Creative Software Design

1 - Lab2 - g++, make, gdb

Yoonsang Lee
Fall 2020

Today Topic

- G++
- Make
- GDB
- Creating a Gitlab account
- Time for Assignment 1-1

What is G++ ?

- Open-sourced C++ compiler

- Most formats and options are the same as the default C compiler (cc)
  - **g++** [options] <infile> ...
    - -c : compile and assemble, but do not link Create only object file (.o) without creating executable
    - -g : debug info. Contains information necessary for debugging (source code, etc.)
    - -o <outfile> : Place the output into <outfile>
    - -I<dir> : include directory. (directory name to look for headers when compiling)
    - -L<dir> : library directory. (Directory name to look for library files when linking)
    - -D<symbol>[=def] : define a macro to use at compile time
    - ... : There are numerous other options.

Example : Compile & Link

- Write main.cpp, print.cpp, print.h

```
// main.cpp
#include "print.h"
int main() {
  print_hello();
  return 0;
}
```

```
// print.cpp
#include <iostream>
void print_hello() {
  std::cout << “hello world!” << endl;
}
```

```
// print.h
void print_hello();
```

Example : Compile & Link

- Compile and link the two source files (main.cpp, print.cpp)

(Shell – working directory)

```
$ g++ -c -o main.o main.cpp
$ g++ -c -o print.o print.cpp
$ g++ -o hello_world main.o print.o
```

(Shell – working directory)

```
$ g++ -o hello_world main.cpp print.cpp
```

- Run the created executable

(Shell – working)

```
$ ./hello_world
```

Compile & Link

- This is a very brief introduction on how to compile and link using g++.

- The details of compile & link process will be covered in the lecture "5-Compilation and Linkage, CMD Args".

Make

- Build tools that have been around for a long time on Unix operating systems
  - Rules for how to compile and link the source to create an executable

Makefile

- When “make” is run, find Makefile (or makefile) in that directory and runs it as usual

- How to write Makefile

```
target: prerequisites
<TAB>command1
<TAB>command2
```

  - target : File or state to create(such as.o or excutable) 등)
  - prerequisites : List of files needed to create target
  - command(s) :Each step command to create a target. <Tab> must be placed before the command.

Example: Writing / Running makefile

- Write makefile

(Shell – working directory)

```
$ vi Makefile
```

```
hello_world: main.o print.o
	g++ -o hello_world main.o print.o
main.o: main.cpp
	g++ -c main.cpp
print.o: print.cpp
	g++ -c print.cpp
clean:
	rm hello_world main.o print.o
```

Example: Writing / Running makefile

- Execute makefile (1) : generate executable file

(Shell – working directory)

```
$ make
```

- Execute makefile (2) : Remove Excutable file and All object files

(Shell – working directory)

```
$ make clean
```

GDB

Debugging tools - help you find the wrong parts of your program by checking its status when the program is running or when it crashes.

When you build a program, you need to give it the -g option to see the information you need.

gdb [options] <command>

- <command> : If the current directory is not in your PATH, you must include ./.
- Basic command
  - r [arguments] : Run the given command.
  - bt : backtrack. Show current call stack status.
  - up/down [steps] : Move up / down a given step from the current position of the call stack.
  - p <variable> : Display the value of a given variable.
  - q : exit gdb process.
  - Use more easy-to-use improved programs such as cgdb and ddd

Example

(Shell – working directory)

```
$ vi test.cc
```

```
void IncorrectAccess(int* array, int i, int n) {
  if (i < n) {
    array[i] = 0;
    IncorrectAccess(array, i + 1, n);
  }
}
int main() {
  int array[10];
  IncorrectAccess(array, 0, 20);
  return 0;
}
```

(Shell – working directory)

```
$ g++ -o test test.cc
$ gdb ./test
…
(gdb)
…
```

Creating a Gitlab account

Gitlab

- For today's lab assignment, submit your files via Blackboard course homepage.

