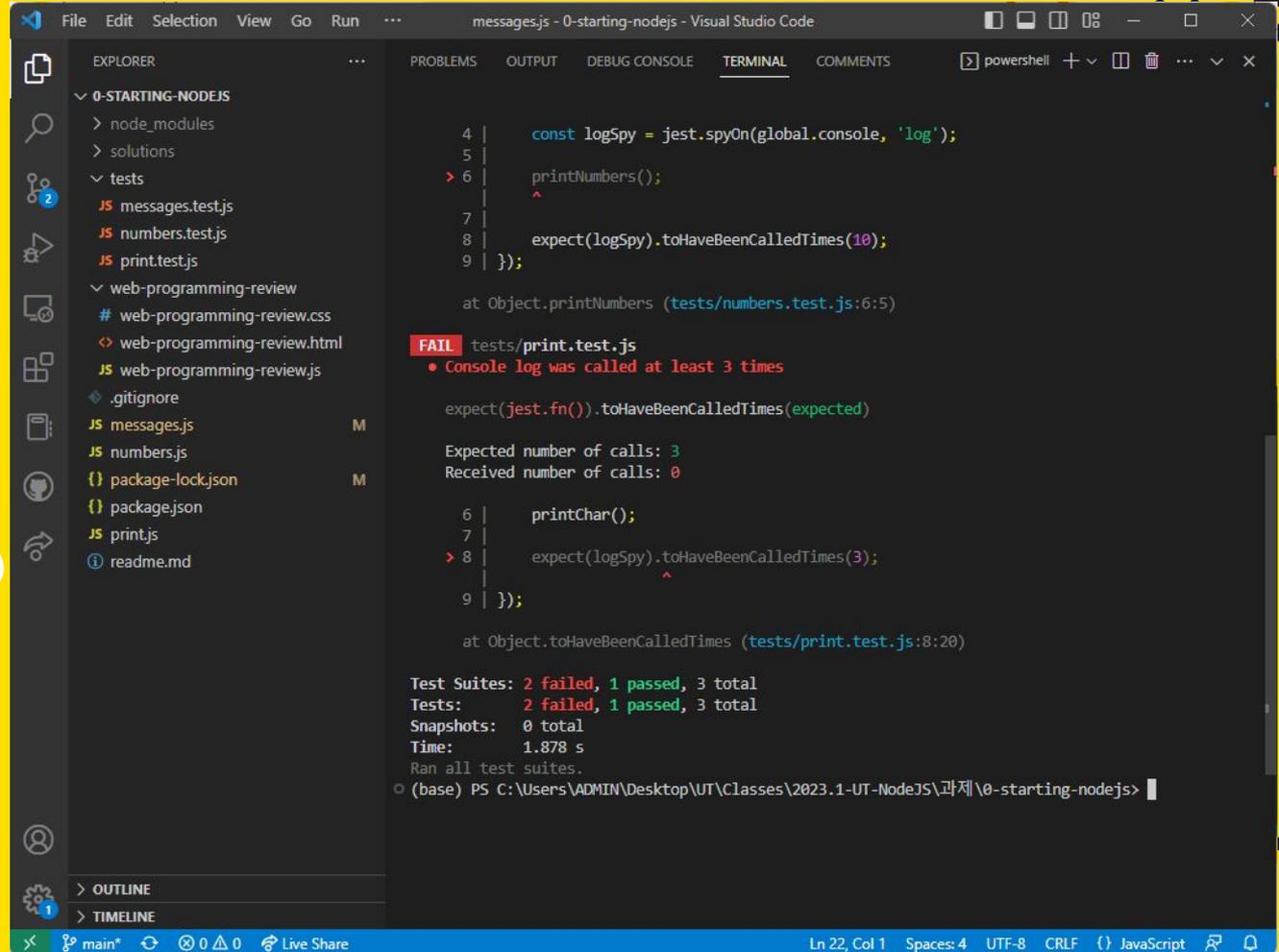
    at console.<anonymous> (node_modules/jest-mock/build/index.js:709:23)
    at Array.forEach (<anonymous>)
```


2.

명령줄에서

npm install + npm test

1. npm install는 Jest 테스트 패키지 설치
2. **npm test**는 Jest 테스트 패키지 실행



```
File Edit Selection View Go Run ... messages.js - 0-starting-nodejs - Visual Studio Code
EXPLORER PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
0-STARTING-NODEJS
  > node_modules
  > solutions
  > tests
    JS messages.test.js
    JS numbers.test.js
    JS print.test.js
  > web-programming-review
    # web-programming-review.css
    <> web-programming-review.html
    JS web-programming-review.js
    .gitignore
    JS messages.js M
    JS numbers.js M
    {} package-lock.json
    {} package.json
    JS print.js
    readme.md

4 |     const logSpy = jest.spyOn(global.console, 'log');
5 |
6 |     printNumbers();
7 |     ^
8 |     expect(logSpy).toHaveBeenCalledTimes(10);
9 |   });

at Object.printNumbers (tests/numbers.test.js:6:5)

FAIL tests/print.test.js
  • Console log was called at least 3 times

expect(jest.fn()).toHaveBeenCalledTimes(expected)

Expected number of calls: 3
Received number of calls: 0

6 |     printChar();
7 |
8 |     expect(logSpy).toHaveBeenCalledTimes(3);
9 |   });

at Object.toHaveBeenCalledTimes (tests/print.test.js:8:20)

Test Suites: 2 failed, 1 passed, 3 total
Tests:       2 failed, 1 passed, 3 total
Snapshots:  0 total
Time:        1.878 s
Ran all test suites.
