

Eclipse에서 github 사용

동의과학대학교 컴퓨터정보과
김진숙

내용

1. 원격 저장소 생성
2. 비밀번호를 위한 token 생성

github.com에서 진행

3. 로컬 저장소 생성
4. 원격 저장소와 연결 설정
 - fetch로 먼저 원격에서 가져오기
 - push로 로컬 파일을 원격으로 보내기
5. 로컬 저장소 디폴트 브랜치 변경

eclipse에서 진행

[필요시] .gitignore 파일 변경

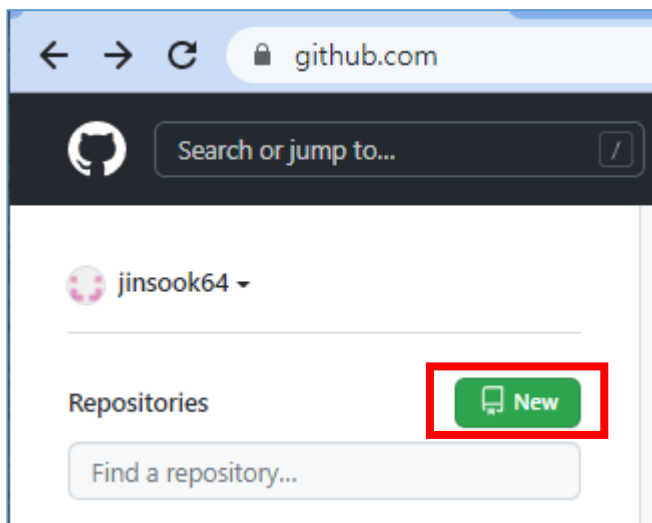
- 원격 저장소를 생성할 때 템플릿을 java로 선택했으면 필요없는 부분

설정 순서

1. github에 repository를 생성한다.
2. github에서 비밀번호로 사용할 token을 생성한다.
3. 이클립스에서 프로젝트를 생성한다.
4. 프로젝트를 git으로 관리하도록 로컬 repository로 만든다.
5. github에 만들어진 repository와 이클립스 내 로컬 repository를 연결하기 위해 remote origi을 생성한다.
6. 우선 github의 내용을 fetch해서 로컬로 가져온다.
7. 원격 github로 올릴 파일을 선택하여 git stage로 올린다.
8. commit 메시지를 작성한 후 push한다.
9. 로컬 repository의 디폴트 브랜치명을 변경한다.

1. 원격 저장소 생성

- <http://github.com> 에서 로그인하여 새로운 repository 생성



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * jinsook64

Repository name *

Great repository names are short and memorable. Need inspiration? How about [didactic-guacamole?](#)

Description (optional)

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

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

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

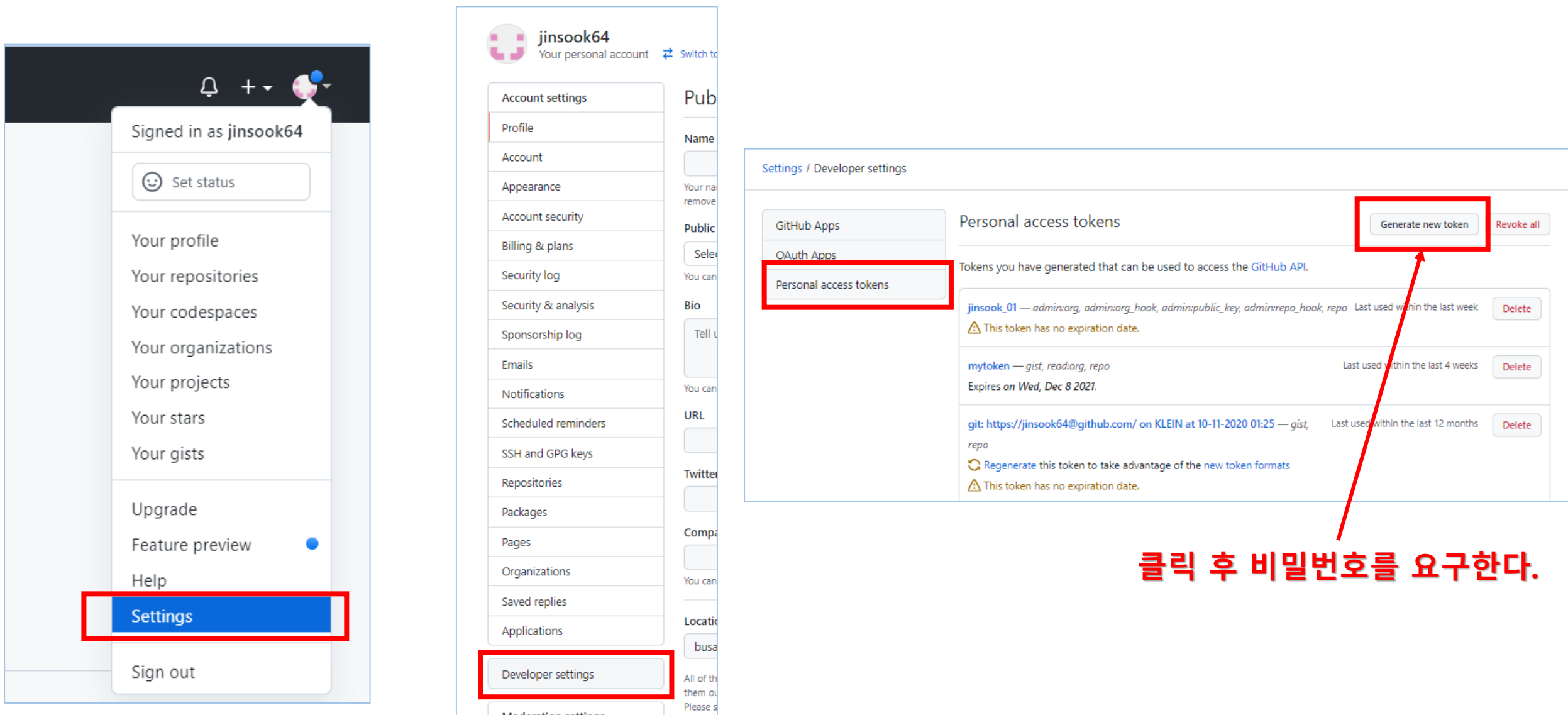
☐ Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository

저장소명을 입력한다.

2. 비밀번호를 위한 token 생성



The image shows three sequential steps to generate a GitHub Personal Access Token:

- Step 1:** A screenshot of the GitHub mobile app's user menu. The 'Settings' option at the bottom is highlighted with a red box.
- Step 2:** A screenshot of the GitHub web interface's 'Developer settings' page. The 'Personal access tokens' option in the left sidebar is highlighted with a red box.
- Step 3:** A screenshot of the 'Personal access tokens' page. The 'Generate new token' button in the top right corner is highlighted with a red box. A red arrow points from this button to the text '클릭 후 비밀번호를 요구한다.' (Click to request password).

2. 비밀번호를 위한 token 생성

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

jskim_token

What's this token for?

Expiration *

No expiration ↕

The token will never expire!

GitHub strongly recommends that you set an expiration date for your token to help keep your information secure. [Learn more](#)

2. 비밀번호를 위한 token 생성

- 다음은 사항은 임의로 체크한 것이므로 자세한 것은 더 알아 볼 것
- repo 부분은 반드시 체크할 것

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry
<input checked="" type="checkbox"/> admin:org	Full control of orgs and teams, read and write org projects
<input checked="" type="checkbox"/> write:org	Read and write org and team membership, read and write org projects
<input checked="" type="checkbox"/> read:org	Read org and team membership, read org projects
<input type="checkbox"/> admin:public_key	Full control of user public keys
<input type="checkbox"/> write:public_key	Write user public keys
<input type="checkbox"/> read:public_key	Read user public keys
<input type="checkbox"/> admin:repo_hook	Full control of repository hooks
<input type="checkbox"/> write:repo_hook	Write repository hooks
<input type="checkbox"/> read:repo_hook	Read repository hooks
<input type="checkbox"/> admin:org_hook	Full control of organization hooks
<input type="checkbox"/> gist	Create gists
<input type="checkbox"/> notifications	Access notifications

<input checked="" type="checkbox"/> user	Update ALL user data
<input checked="" type="checkbox"/> read:user	Read ALL user profile data
<input checked="" type="checkbox"/> user:email	Access user email addresses (read-only)
<input checked="" type="checkbox"/> user:follow	Follow and unfollow users
<input type="checkbox"/> delete_repo	Delete repositories
<input checked="" type="checkbox"/> write:discussion	Read and write team discussions
<input checked="" type="checkbox"/> read:discussion	Read team discussions
<input checked="" type="checkbox"/> admin:enterprise	Full control of enterprises
<input checked="" type="checkbox"/> manage_runners:enterprise	Manage enterprise runners and runner-groups
<input checked="" type="checkbox"/> manage_billing:enterprise	Read and write enterprise billing data
<input checked="" type="checkbox"/> read:enterprise	Read enterprise profile data
<input checked="" type="checkbox"/> admin:gpg_key	Full control of public user GPG keys (Developer Preview)
<input checked="" type="checkbox"/> write:gpg_key	Write public user GPG keys
<input checked="" type="checkbox"/> read:gpg_key	Read public user GPG keys