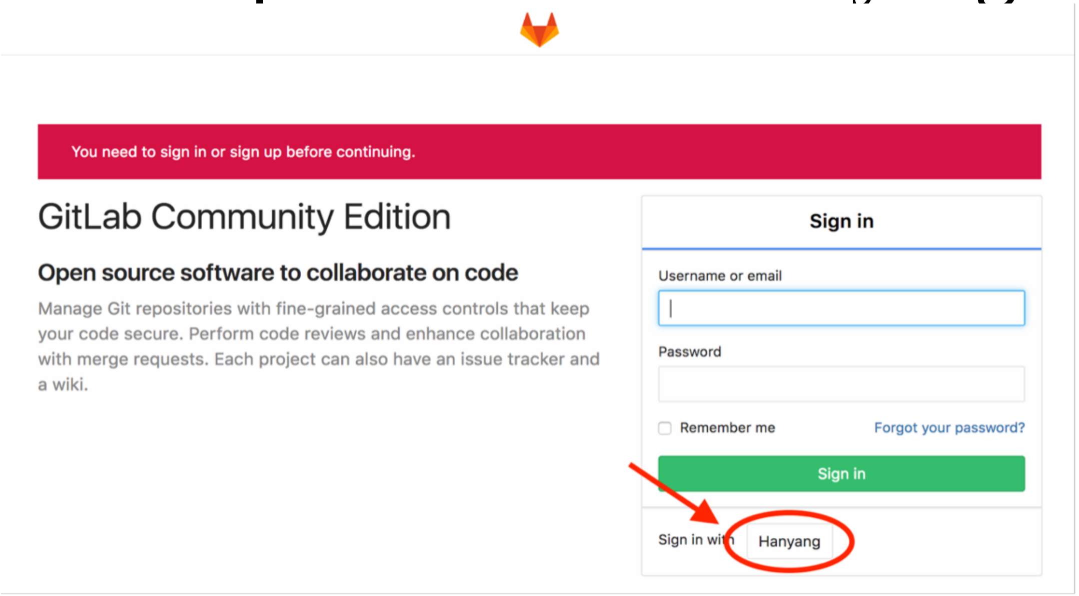
- From next week's lab assignment, submit your files via the gitlab at https://hconnect.hanyang.ac.kr/

- Be sure to create a hconnect account in advance.

- If you already have a hconnect account, just skip this part.

Git

- Access to https://hconnect.hanyang.ac.kr/



You need to sign in or sign up before continuing.

GitLab Community Edition

Open source software to collaborate on code

Manage Git repositories with fine-grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

Sign in

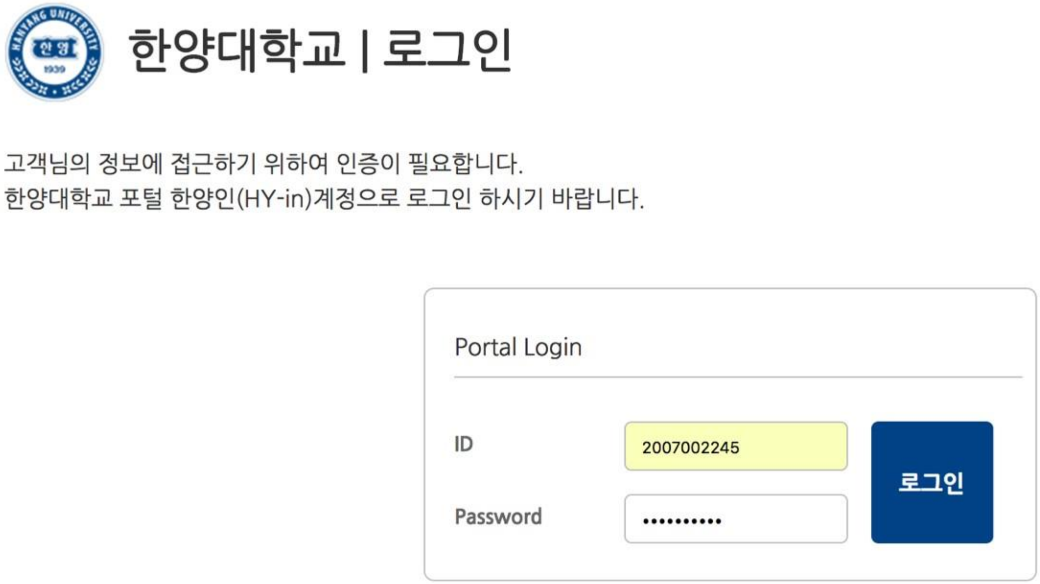
Username or email

Password

□ Remember me

Git

- Login hanyang account



한양대학교 | 로그인

고객님의 정보에 접근하기 위하여 인증이 필요합니다.
한양대학교 포털 한양인(HY-in)계정으로 로그인 하시기 바랍니다.

