

Git classroom에서 eclipse로
프로젝트 clone 및 commit 방법

순서

1. 학교 PC 사용시 로컬 폴더 만들기
2. 새로운 과제가 생기면 workspace 설정 바꿔주기
3. Git클래스룸에서 clone&import(토큰생성 방법)
4. Java 설정 맞추기
5. 문제마다 새 java project 생성하기
6. 작업 후 반드시 git commit&push하기

1) 학교에서 github push할 때

1. 학교 pc에 로컬 폴더 만들기

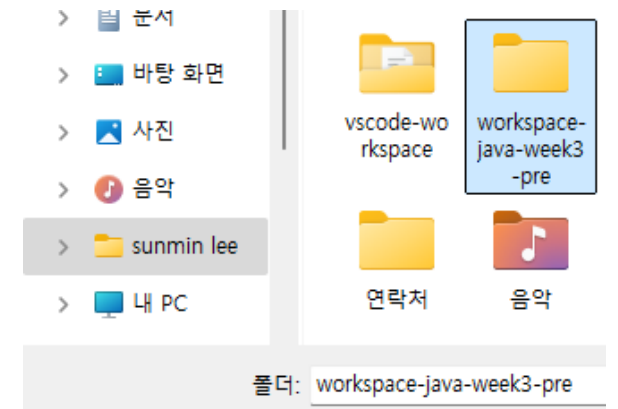
C:\Users\학번\git\java-week3-pre/practicum/late

2. Eclipse에서 워크스페이스 변경

File > Switch Workspace > Other...

C:\Users\학번\workspace-java-week3-pre/practicum/late

*git 폴더와 workspace 폴더 분리하기!