Generate token

Cancel

2. 비밀번호를 위한 token 생성

Settings / Developer settings

GitHub Apps
OAuth Apps
Personal access tokens

Personal access tokens

Generate new token Revoke all

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

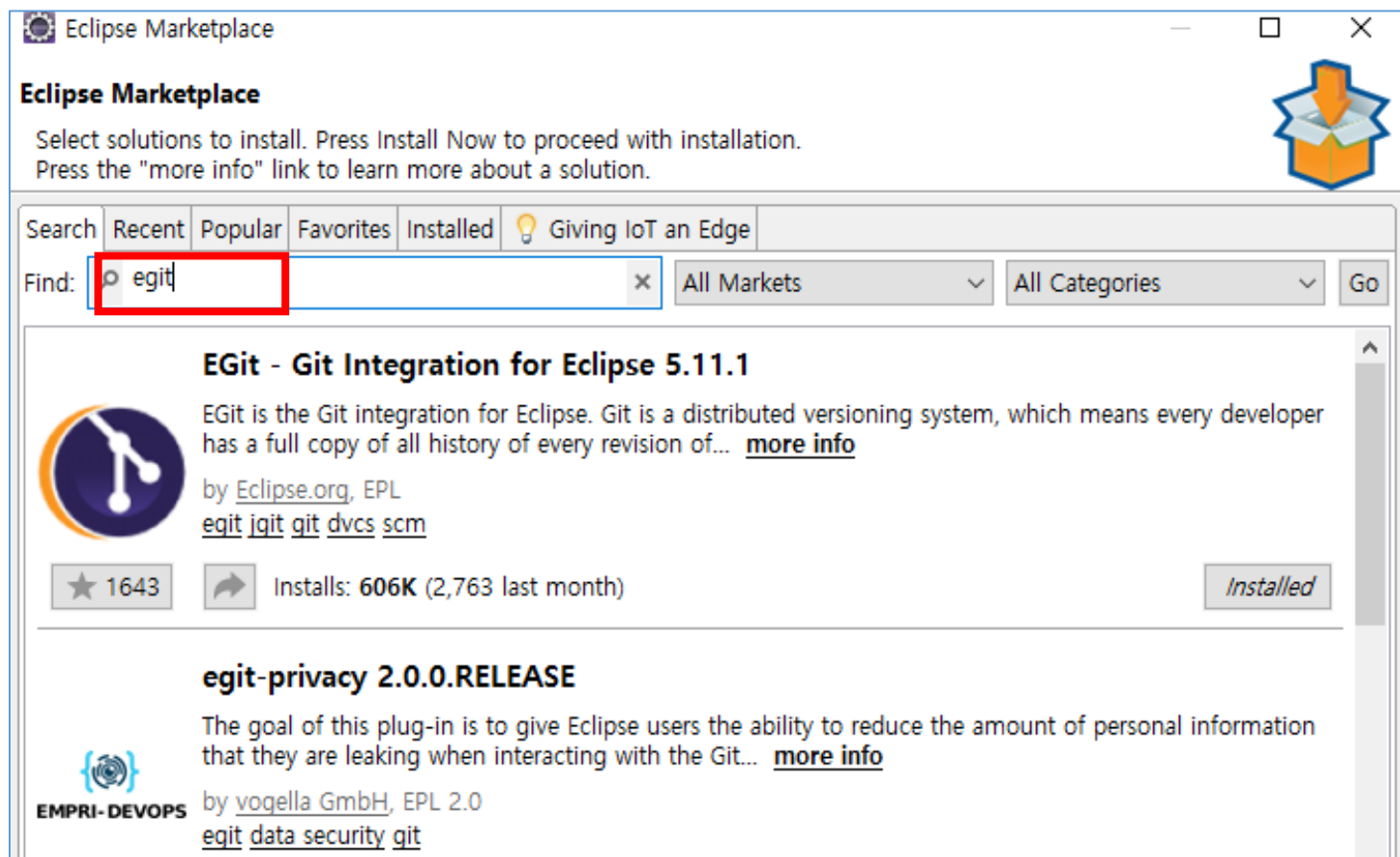
✓	[Redacted token]	Delete
jinsook_01 — admin:org, admin:org_hook, admin:public_key, admin:repo_hook, repo Last used within the last week ⚠ This token has no expiration date. Delete		
mytoken — gist, read:org, repo Last used within the last 4 weeks Expires on Wed, Dec 8 2021. Delete		
git: https://jinsook64@github.com/ on KLEIN at 10-11-2020 01:25 — gist, repo Last used within the last 12 months 🔄 Regenerate this token to take advantage of the new token formats ⚠ This token has no expiration date. Delete		

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

토큰은 반드시 캡처해 두거나 텍스트로 저장해야 함!!
로컬 저장소에 txt 파일로 저장해 둘 것!!

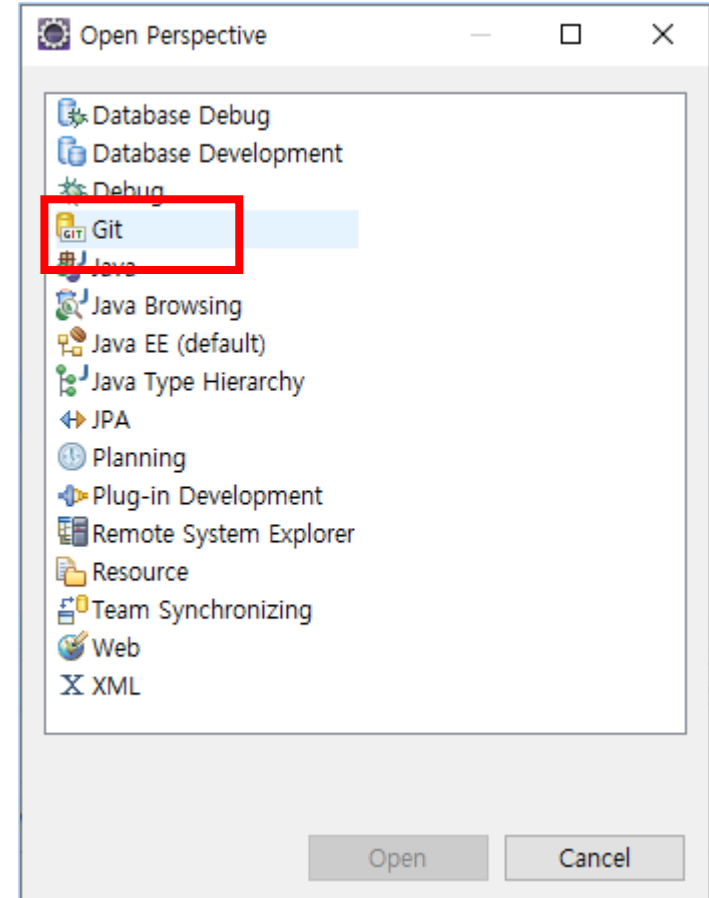
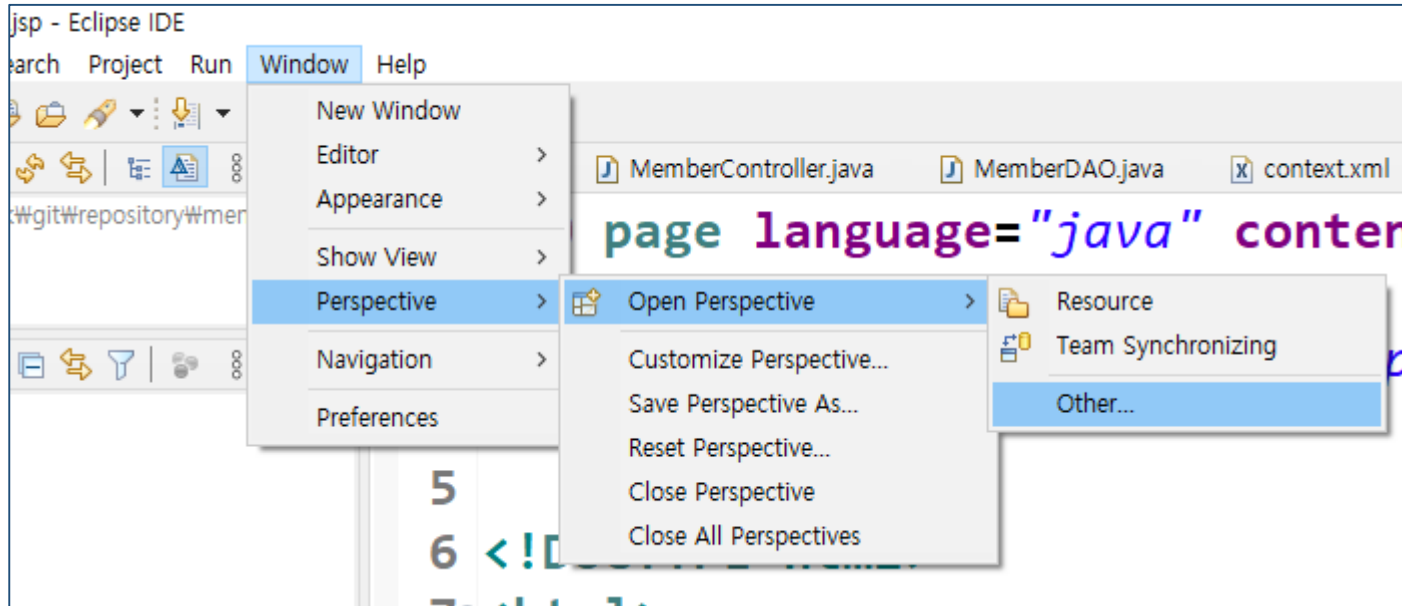
[필요시] EGit 설치(현재 버전에서는 불필요)