(base) PS C:\Users\ADMIN\Desktop\UT\Classes\2023.1-UT-NodeJS\과제\0-starting-nodejs>
```



과제를 어떻게 확인할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



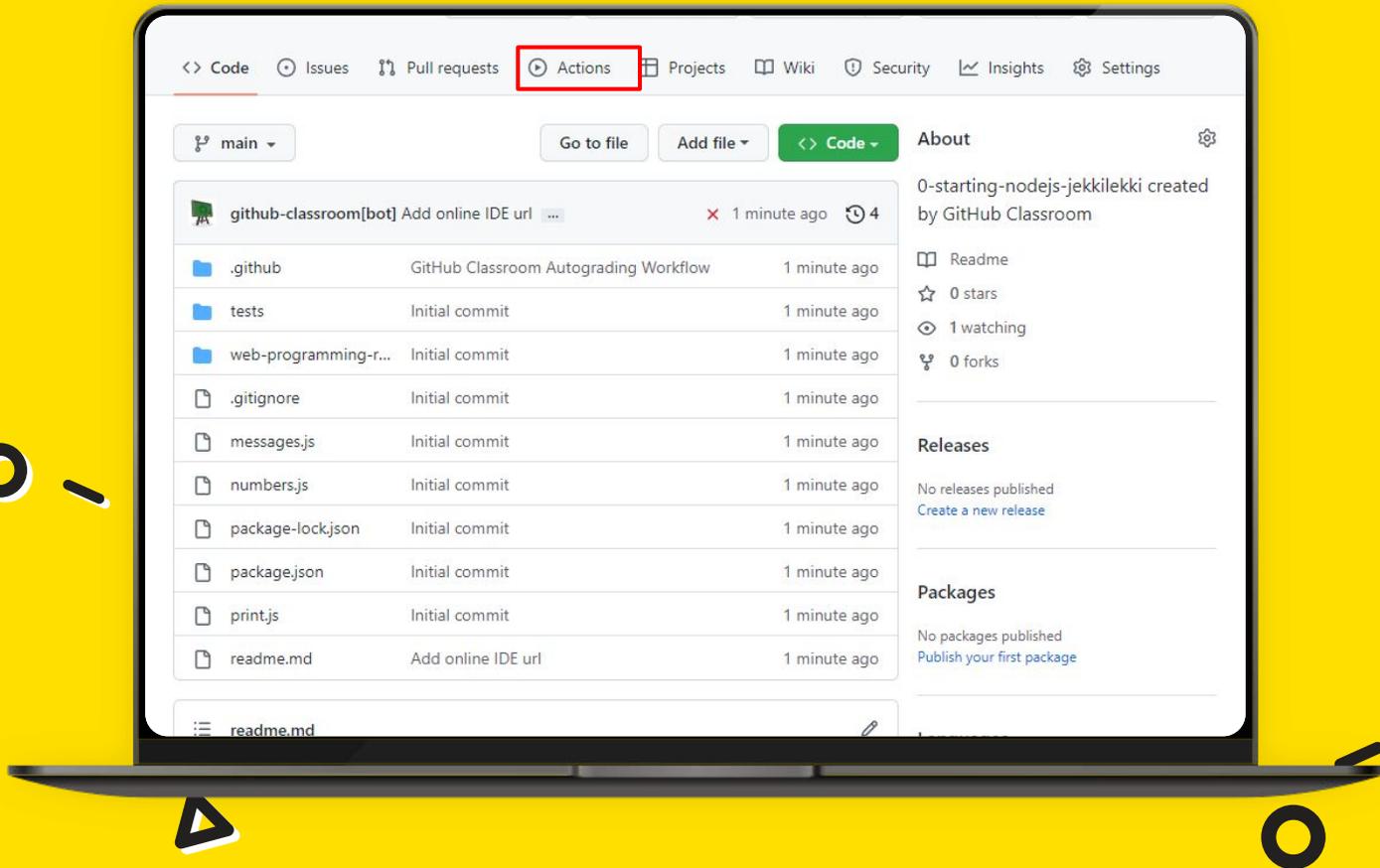
1. 로컬 컴퓨터 명령줄에서 코드 실행
2. 명령줄에서 **npm install + npm test**
3. 온라인 **GitHub Actions**에서

*** PASS/FAIL** 배지 추가 하면?

3.

온라인 **GitHub Actions**에서

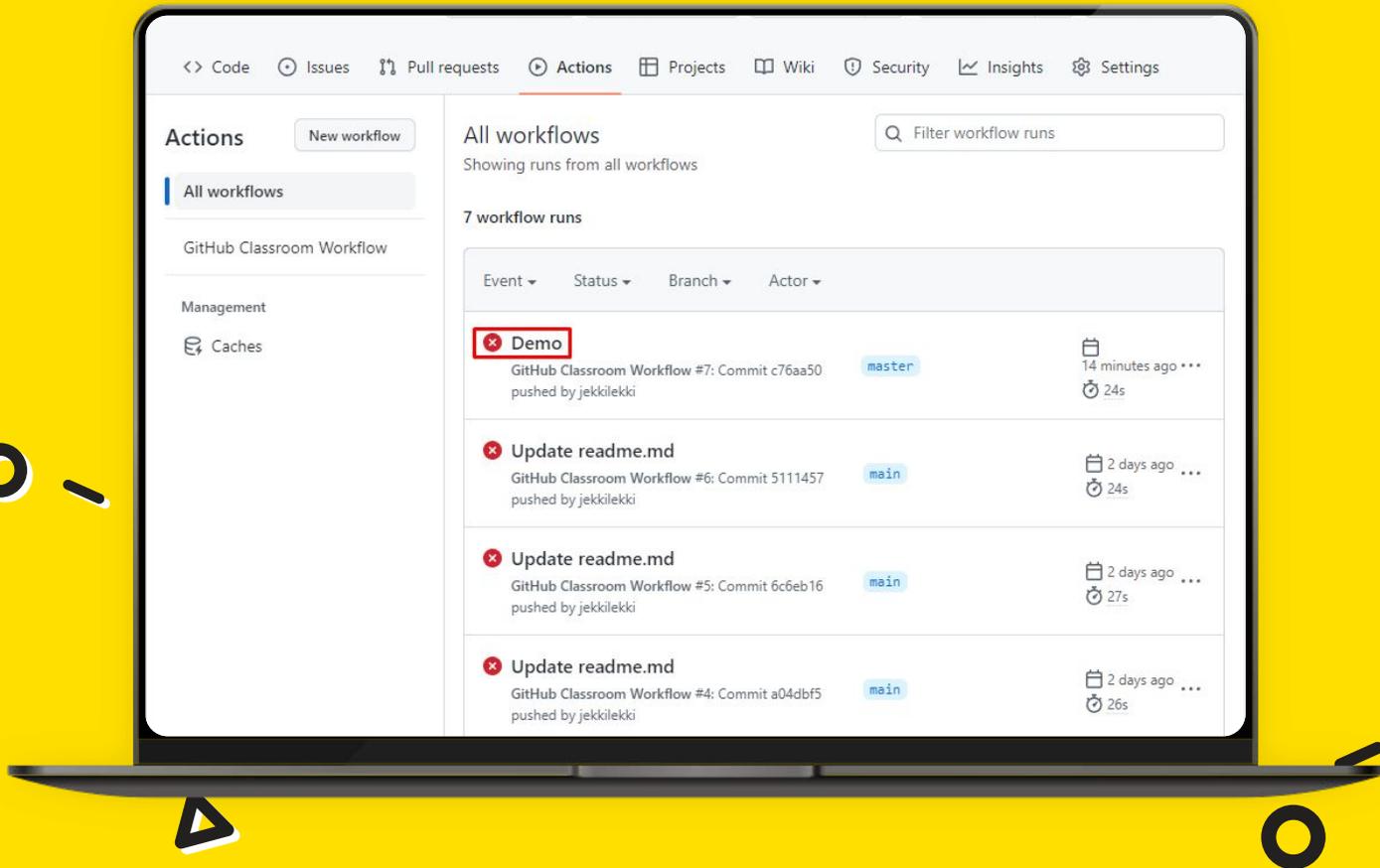
1. 온라인 저장소로 가고
2. “Actions” 탭 클릭
3. 최신 Workflow 클릭
4. “Autograding” 클릭
5. “Run autograding” 클릭
6. 테스트 확인



3.