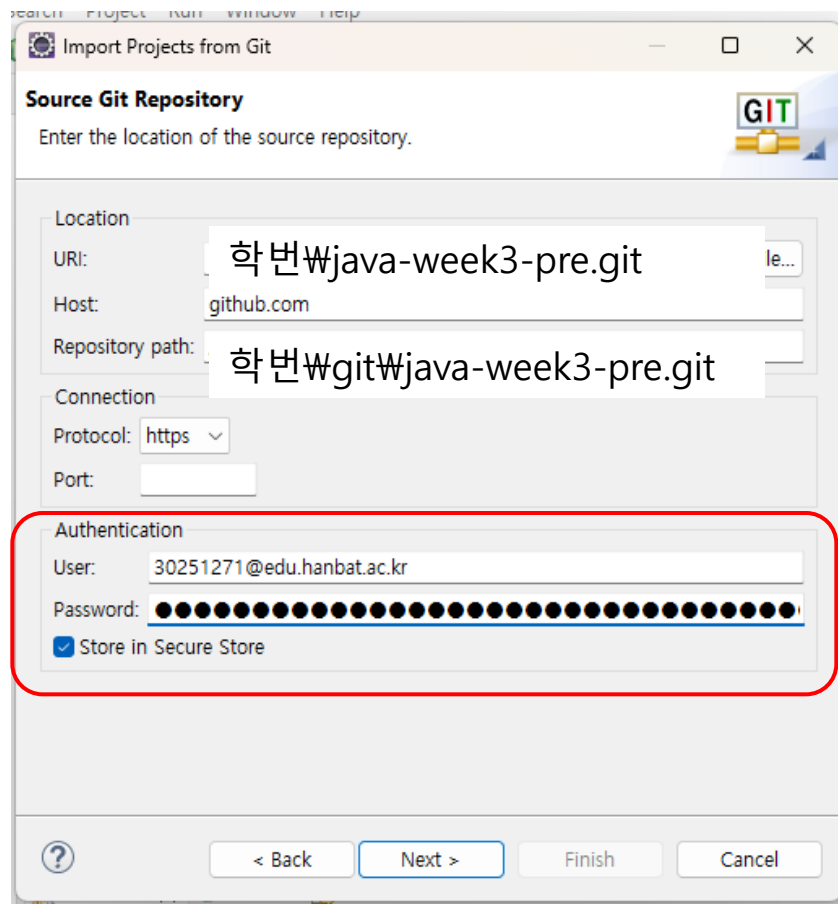


3. Git클래스룸에서 clone&import

File > Import > Git > Projects from Git

Clone URI 선택 후 GitHub URL 붙여넣기

클론할 폴더를 C:\Users\학번\git\java-week3-pre\practicum\late로 설정



이때 토큰이 필요하다
토큰 생성 방법은 다음 페이지에 설명하겠다

토큰 생성 방법

깃허브 프로필 눌러서
settings 찾기



30251271



Set status



Your profile



Your repositories



Your Copilot



Your projects



Your stars



Your gists



Your organizations



Your enterprises



Your sponsors



Settings

Free

New

← Saved replies

Security

🔒 Code security

Integrations

🔗 Applications

🕒 Scheduled reminders

Archives

📖 Security log

📖 Sponsorship log

🔗 Developer settings

🔗 GitHub Apps

🔗 OAuth Apps

🔑 Personal access tokens

Fine-grained tokens

Preview

Tokens (classic)



Link to social profile 1



Link to social profile 2



Link to social profile 3



Link to social profile 4

Company

You can @mention your company's GitHub organization to link it.

Location

☐ Display current local time

Other users will see the time difference from their local time.

토큰 생성 방법

Generate new token ▼

Generate new token Beta
Fine-grained, repo-scoped

Generate new token (classic)
For general use

... OAuth access tokens. They can be used instead of a
[Authenticate to the API over Basic Authentication](#).

Tokens (classic)

Note

token

토큰이름 아무거나

What's this token for?

Expiration *

No expiration ▼

기한 설정 No expriation으로

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

Select scopes

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

- ☒ repo Full control of private repositories
 - ☒ repo:status Access commit status
 - ☒ repo_deployment Access deployment status
 - ☒ public_repo Access public repositories
 - ☒ repo:invite Access repository invitations
 - ☒ security_events Read and write security events

repo만 체크하기

Generate token

클릭!

토큰 생성 방법

Personal access tokens

Generate new token

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

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



토큰 생성됨



토큰은 생성된 직후, 딱 한번만 볼 수 있다
반드시 안전한 곳에 저장하자

토큰 생성 방법

Personal access tokens (classic)