- 설치되어 있는 경우가 대부분
- Help – Eclipse Marketplace
- Egit 검색 - 설치

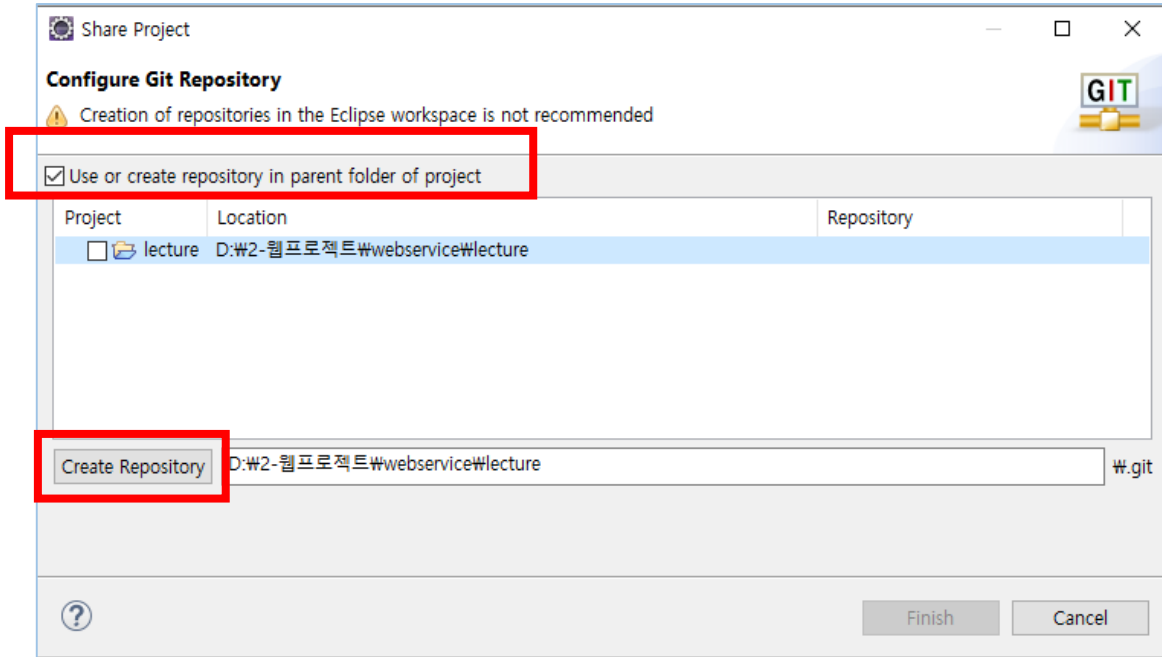
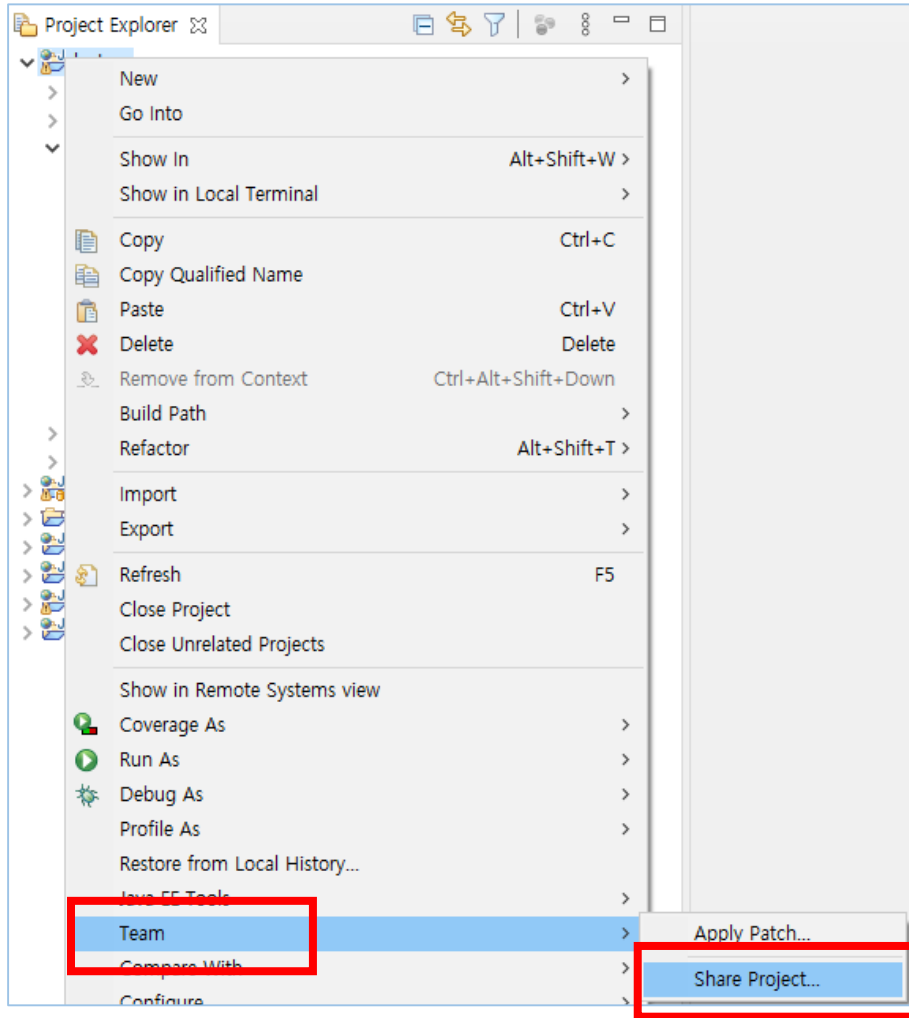


[필요시] EGit 설치(필요 시)

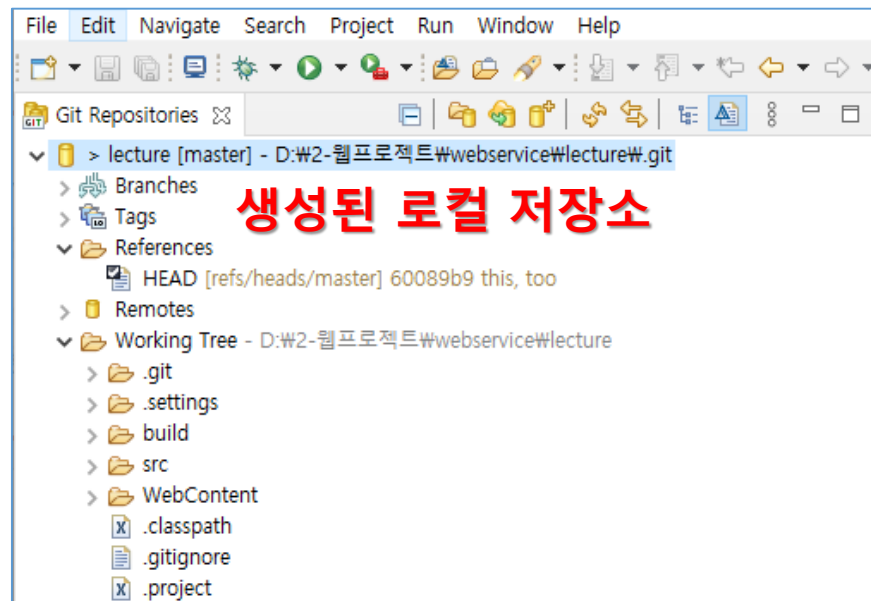
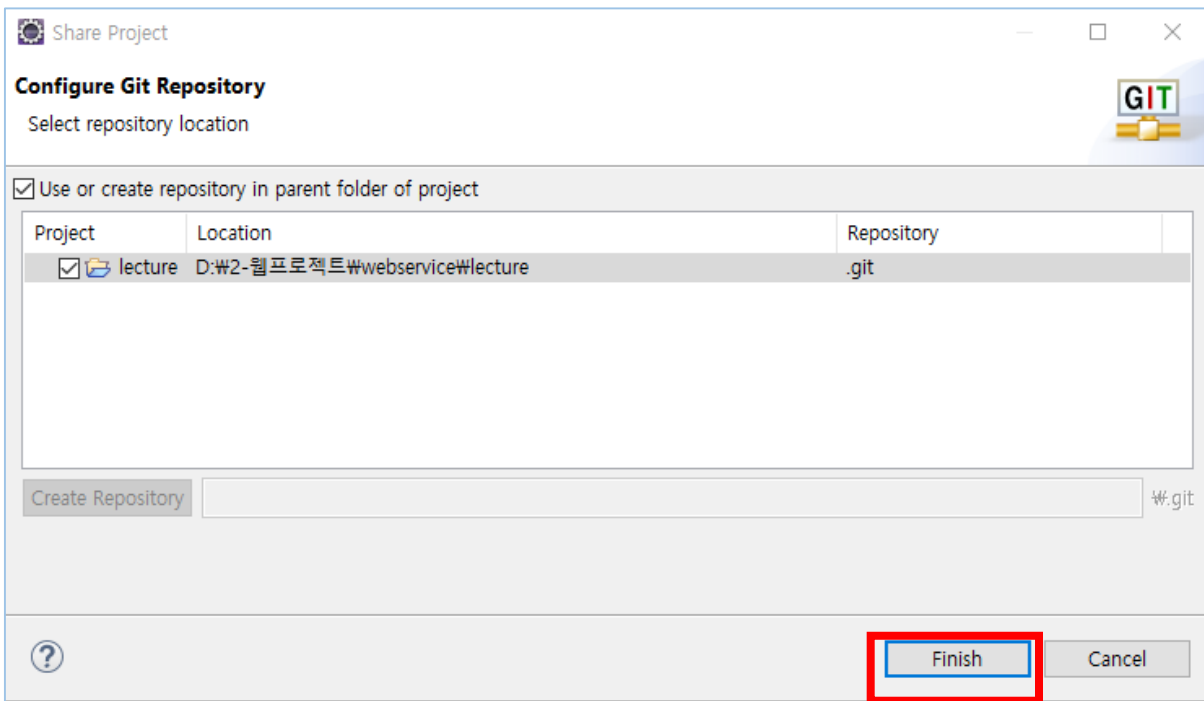
- Window – Perspective – Open Perspective – Other – Git - Open