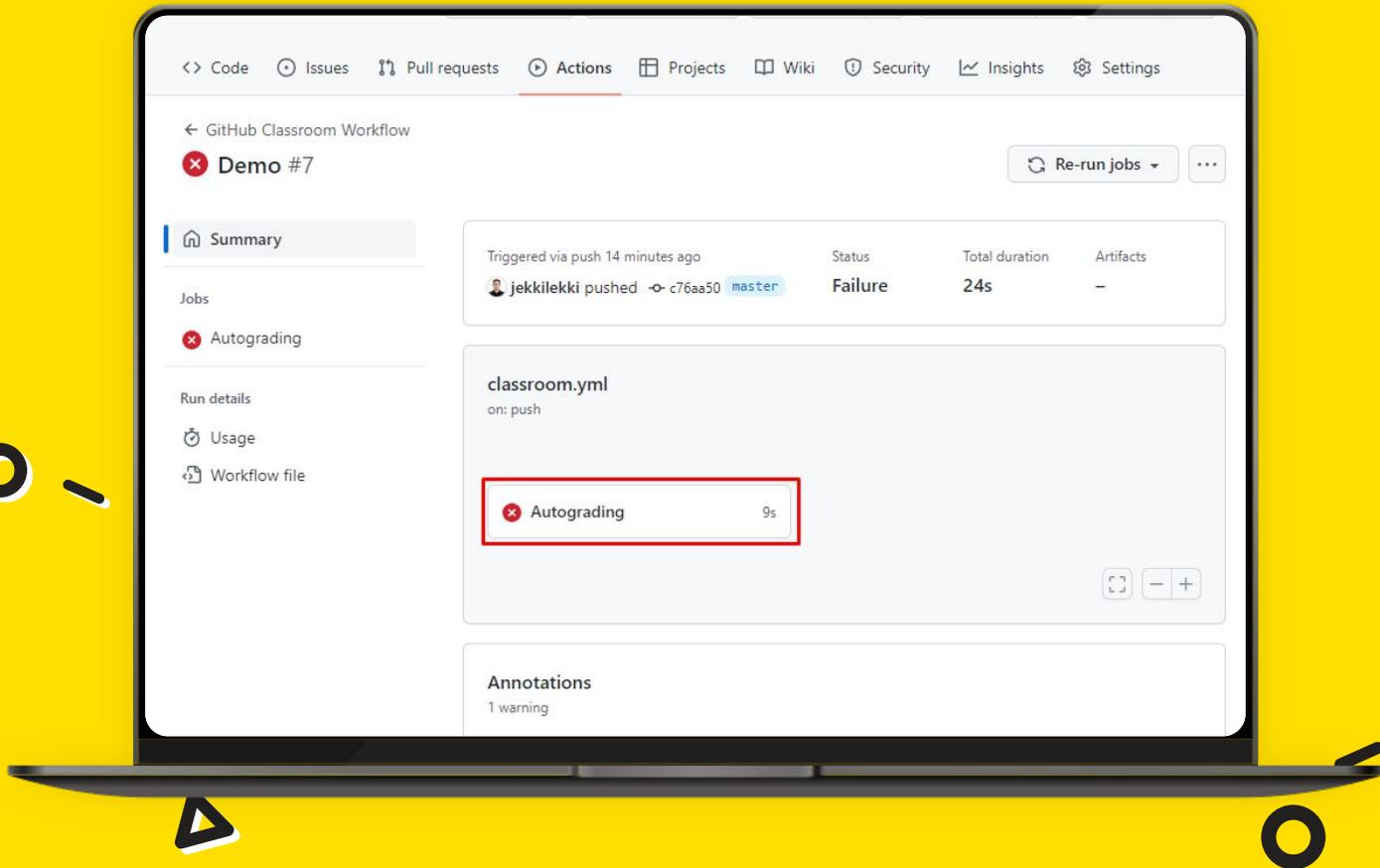
온라인 **GitHub Actions**에서

1. 온라인 저장소로 가고
2. "Actions" 탭 클릭
- 3. 최신 Workflow 클릭**
4. "Autograding" 클릭
5. "Run autograding" 클릭
6. 테스트 확인



3. 온라인 GitHub Actions에서

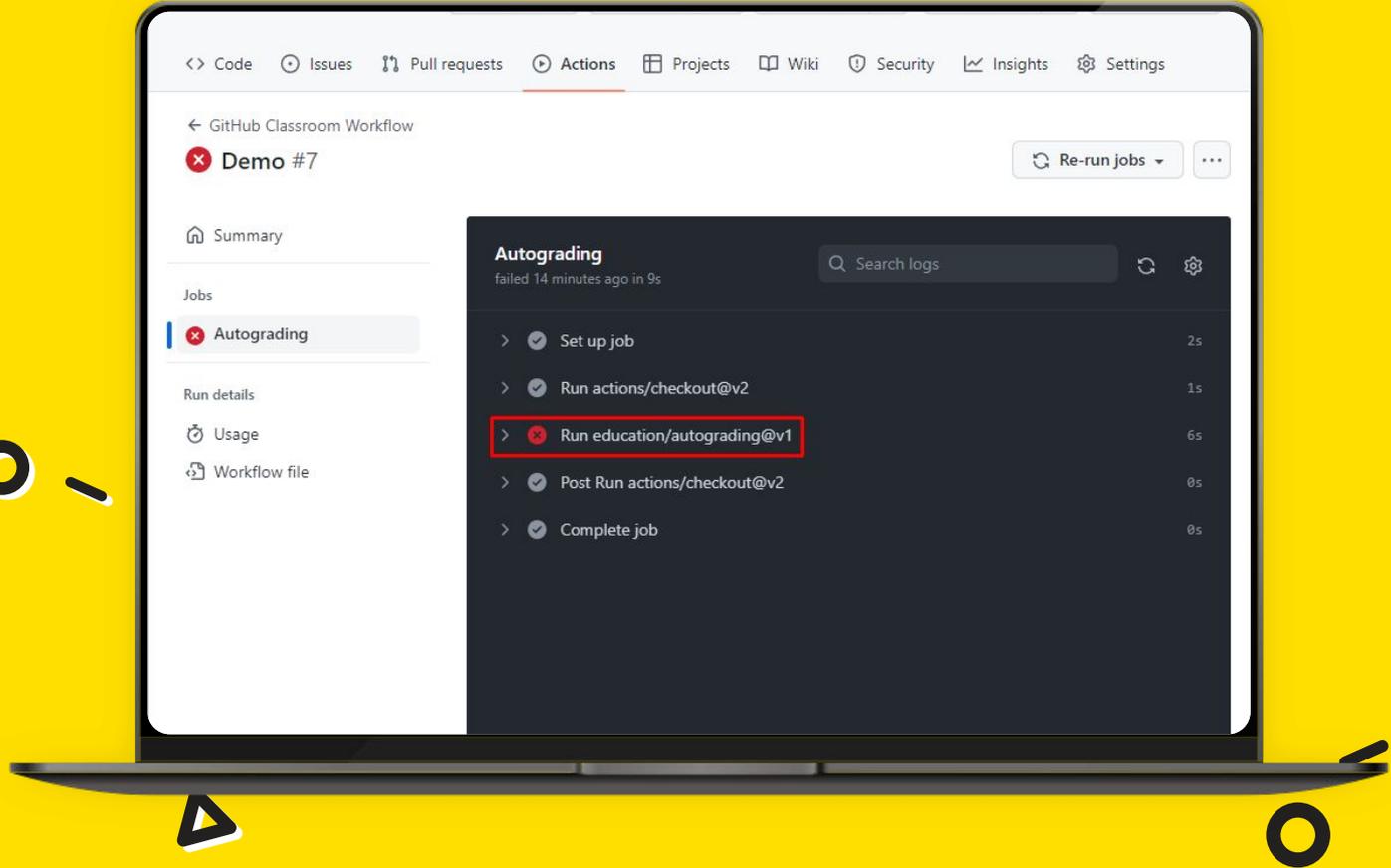
1. 온라인 저장소로 가고