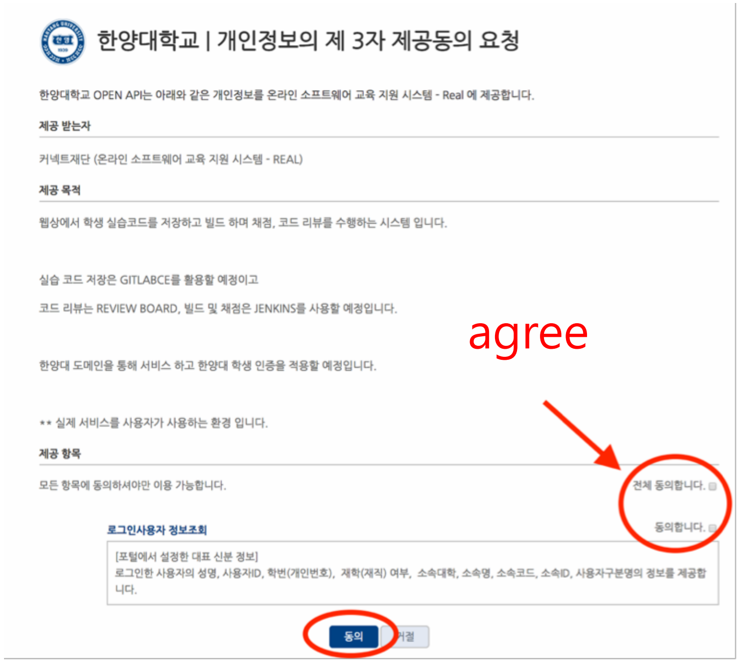
Portal Login

ID 2007002245

Password ••••••••••

Git

- Consent for information provision



한양대학교 | 개인정보의 제 3자 제공동의 요청

한양대학교 OPEN API는 아래와 같은 개인정보를 온라인 소프트웨어 교육 지원 시스템 - Real 에 제공합니다.

제공 받는자

커넥트재단 (온라인 소프트웨어 교육 지원 시스템 - REAL)

제공 목적

웹상에서 학생 실습코드를 저장하고 빌드 하며 채점, 코드 리뷰를 수행하는 시스템 입니다.

실습 코드 저장은 GITLABCE를 활용할 예정이고

코드 리뷰는 REVIEW BOARD, 빌드 및 채점은 JENKINS를 사용할 예정입니다.

한양대 도메인을 통해 서비스 하고 한양대 학생 인증을 적용할 예정입니다.