3. 로컬 저장소 생성

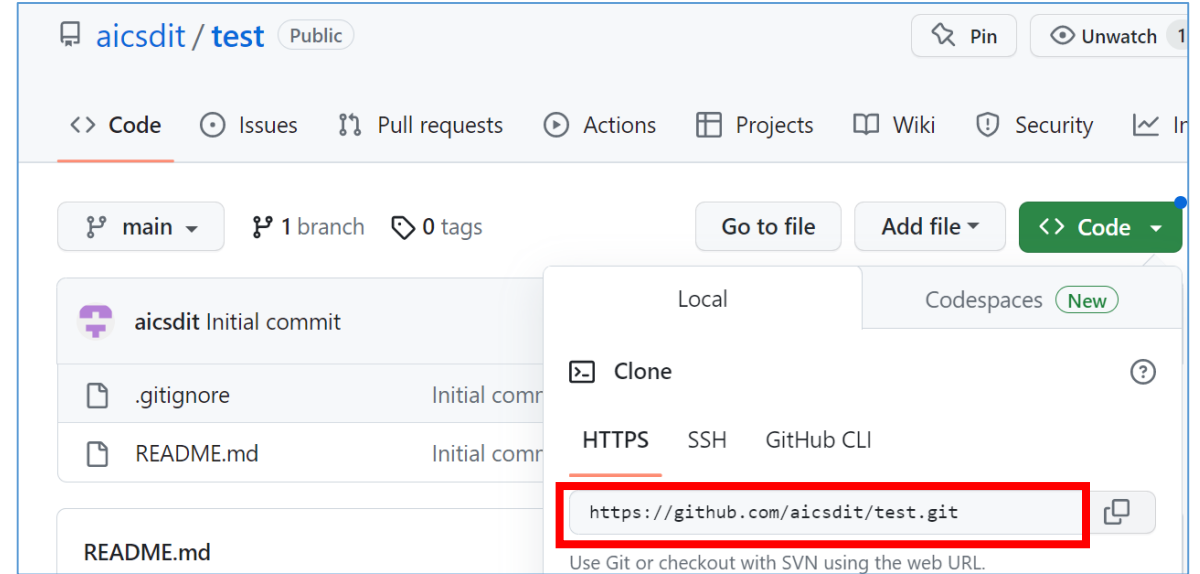


3. 로컬 저장소 생성

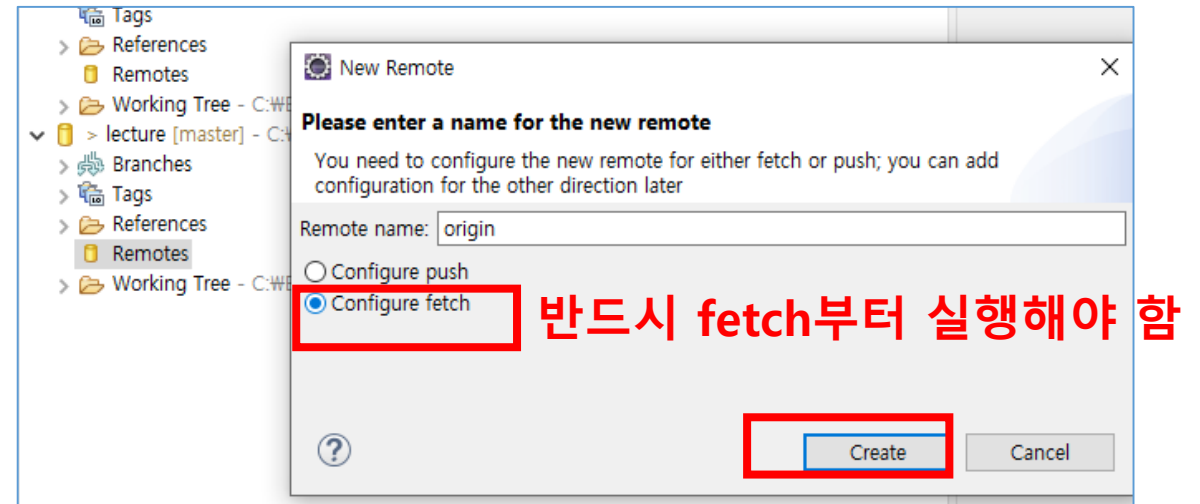


4. 원격 저장소와 연결 - fetch

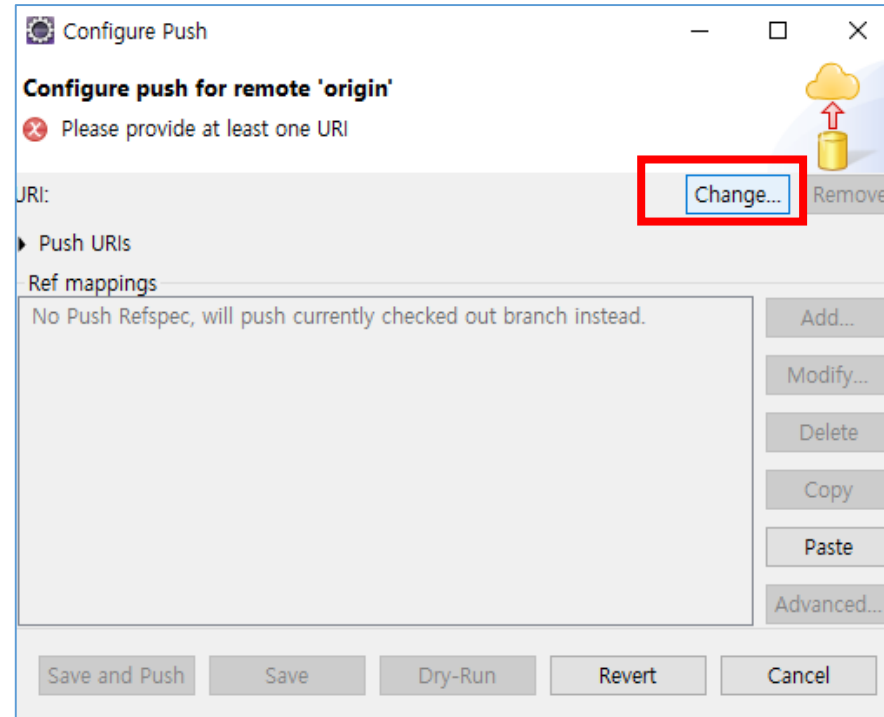
- 원격 저장소의 URL을 복사해 둔다.



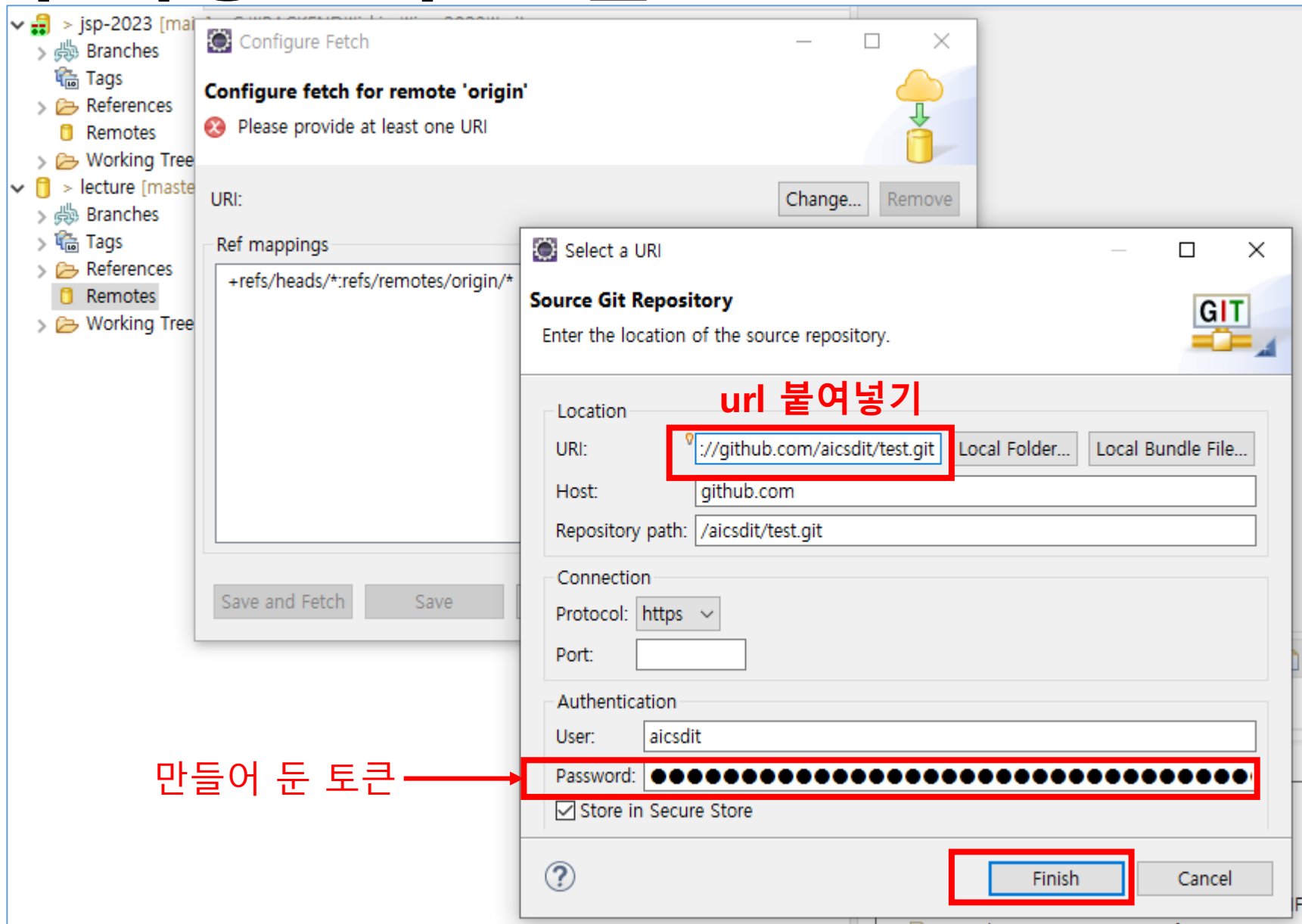
- 로컬 저장소의 Remote 영역을 생성한다.



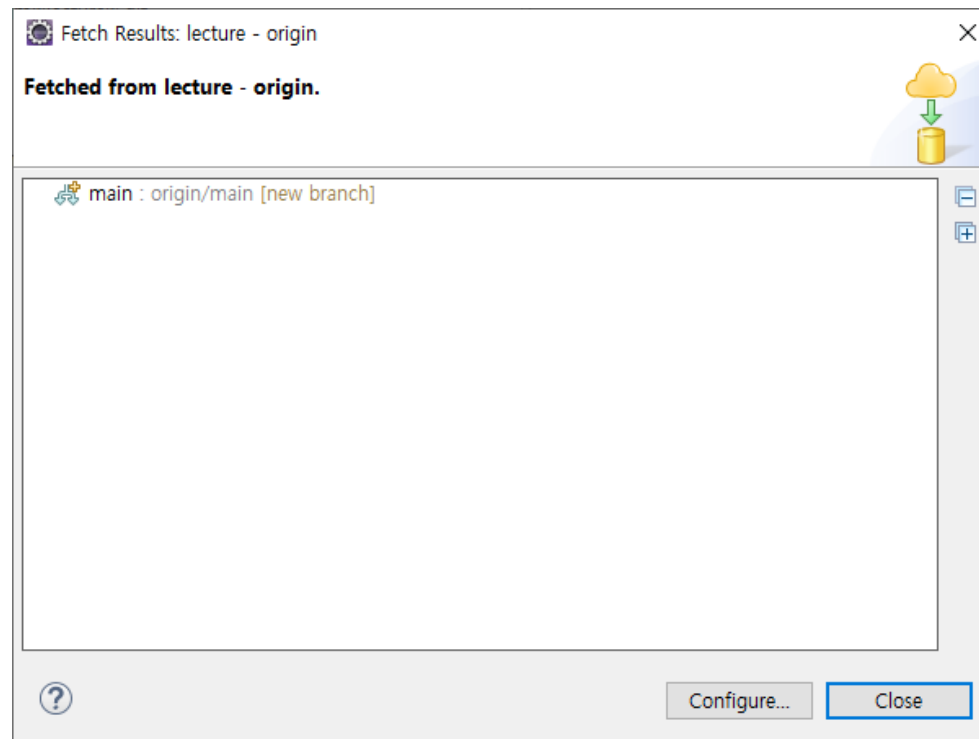
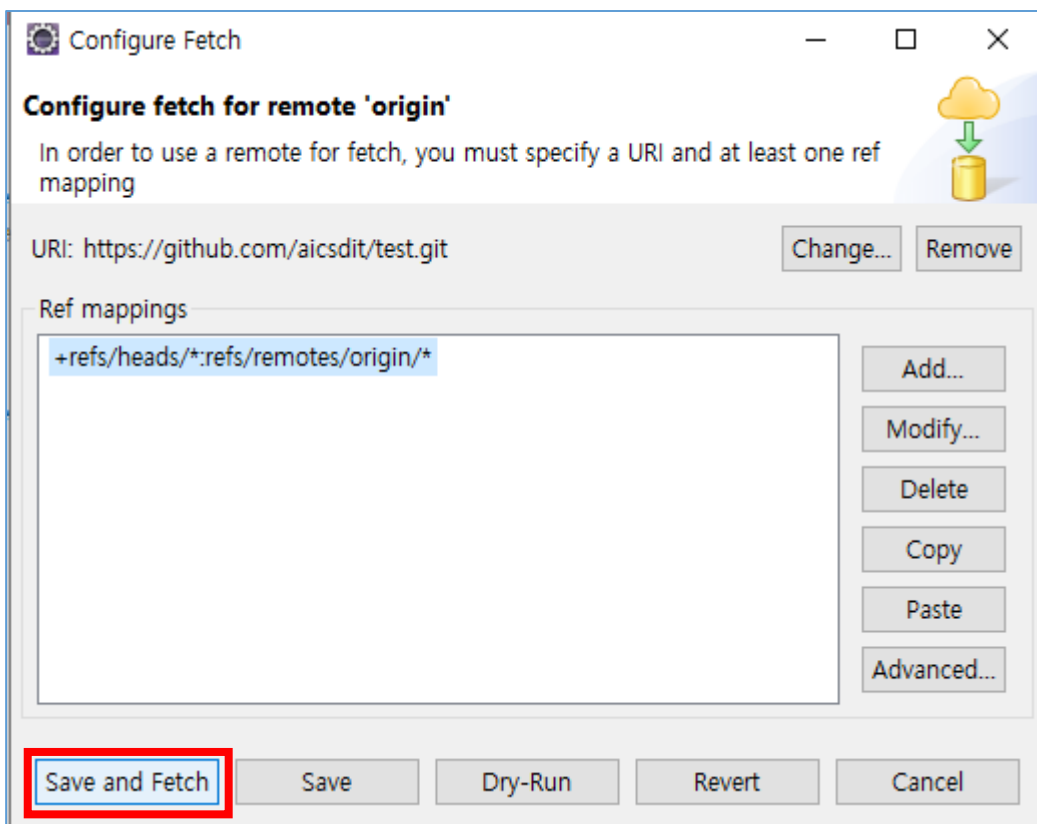
4. 원격 저장소와 연결 - fetch