2. "Actions" 탭 클릭
3. 최신 Workflow 클릭
4. "Autograding" 클릭
5. "Run autograding" 클릭
6. 테스트 확인



3.

온라인 **GitHub Actions**에서

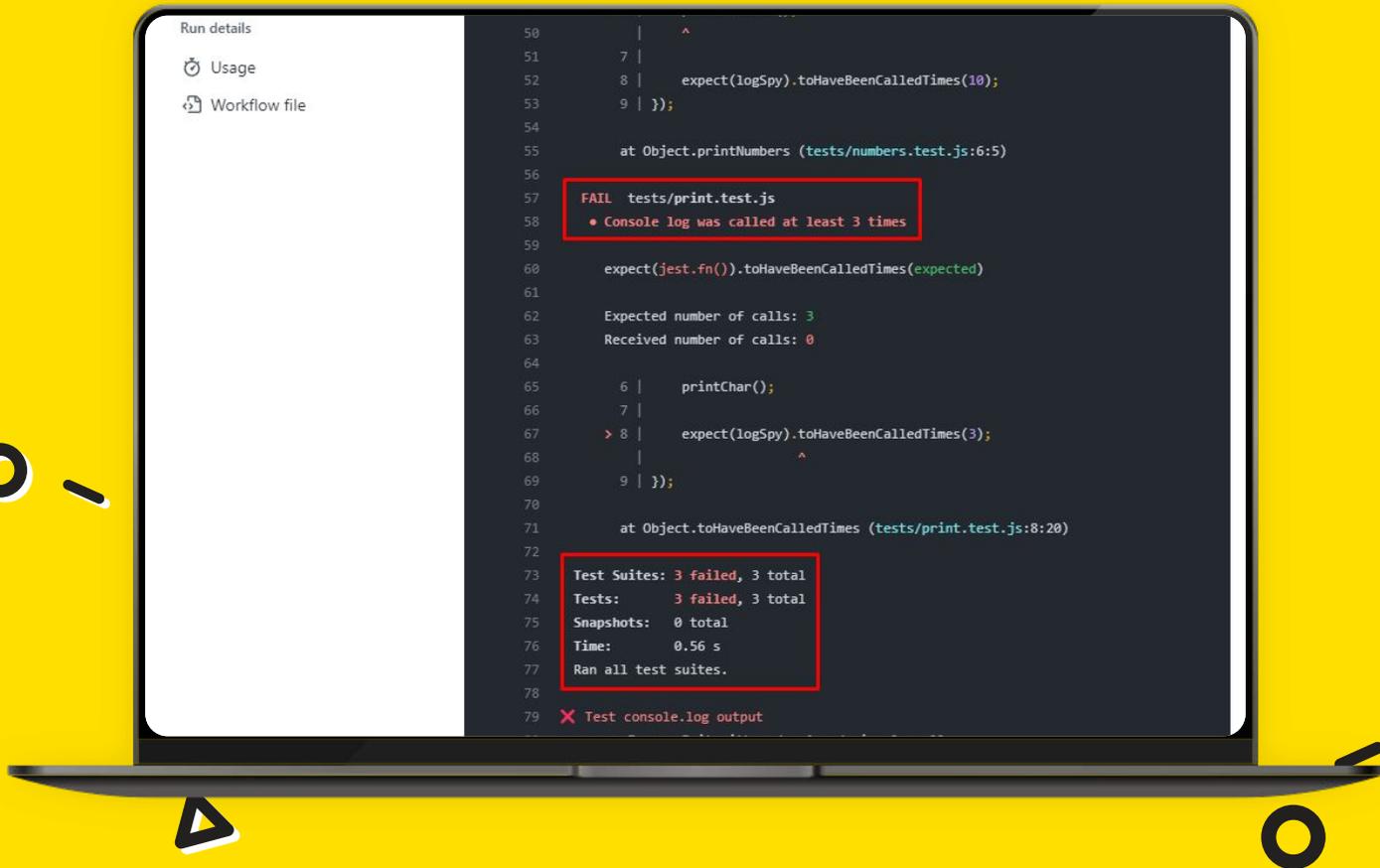
1. 온라인 저장소로 가고
2. "Actions" 탭 클릭
3. 최신 Workflow 클릭
4. "Autograding" 클릭
5. **"Run autograding"** 클릭
6. 테스트 확인



3.