Generate new token ▾

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

token — repo

Last used within the last week

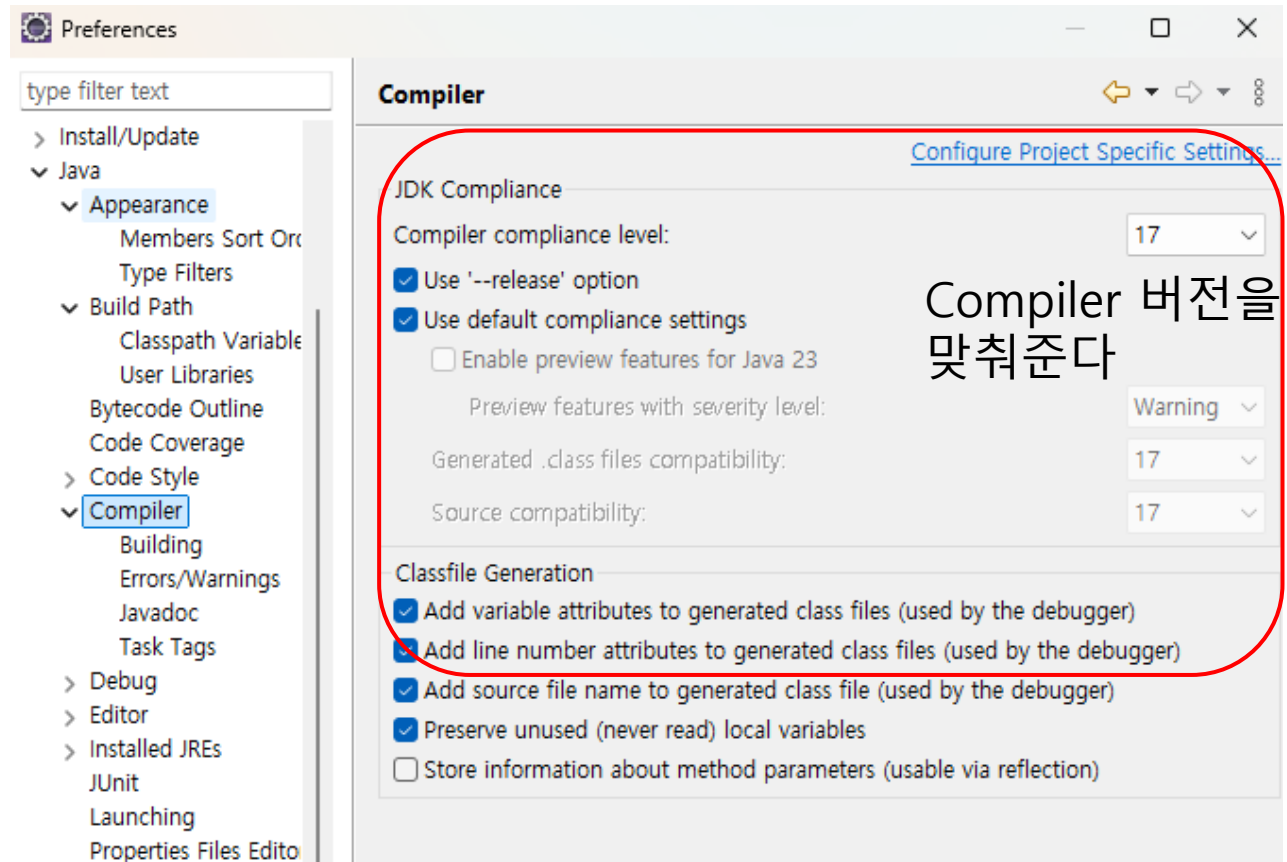
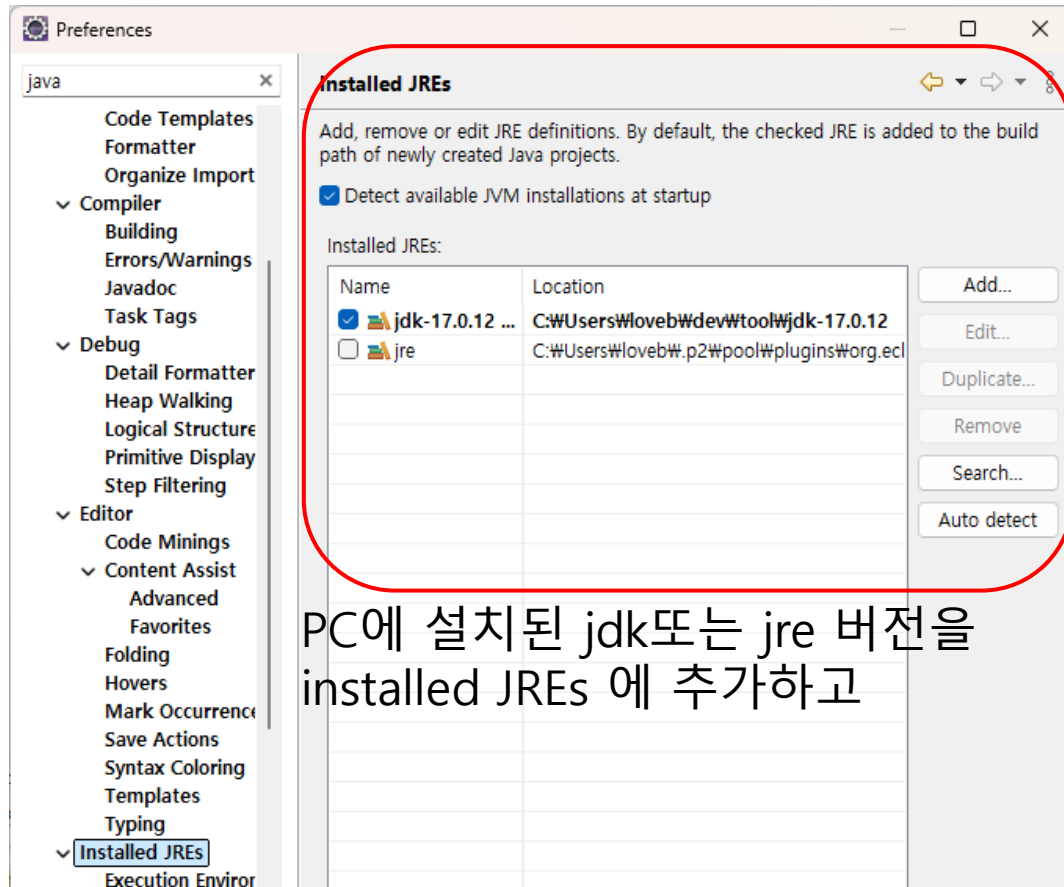
Delete

⚠ This token has no expiration date.