4. 원격 저장소와 연결 - fetch

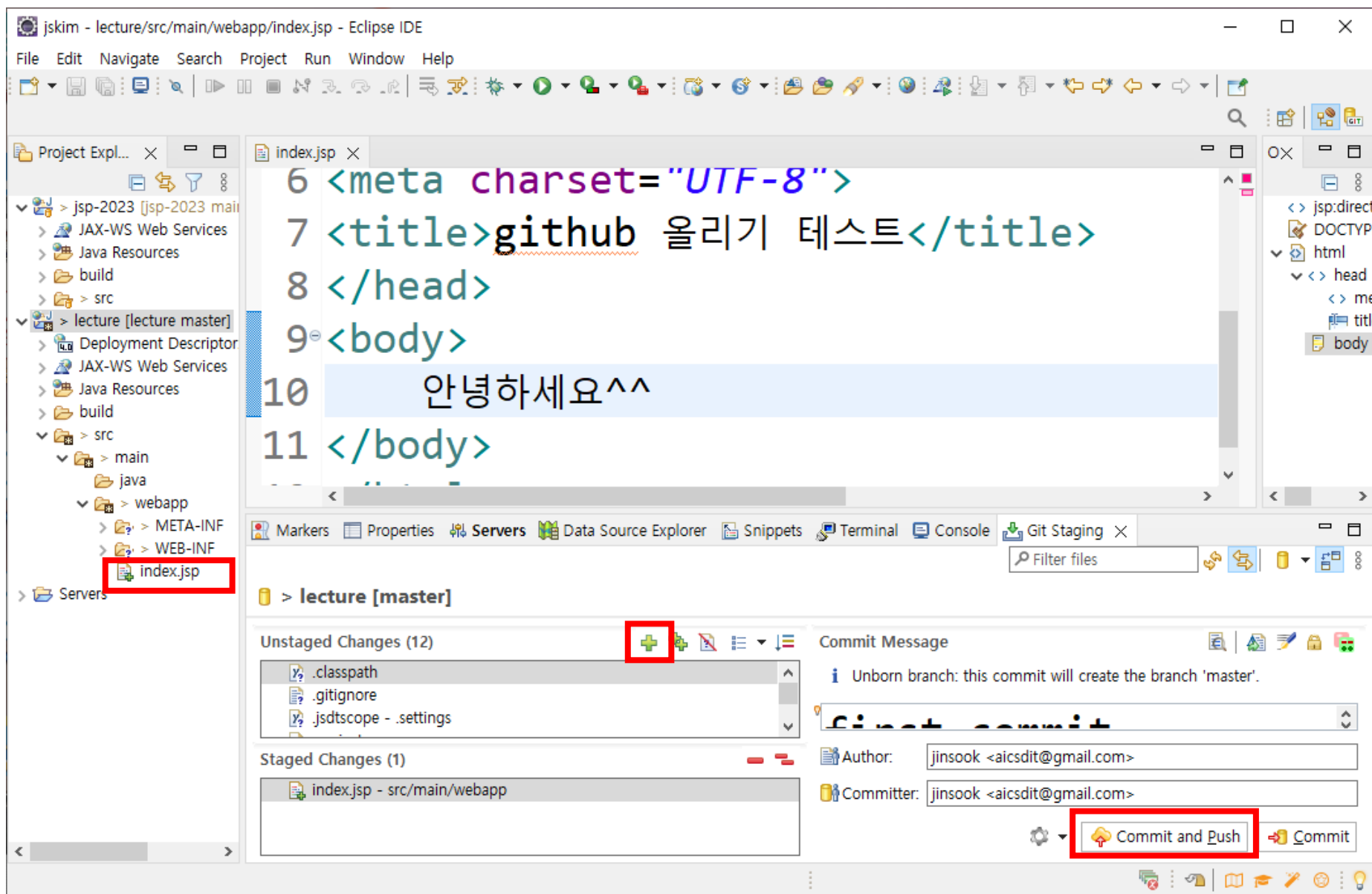


4. 원격 저장소와 연결 - fetch

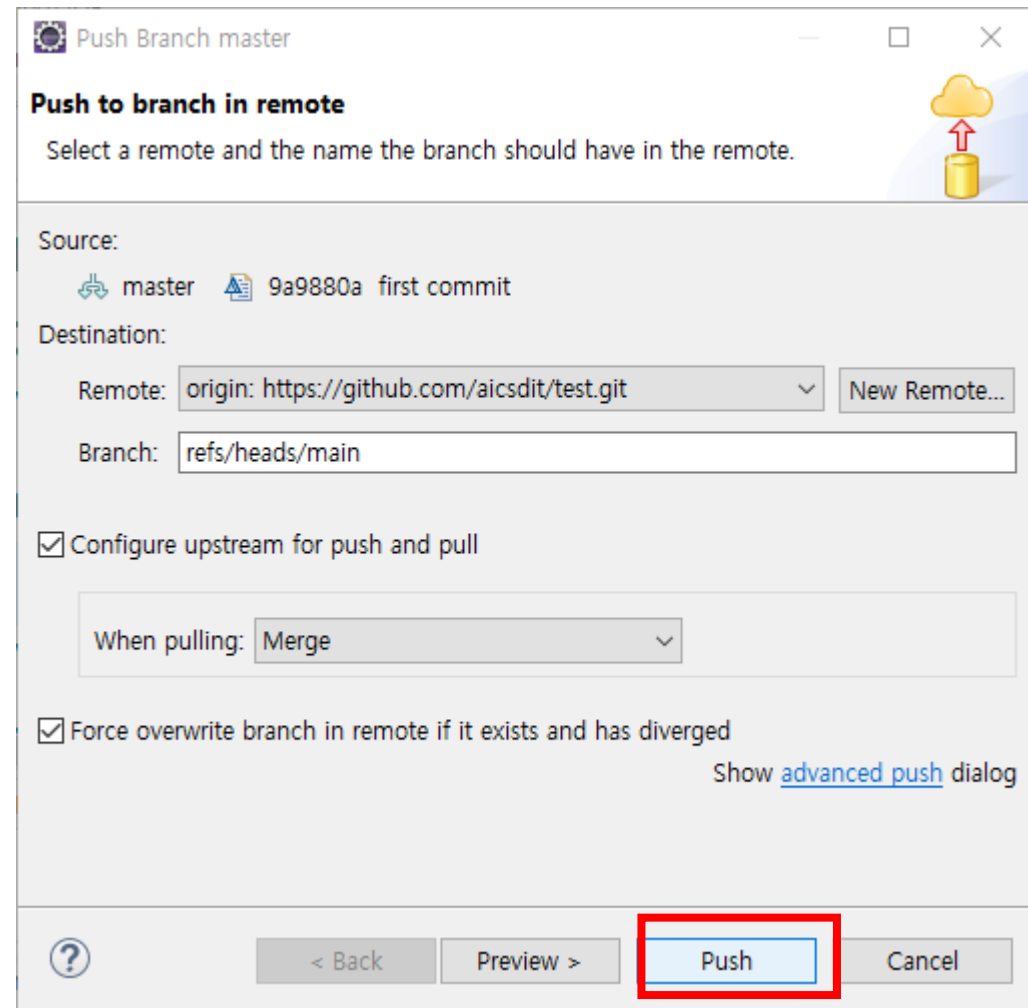
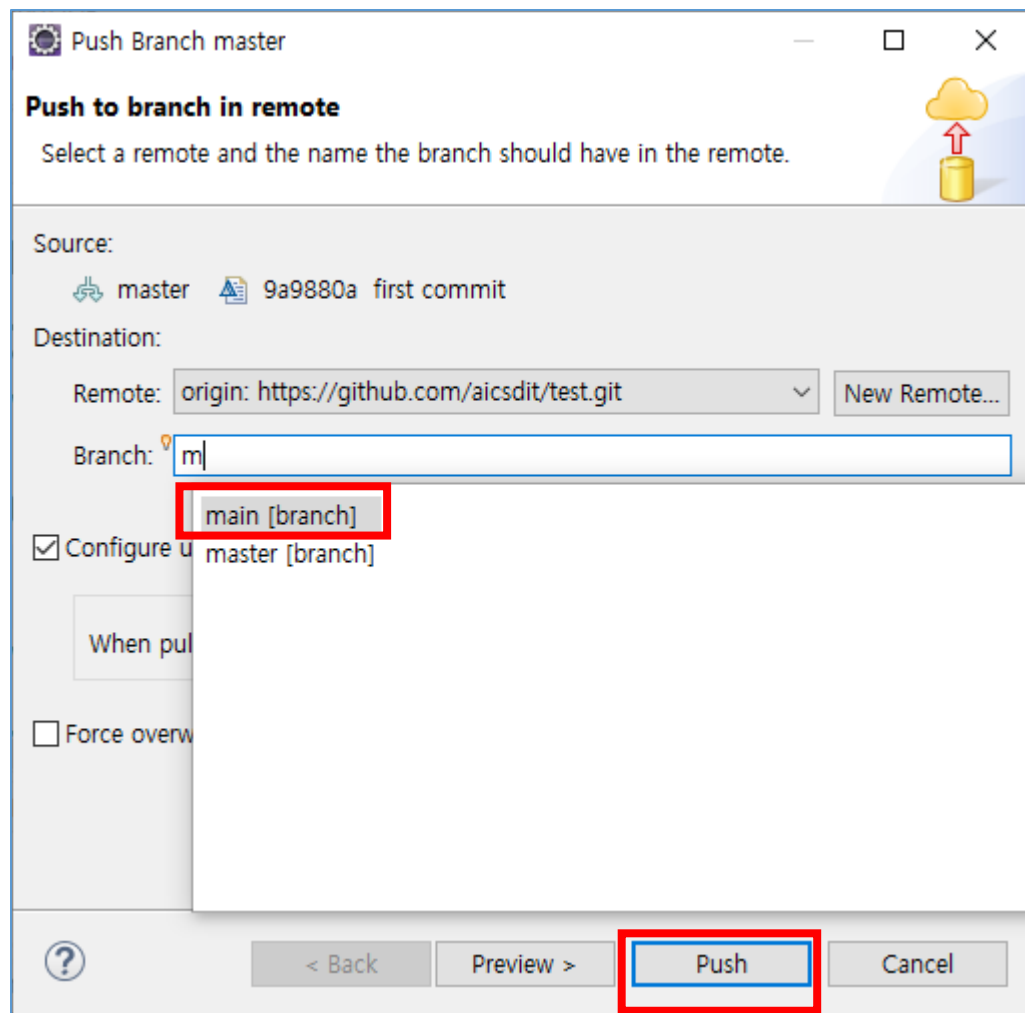