** 실제 서비스를 사용자가 사용하는 환경 입니다.

제공 항목

모든 항목에 동의하셔야만 이용 가능합니다. 전체 동의합니다. □

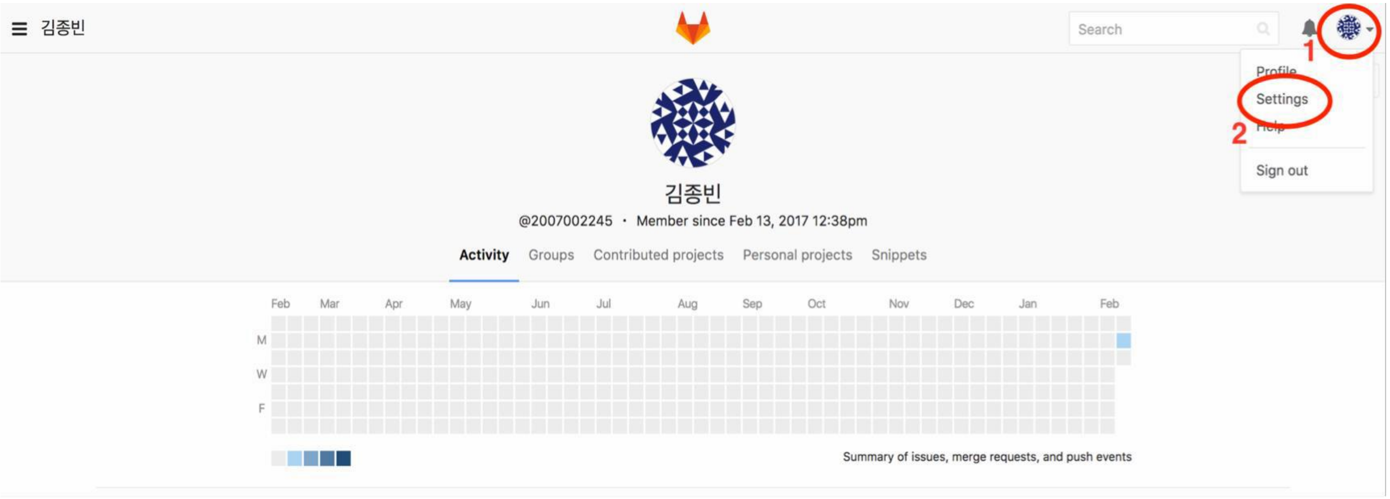
로그인사용자 정보조회 동의합니다. □

[포털에서 설정한 대표 신분 정보]
로그인한 사용자의 성명, 사용자ID, 학번(개인번호), 재학(재직) 여부, 소속대학, 소속명, 소속코드, 소속ID, 사용자구분명의 정보를 제공합니다.

agree

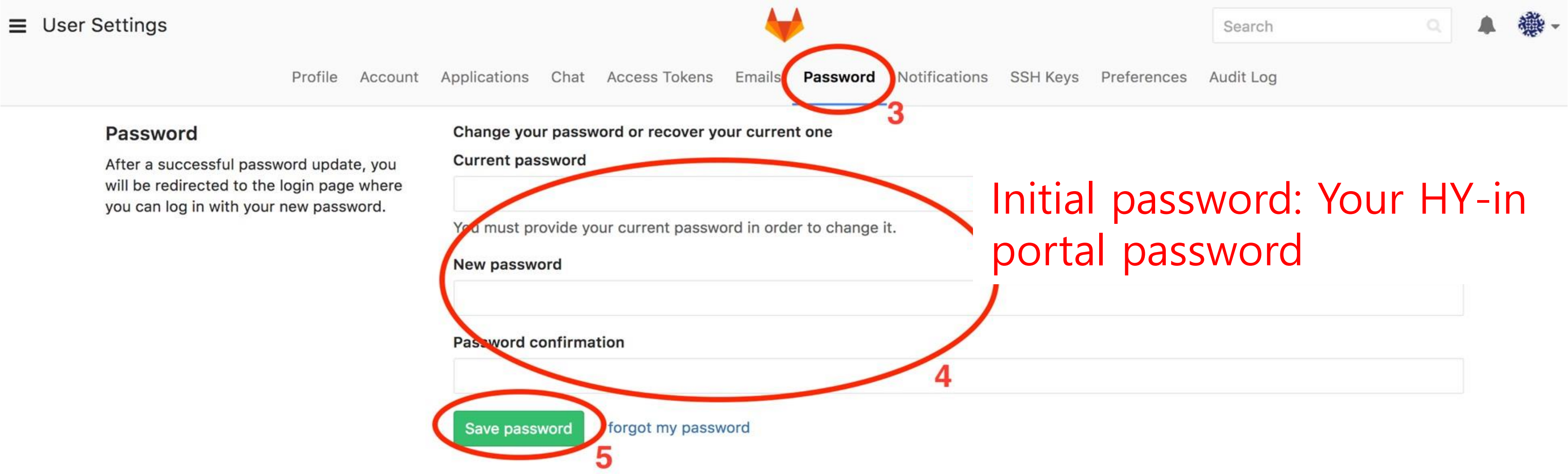
Git

- Set up Password



Git

- Set up Password



Password

After a successful password update, you will be redirected to the login page where you can log in with your new password.

Change your password or recover your current one

Current password

You must provide your current password in order to change it.