온라인 **GitHub Actions**에서

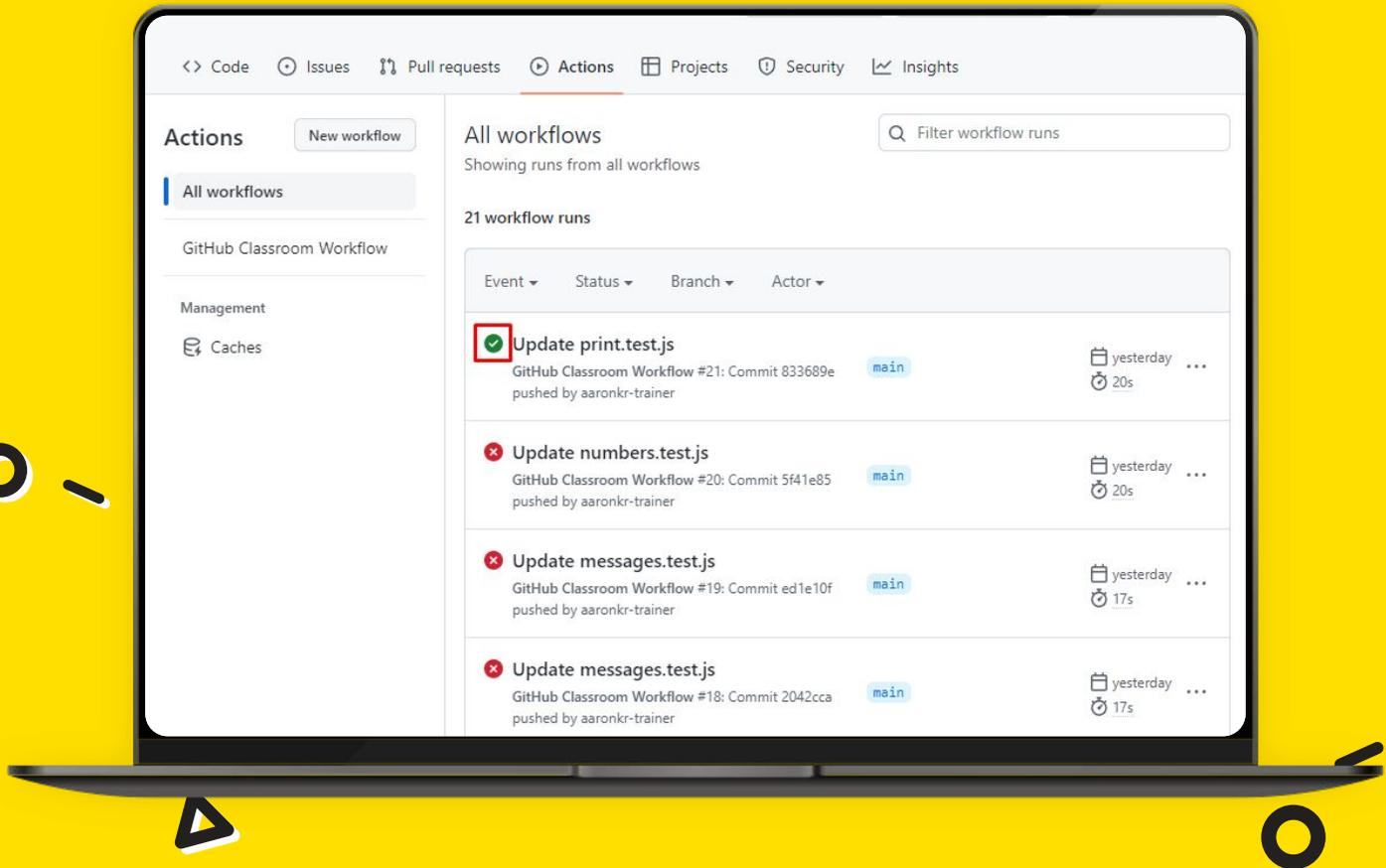
1. 온라인 저장소로 가고
2. "Actions" 탭 클릭
3. 최신 Workflow 클릭
4. "Autograding" 클릭
5. "Run autograding" 클릭
6. 테스트 확인



3.

온라인 **GitHub Actions**에서

1. 온라인 저장소로 가고
2. “Actions” 탭 클릭
3. 최신 Workflow 클릭
4. “Autograding” 클릭
5. “Run autograding” 클릭
6. 테스트 확인





과제를 어떻게 확인할 수 있나요?

문제를 생기면? 다른 방법으로 해 보세요.