4. 원격 저장소와 연결 - push

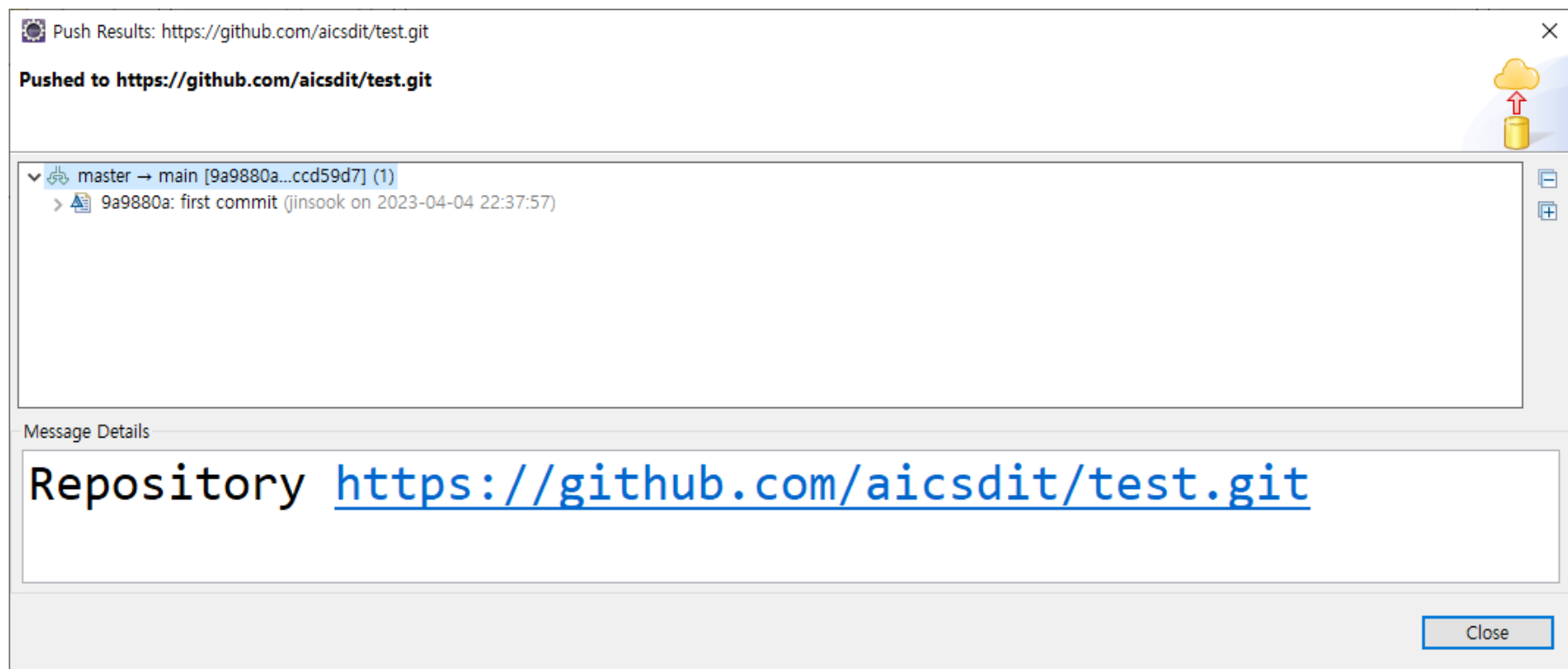
- 테스트를 위해 프로젝트에 jsp 파일을 하나만 들어 push 테스트



4. 원격 저장소와 연결 - push

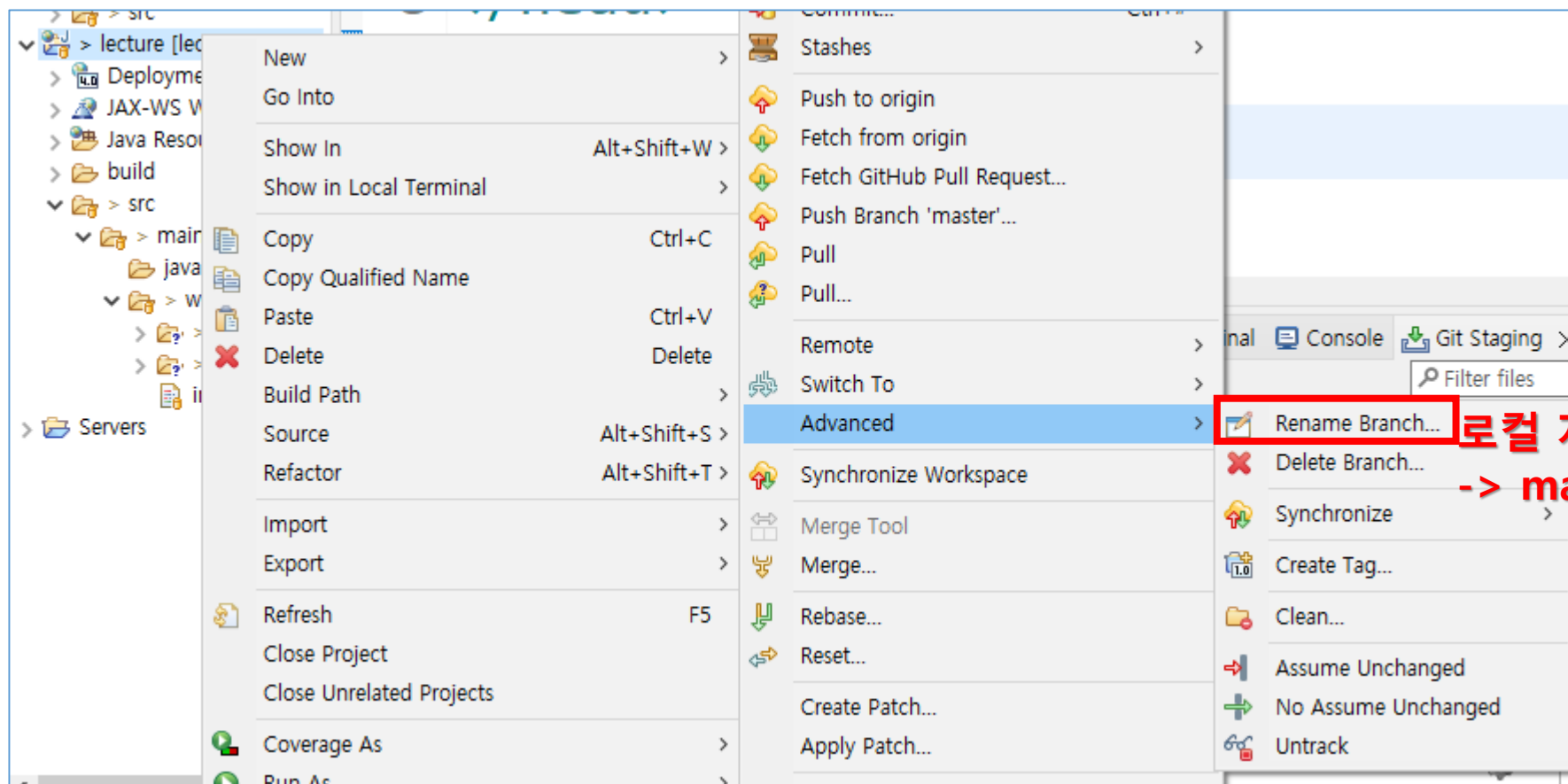


4. 원격 저장소와 연결 - push



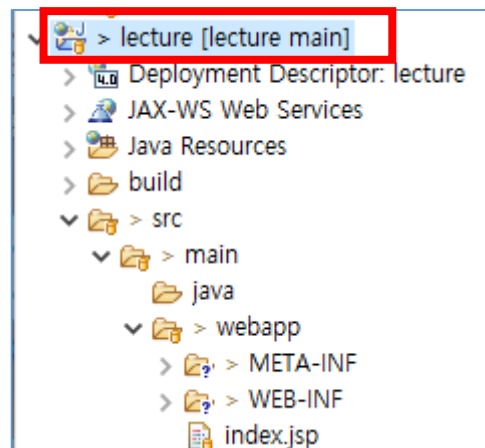
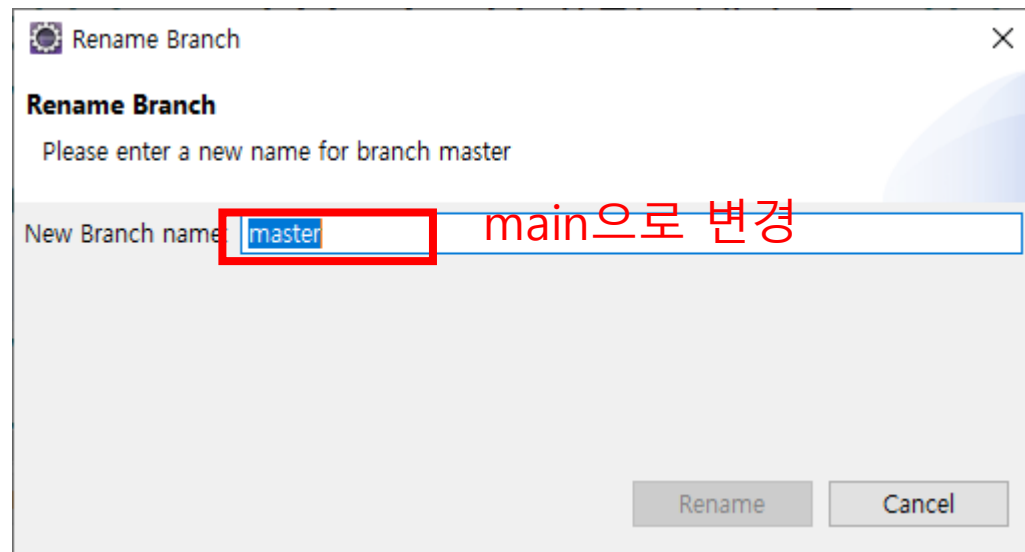
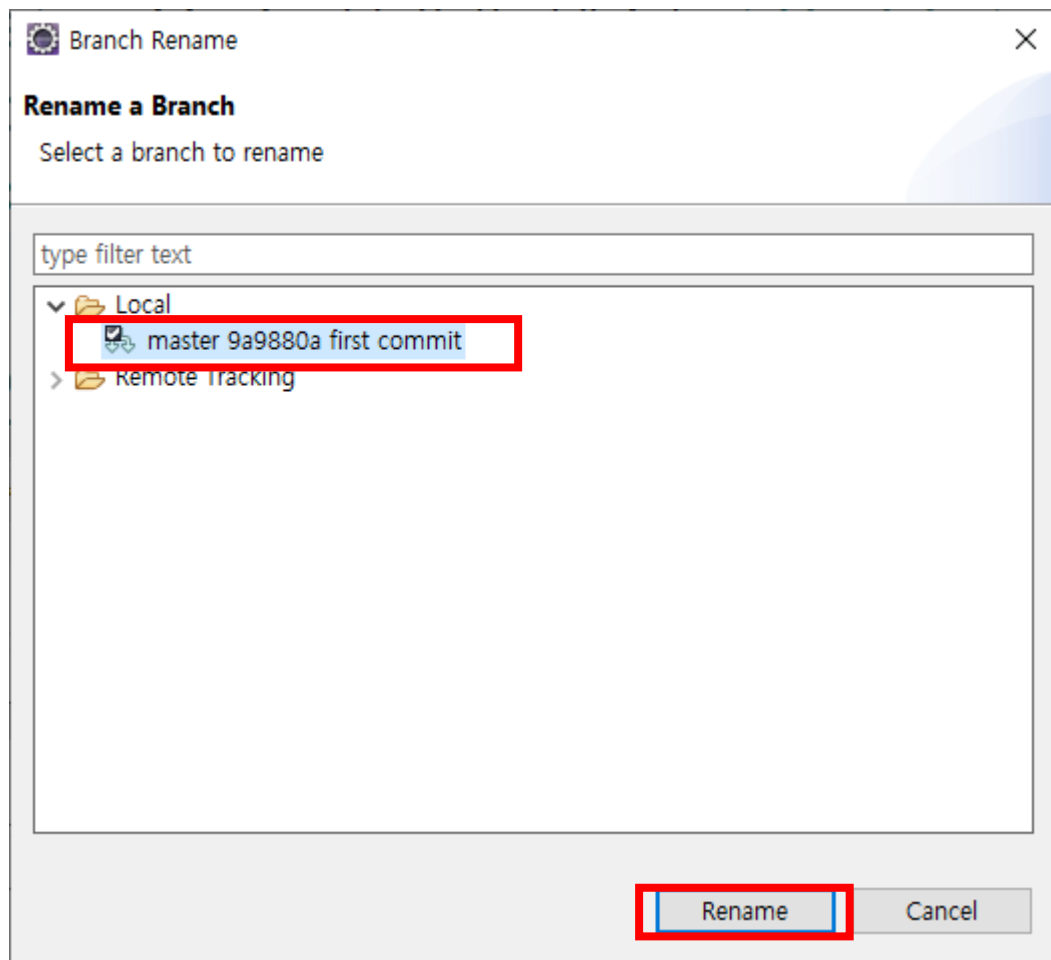
5. 로컬 저장소 디폴트 브랜치 변경

- 원래 master로 되어 있는 브랜치를 원격에서 프로젝트를 fetch해 온 뒤에 변경



로컬 저장소 기본 브랜치명 변경
-> main으로

5. 로컬 저장소 디폴트 브랜치 변경



[필요시] .gitignore 파일 변경

- .gitignore 란
 - 로컬 저장소의 파일들을 원격 저장소에 push하려고 할 때 제외하는 파일들을 미리 설정하는 파일
 - 사용하는 이유
 - 백업, 빌드 정보 등 업로드될 필요가 없는 파일을 자동으로 제외하기 위해
 - 보안상으로 위험성이 있는 파일을 업로드되지 않도록 하기 위해
 - 원하지 않는 파일이 업로드 되지 않도록 하기 위해(너무 용량이 큰 파일 등)
 - 데이터 충돌로 인한 손실 방지 등

[필요시] .gitignore 파일 변경

 gitignore.io

자신의 프로젝트에 꼭 맞는 .gitignore 파일을 만드세요

Windows X

Eclipse X

Java X

생성

[소스 코드](#) | [커맨드라인 문서](#)

운영체제, 사용IDE, 프로그래밍 언어를 엔터를 치면서 입력한 뒤 생성 버튼을 누르면 오른쪽 파일이 생성된다.

https://www.toptal.com/develop x +

toptal.com/developers/gitignore/api/windows,eclipse,java

```
# Created by https://www.toptal.com/developers/gitignore/api/windows,eclipse,java
# Edit at https://www.toptal.com/developers/gitignore?templates=windows,eclipse,java
```

```
### Eclipse ###
```

```
.metadata
bin/
tmp/
*.tmp
*.bak
*.swp
*~.nib
local.properties
.settings/
.loadpath
.recommenders