토큰은 생성된 직후, 딱 한번만 볼 수 있다
반드시 안전한 곳에 저장하자

4. Java 설정 맞추기

Window-preferences에서 installed JREs와 Compiler 둘 다 맞추기



5. 문제마다 새 java project 생성하기

New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name: problem03

☐ Use default location **!!!!체크해제!!!!**

Location: C:\Users\loveb\git\java_week3_pre\problem03 [Browse...](#)

JRE

☐ Use an execution environment JRE: JavaSE-17

☒ Use a project specific JRE: jdk-17.0.12

☐ Use default JRE 'jdk-17.0.12' and workspace compiler preferences [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets [New...](#)

Working sets: [Select...](#)

Module

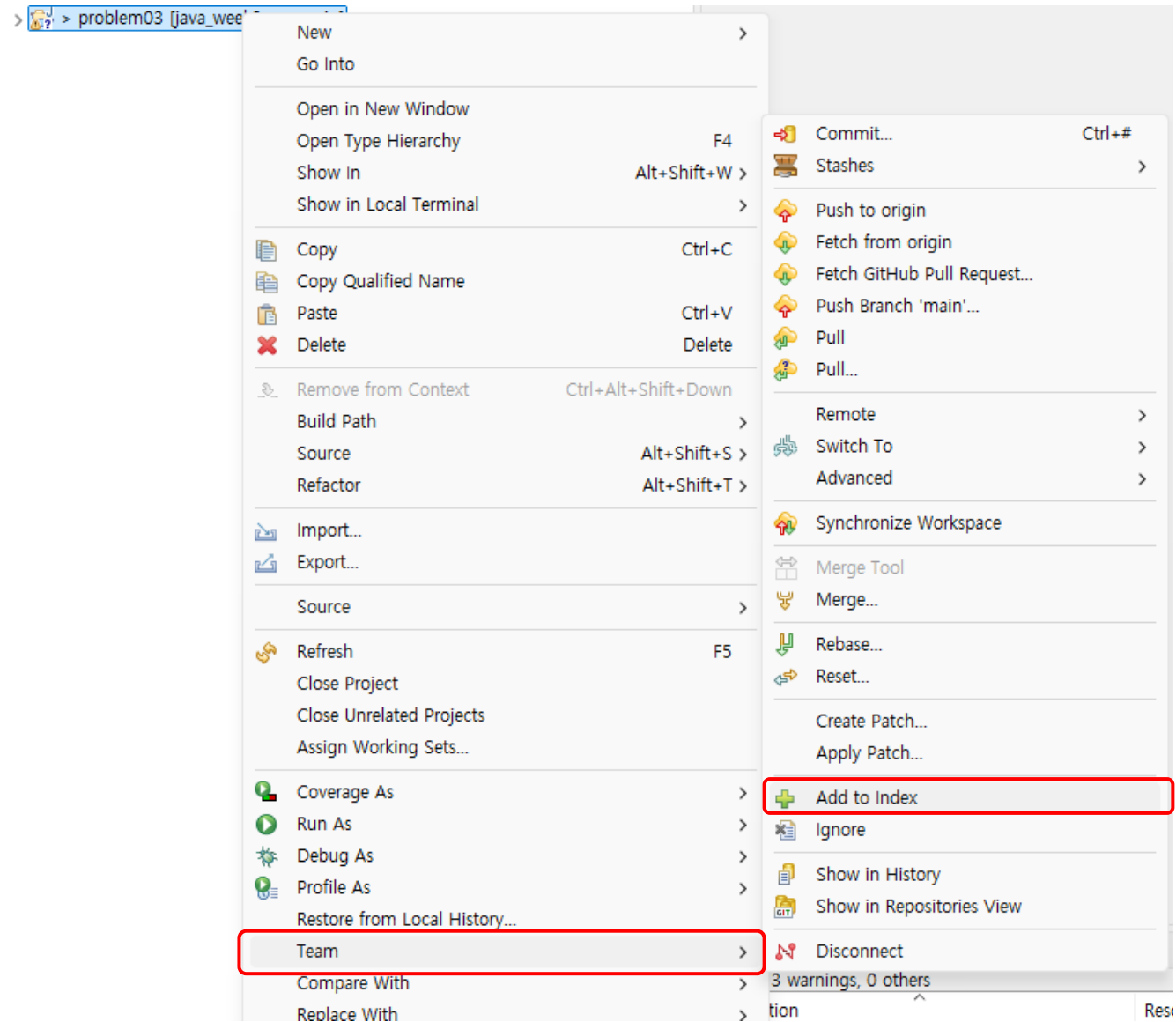
☐ Create module-info.java file **!!!!체크해제!!!!**