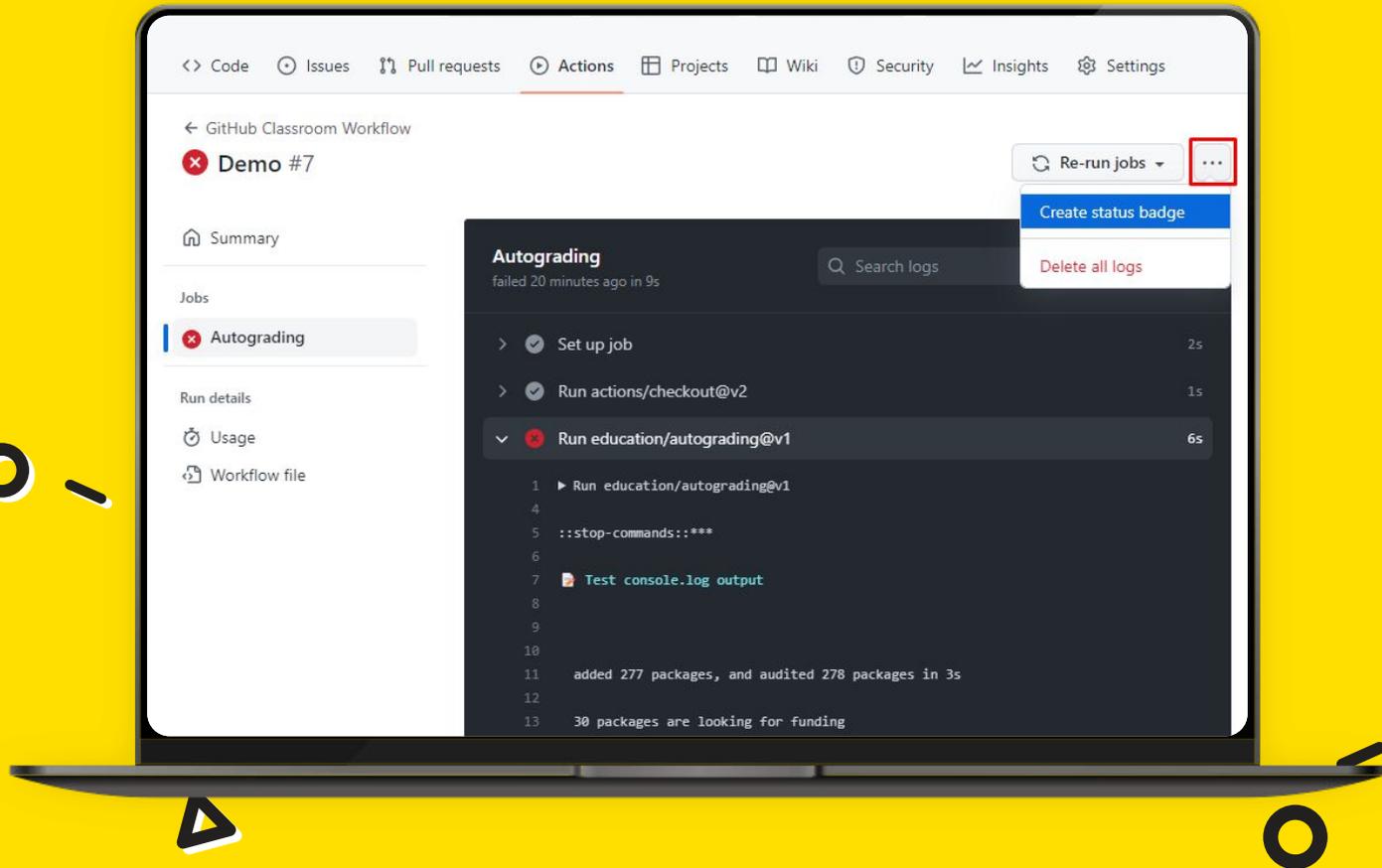
1. 로컬 컴퓨터 명령줄에서 코드 실행
2. 명령줄에서 **npm install + npm test**
3. 온라인 **GitHub Actions**에서

* **PASS/FAIL** 배지 추가 하면?

PASS/FAIL

배지 추가 하면

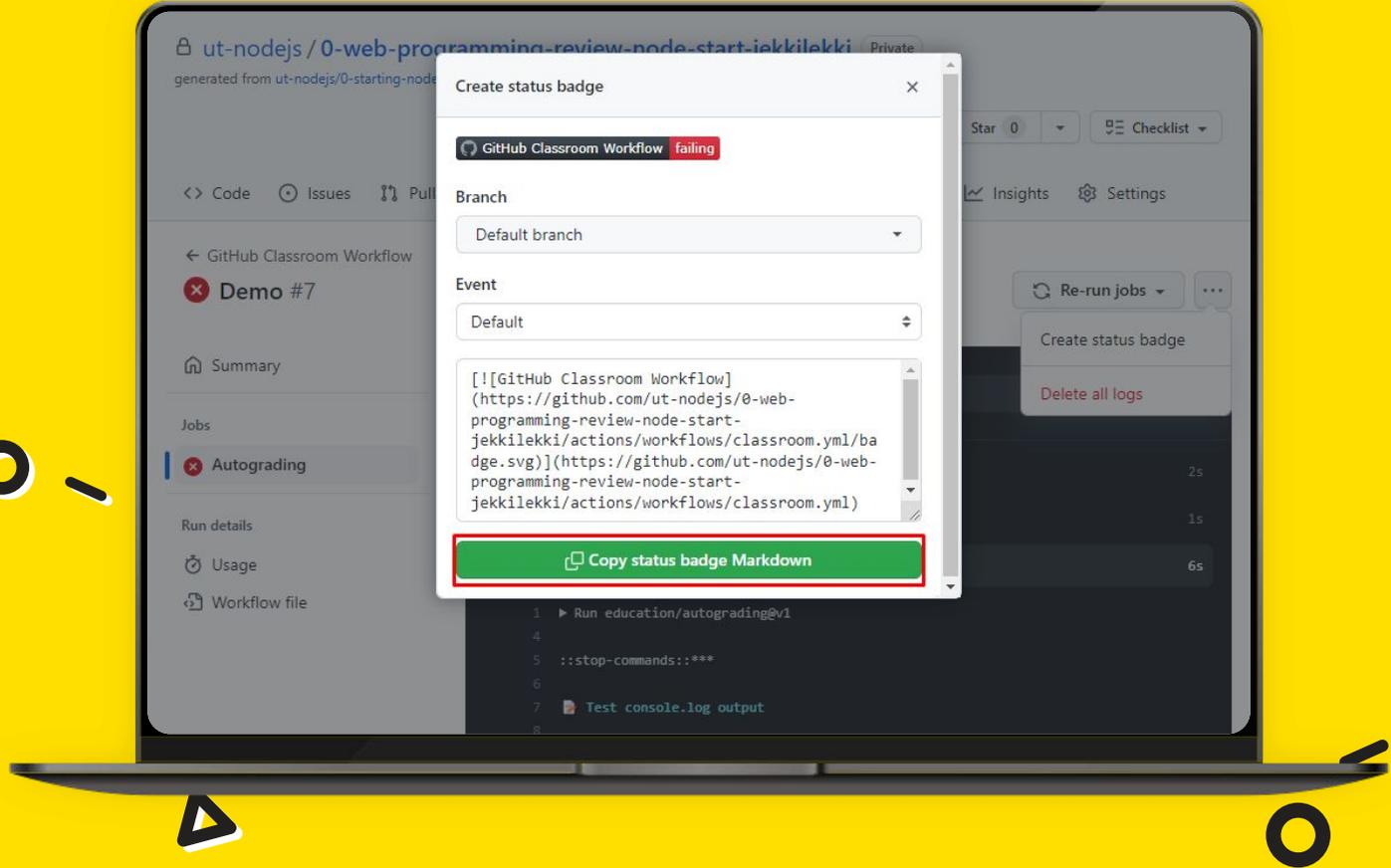
1. **Autograding**에서
“...” ->
2. **Create status badge**
Markdown 코드 복사
3. 저장소에서
readme.md 편집