```

```
# External tool builders
.externalToolBuilders/
```

```
# Locally stored "Eclipse launch configurations"
*.launch
```

```
# PyDev specific (Python IDE for Eclipse)
*.pydevproject
```

```
# CDT-specific (C/C++ Development Tooling)
.cproject
```

```
# CDT- autotools
.autotools
```

```
# Java annotation processor (APT)
.factorypath
```

```
# PDT-specific (PHP Development Tools)
.buildpath
```

6. .gitignore 파일 변경

```
https://www.toptal.com/develop x +
toptal.com/developers/gitignore/api/windows,eclipse,java

# Created by https://www.toptal.com/developers/gitignore/api/windows,eclipse,java
# Edit at https://www.toptal.com/developers/gitignore?templates=windows,eclipse,java

### Eclipse ###
.metadata
bin/
tmp/
*.tmp
*.bak
*.swp
*~.nib
local.properties
.settings/
.loadpath
.recommenders

# External tool builders
.externalToolBuilders/

# Locally stored "Eclipse launch configurations"
*.launch

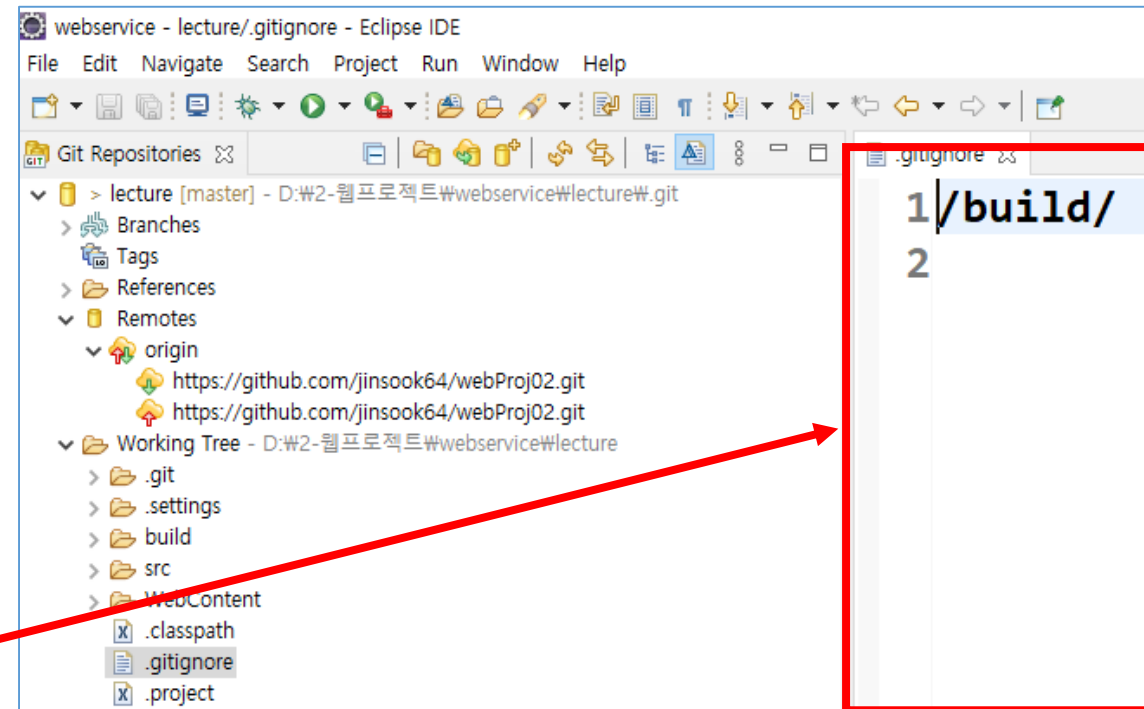
# PyDev specific (Python IDE for Eclipse)
*.pydevproject

# CDT-specific (C/C++ Development Tooling)
.cproject

# CDT- autotools
.autotools

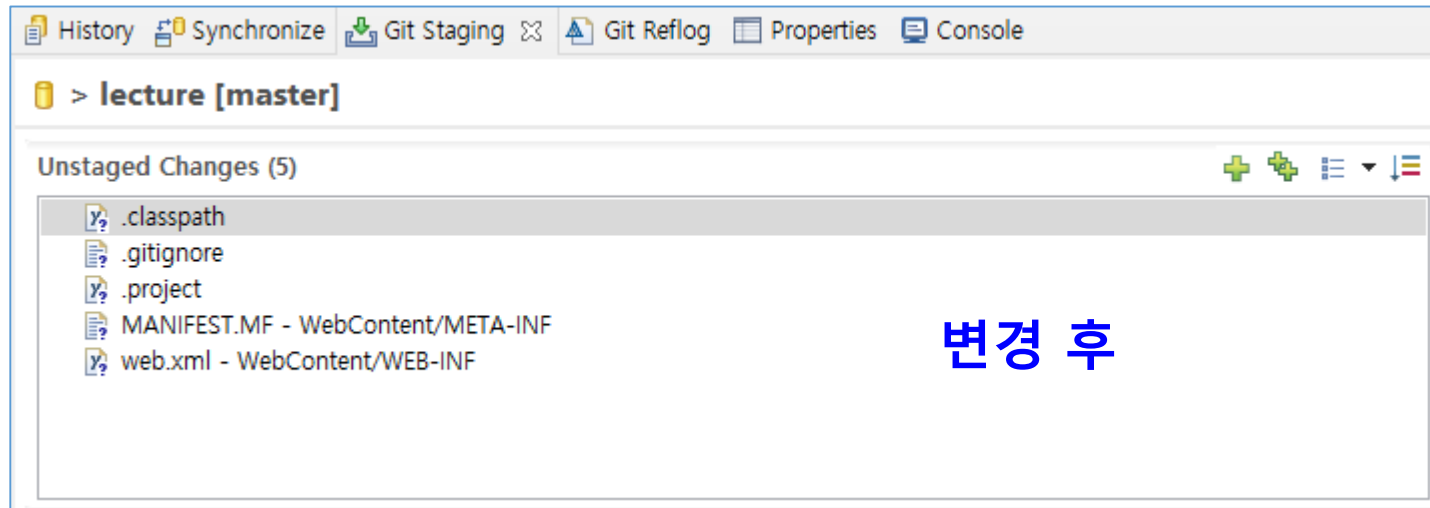
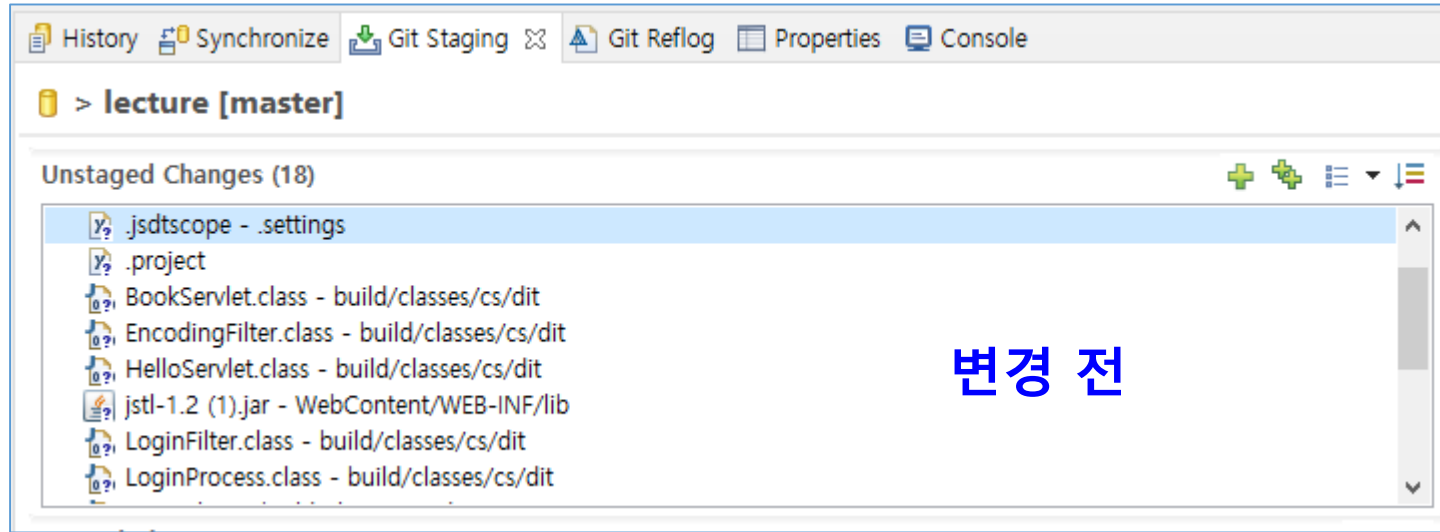
# Java annotation processor (APT)
.factorypath

# PDT-specific (PHP Development Tools)
.buildpath
```