Module name:

☐ Generate comments

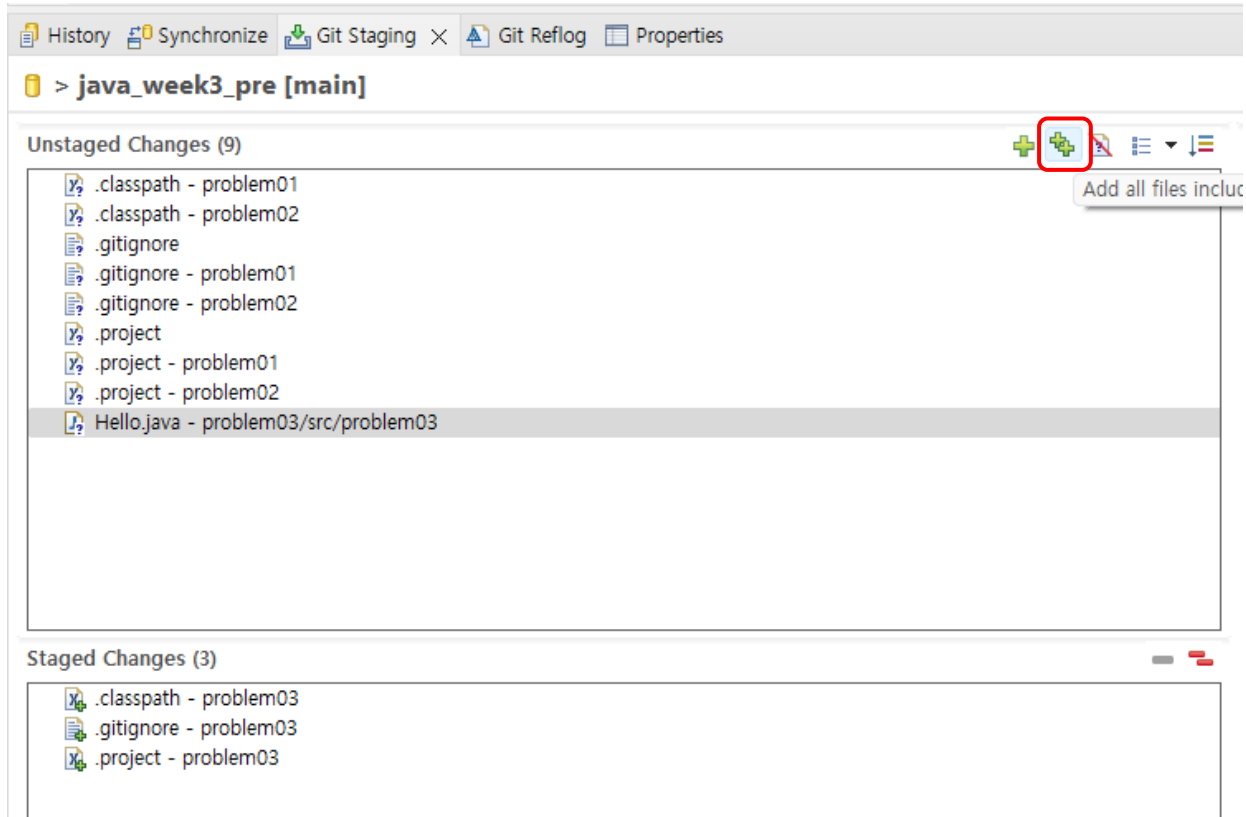
[?](#) [< Back](#) [Next >](#) **Finish** [Cancel](#)

6. 작업 후 반드시 git commit&push하기






이동, Git Staging -  클릭, staged changes에 추가된 거 확인하고
Commit and Push 클릭



Commit Message

 Commit and Push...