4. Markdown 코드 붙여넣기



PASS/FAIL

배지 추가 하면

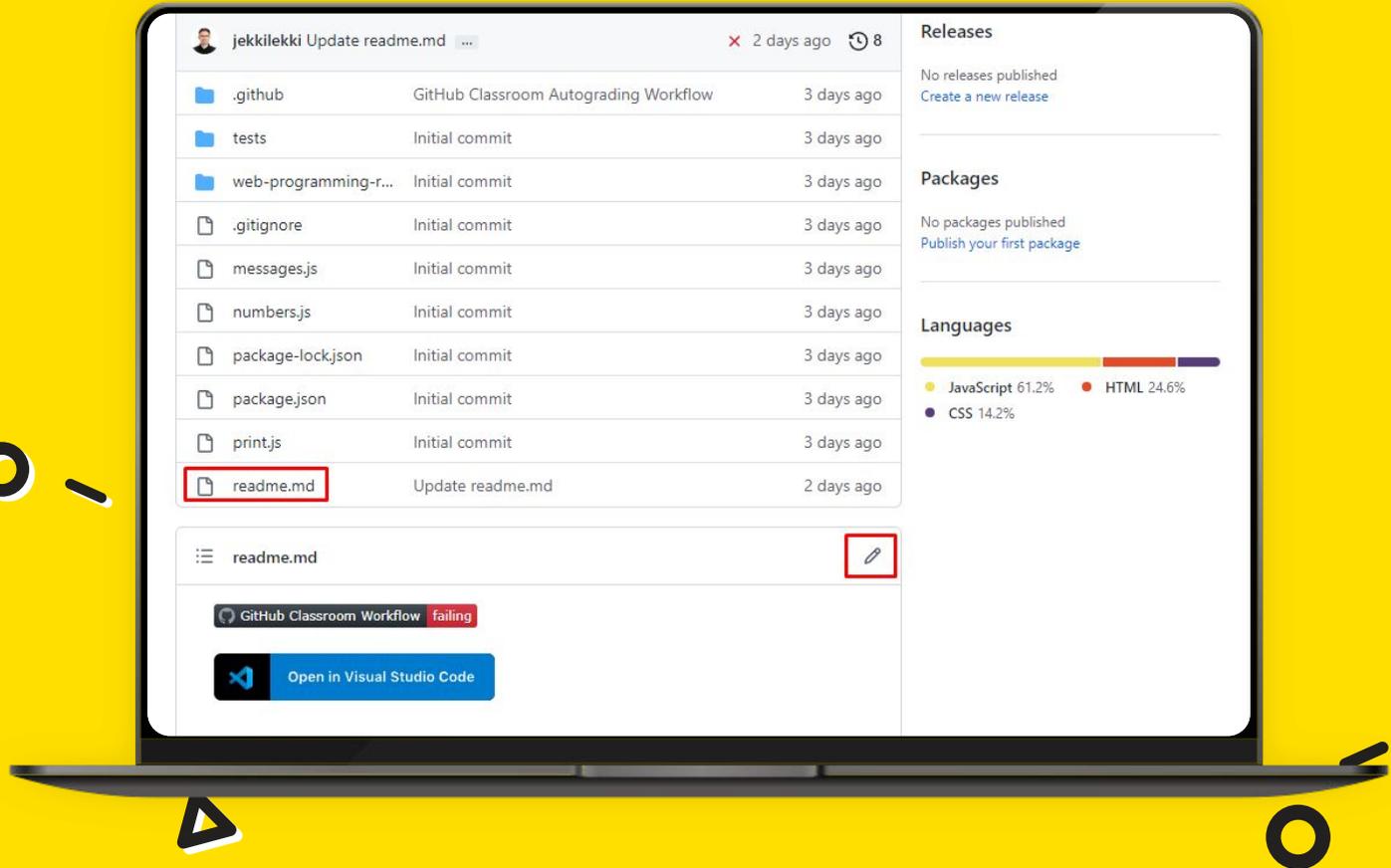
1. Autograding에서
"..." ->
2. Create status badge
3. **Markdown 코드 복사**
4. 저장소에서
readme.md 편집
5. Markdown 코드 부쳐넣기
- 6.



PASS/FAIL

배지 추가 하면

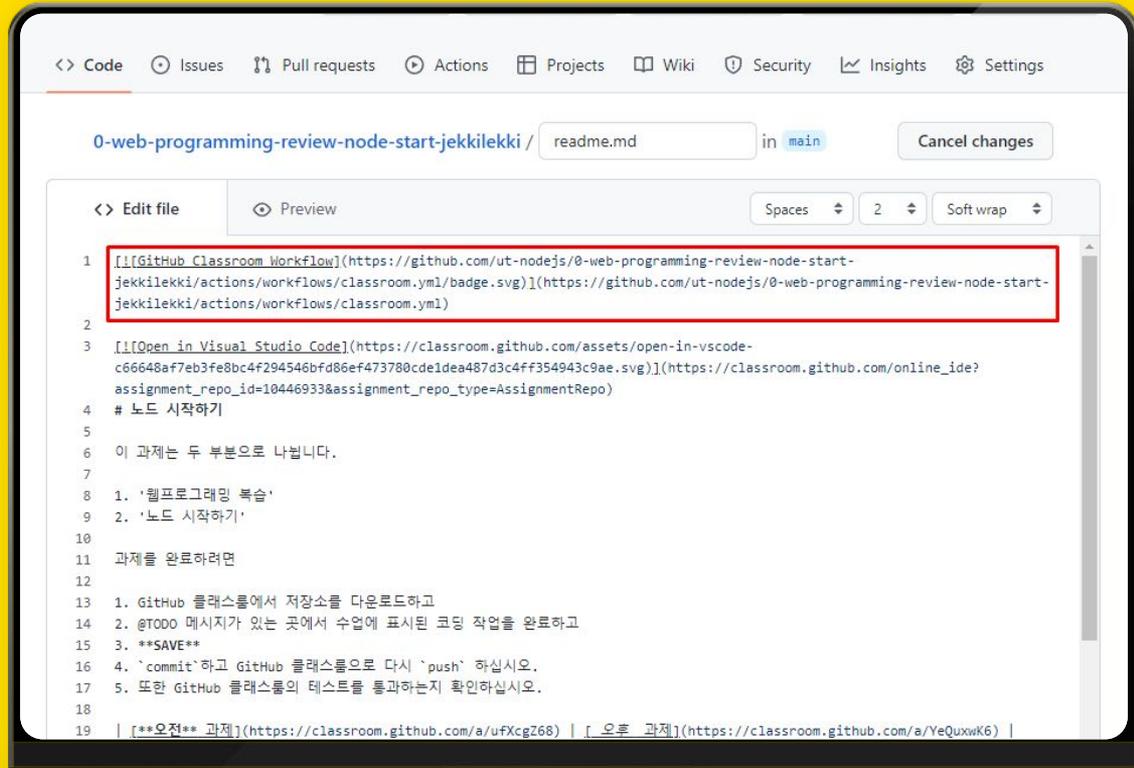
1. Autograding에서
"..." ->
2. Create status badge
Markdown 코드 복사
- 3.