생성된 파일을 복사하여 이클립스 로컬 저장
소인 working directory 의 .gitignore 파
일에 붙여넣기 한다.

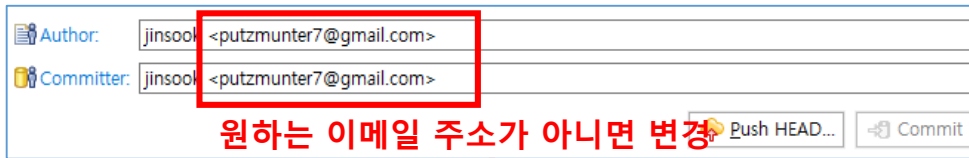
6. .gitignore 파일 변경



위는 .gitignore 파일 변경 전이고
아래는 .gitignore 파일 변경 후이다.
자동생성된 파일 등 github에 업로드
될 필요가 없는 파일들이 빠져있다.

[문제발생 시] Author와 Committer 변경

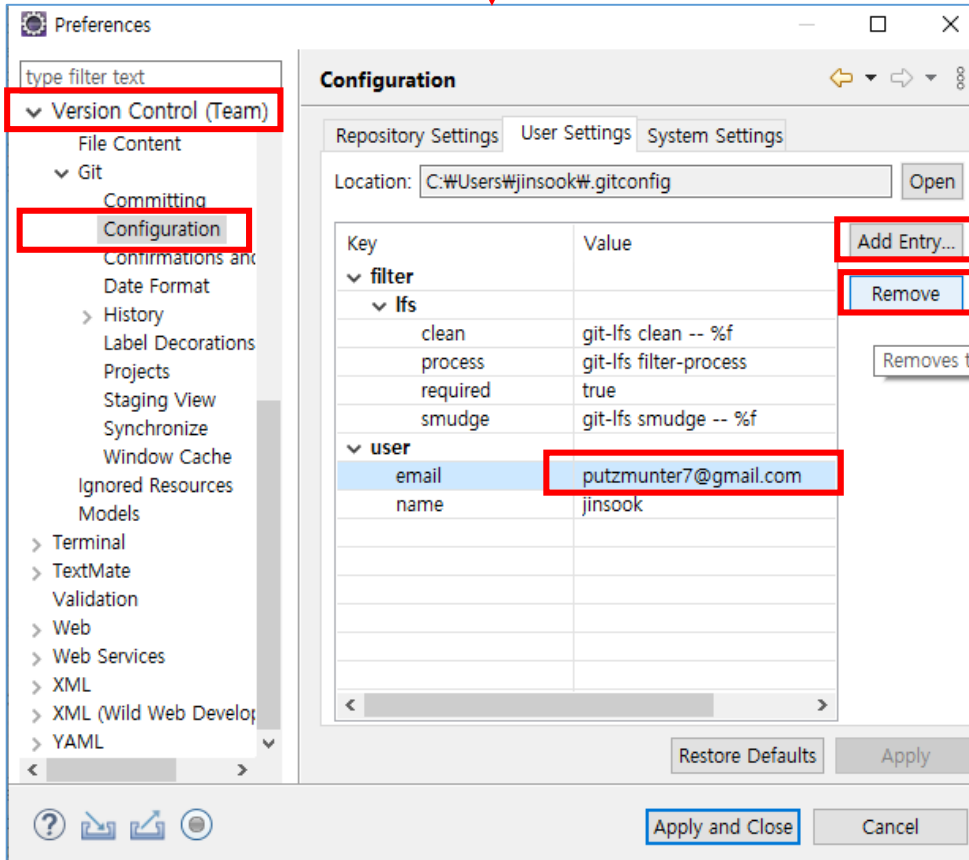
- github의 이메일과 eclipse의 내용이 다를 경우 변경하기



Author: jinsook <putzmunter7@gmail.com>
Committer: jinsook <putzmunter7@gmail.com>

원하는 이메일 주소가 아니면 변경

Windows-Preferences

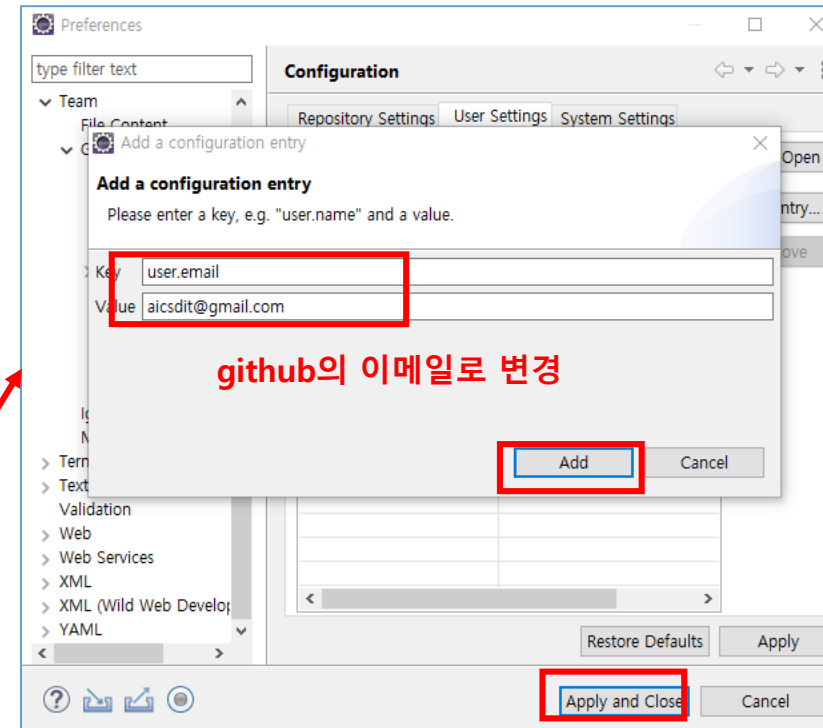


Configuration

Key	Value
filter	
lfs	
clean	git-lfs clean -- %f
process	git-lfs filter-process
required	true
smudge	git-lfs smudge -- %f
user	
email	putzmunter7@gmail.com
name	jinsook

Add Entry... Remove

Restore Defaults Apply Apply and Close Cancel



Add a configuration entry

Please enter a key, e.g. "user.name" and a value.


Key: user.email
Value: aicsdit@gmail.com

github의 이메일로 변경

Add Cancel

Apply and Close Cancel

이클립스 재 실행



Author: jinsook <aicsdit@gmail.com>
Committer: jinsook <aicsdit@gmail.com>

참조 사이트

- 기본 사용법
 - <https://aonee.tistory.com/16?category=377054>
- 토큰 생성과 사용법
 - <https://joytk.tistory.com/58>