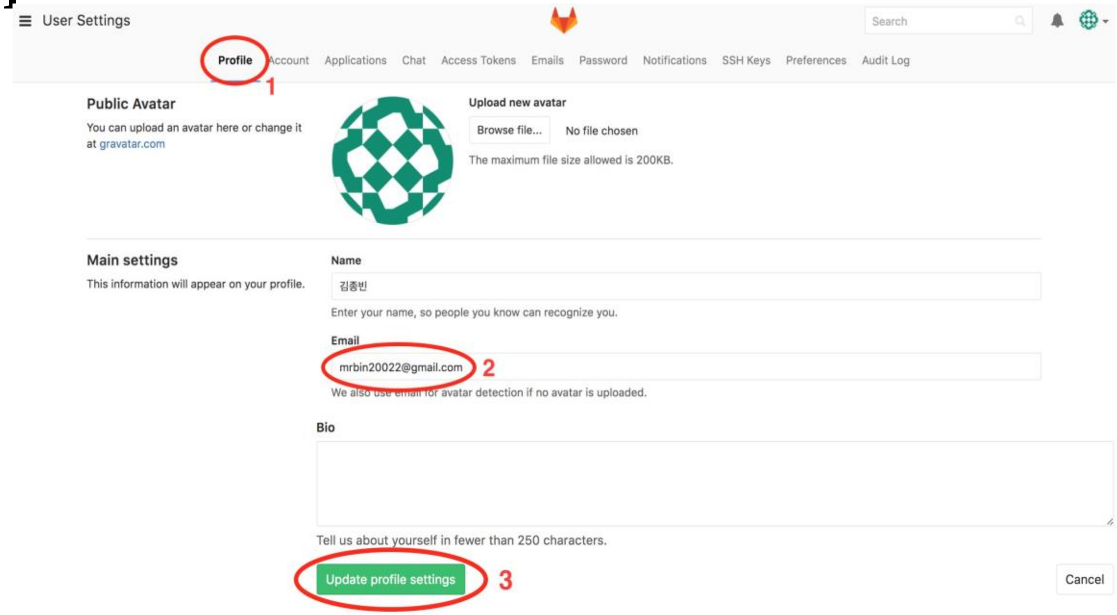
New password

Password confirmation

Initial password: Your HY-in portal password

Git

- Set up Email



Public Avatar

You can upload an avatar here or change it at gravatar.com

Upload new avatar

No file chosen

The maximum file size allowed is 200KB.

Main settings

This information will appear on your profile.

Name

김종빈

Enter your name, so people you know can recognize you.

Email

mrbin20022@gmail.com

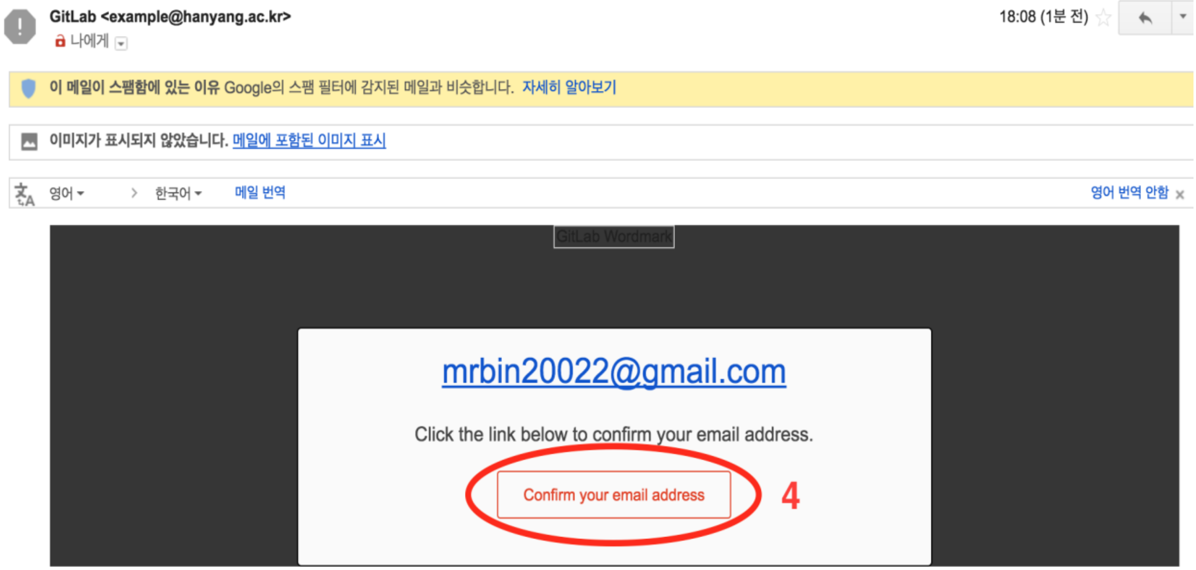
We also use email for avatar detection if no avatar is uploaded.