4. 저장소에서
5. **readme.md** 편집
6. Markdown 코드 부쳐넣기



PASS/FAIL

배지 추가 하면

1. Autograding에서
"..." ->
2. Create status badge
3. Markdown 코드 복사
4. 저장소에서
readme.md 편집
5. **Markdown 코드**
부쳐넣기



The screenshot shows a GitHub repository page for '0-web-programming-review-node-start-jekkilekki'. The file 'readme.md' is open in the 'main' branch. The code editor shows the following content:

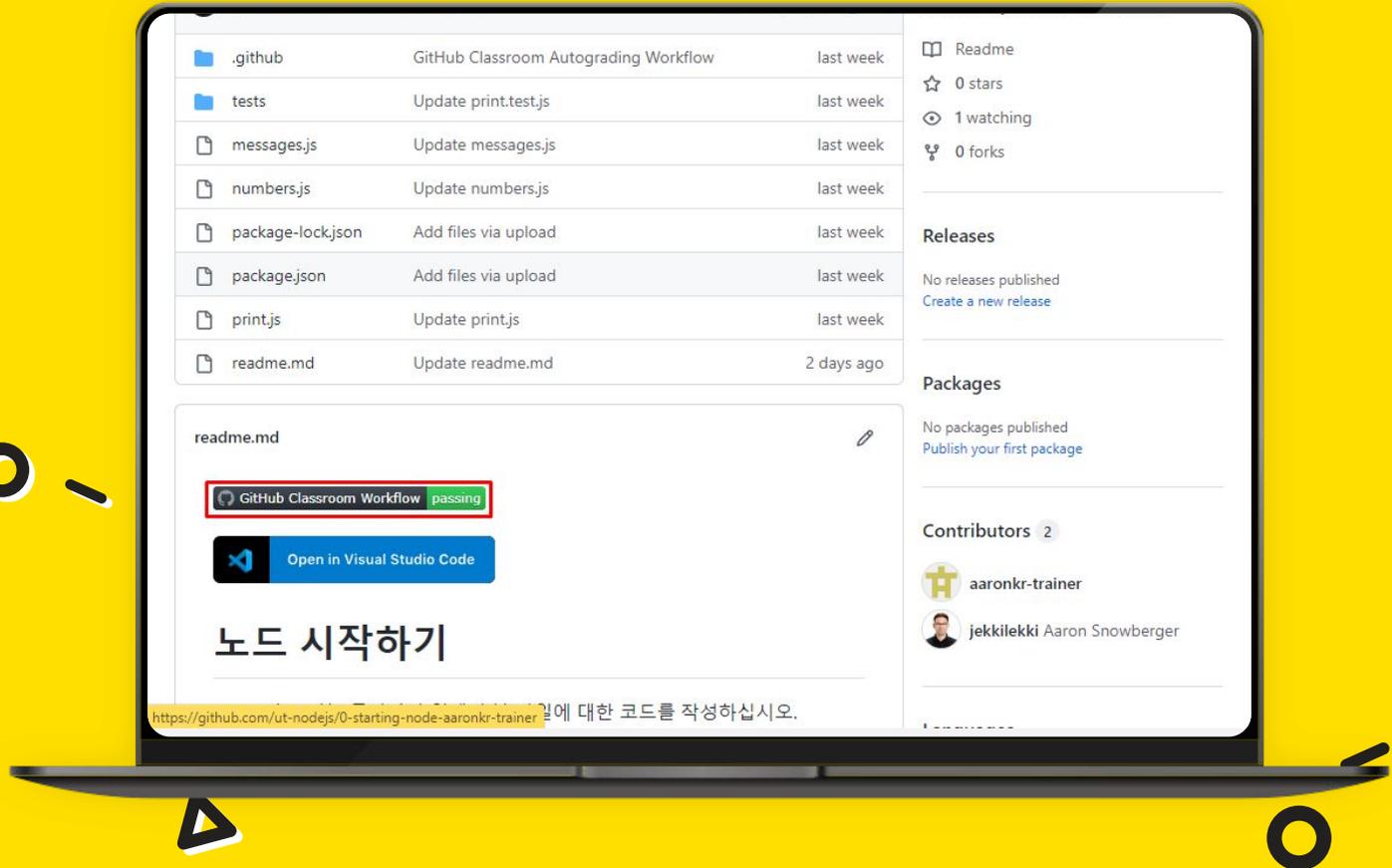
```
1 [[GitHub Classroom Workflow]](https://github.com/ut-nodejs/0-web-programming-review-node-start-jekkilekki/actions/workflows/classroom.yml/badge.svg)[](https://github.com/ut-nodejs/0-web-programming-review-node-start-jekkilekki/actions/workflows/classroom.yml)
```

The badge code is highlighted with a red box. The rest of the file contains Korean text and instructions for the assignment.

PASS/FAIL

배지 추가 하면

1. Autograding에서
"..." ->
2. Create status badge
3. Markdown 코드 복사
4. 저장소에서
readme.md 편집
5. **Markdown** 코드
부쳐넣기
- 6.



Questions?

한번 해 보자!~