Bio

Tell us about yourself in fewer than 250 characters.

Git

- Set up Email – Approve from changed email



GitLab <example@hanyang.ac.kr>

나에게

18:08 (1분 전)

이 메일이 스팸함에 있는 이유 Google의 스팸 필터에 감지된 메일과 비슷합니다. 자세히 알아보기

이미지가 표시되지 않았습니다. 메일에 포함된 이미지 표시

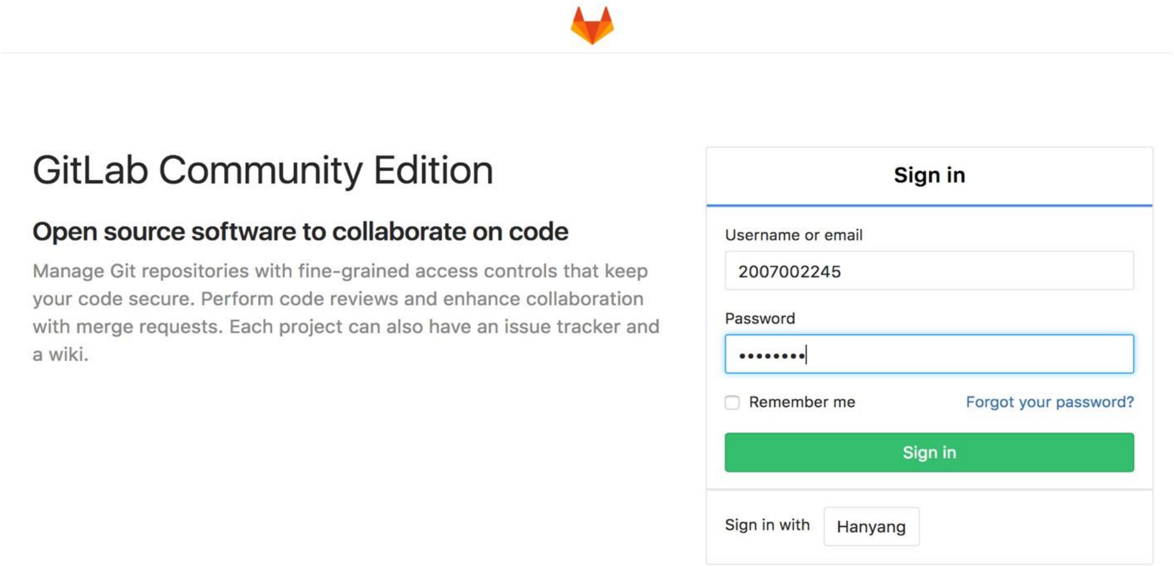
mrbin20022@gmail.com

Click the link below to confirm your email address.

Git

- After this, you can sign in to hconnect with your student ID / email and the password you changed.

  (without using 'Sign in with Hanyang')



GitLab Community Edition

Open source software to collaborate on code

Manage Git repositories with fine-grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

Username or email: 2007002245

Password: •••••••••

Assignment 1-1

- Now, let’s start the assignment 1-1.
- Assignment 1-1 is just for practice, will not be included in the final grade.

- However, you need to complete and submit your answers **to figure out how to set up the environment** and **to create your "hconnect" account** in advance.

- Check the assignment: Blackboard course home - Assignments - “Assignment1-1.pdf”
- Submit your files: Blackboard course home - Assignments - "Assignment1-1, N". (N is the problem number)

- You can leave the lab after submitting your files.