Makefile 만들기

- 과제1에 필요한 Makefile를 만들기
- 기타 다른 Makefile를 만들기
- <를 통한 리다이렉션

개요

1. Makefile?
2. Makefile 만들기
    1) Makefile 만들기 – 단일 .c 파일
    2) Makefile 만들기 – flex와 bison
    3) Makefile 만들기 – 여러 파일

부. <, > Redirection

1. Makefile?

1. Makefile?

- make
  - 주어진 쉘 명령어들을 조건에 맞게 실행하는 프로그램
- Makefile
  - 어떤 조건으로 명령어를 실행할지를 담은 파일
  - 구성요소
    - Target: make할 개체
    - Recipes: 주어진 타겟을 make할 때 실행할 명령어들의 나열, 명령어를 쓸 때 반드시 탭한번으로 들여쓰기를 해야 한다.
    - Prerequisites: 주어진 타겟을 make할 때 사용될 파일들의 목록.

2. Makefile 만들기

2. Makefile 만들기

STEP 1. 프로그램 빌드 도구인 make 설치

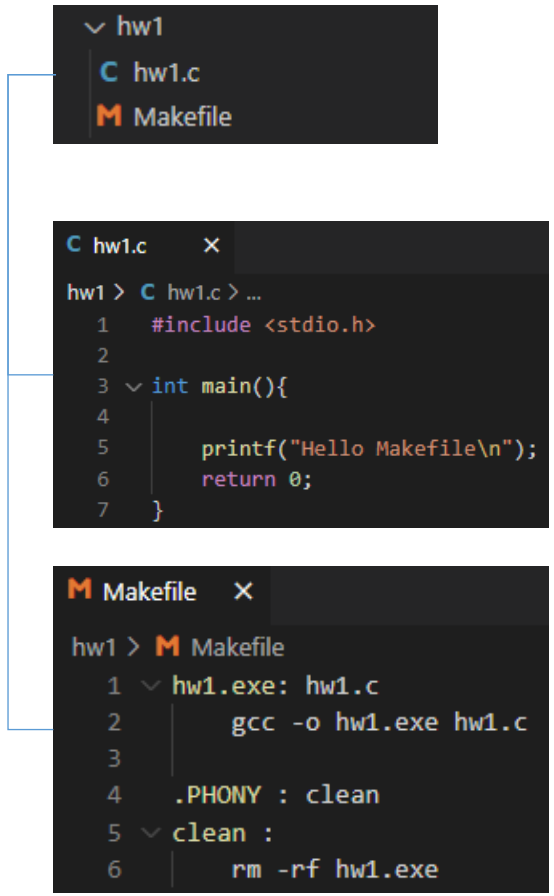
```
ing5uny@DESKTOP-79V69B3:~$ sudo apt-get install make
```

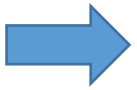
```
ing5uny@DESKTOP-79V69B3:~$ make --version
GNU Make 4.1
Built for x86_64-pc-linux-gnu
Copyright (C) 1988-2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
```

2. 1) 단일 .c

본인이 사용하는 텍스트 에디터로 Makefie 만들기

(프로젝트 디렉토리 안에 생성하세요.)





```
ing5uny@DESKTOP-79V69B3:~/hw1$ ls
Makefile  hw1.c
ing5uny@DESKTOP-79V69B3:~/hw1$ make
gcc -o hw1.exe hw1.c
ing5uny@DESKTOP-79V69B3:~/hw1$ ls
Makefile  hw1.c  hw1.exe
ing5uny@DESKTOP-79V69B3:~/hw1$ ./hw1.exe
Hello Makefile
```

Make 명령어를 통하여 hw1.exe 파일을 생성하여 실행해보는 예시

2. 2) flex와 bison

STEP 1. 본인이 사용하는 텍스트 에디터로 Makefie 만들기

(프로젝트 디렉토리 안에 생성하세요.)

```
ing5uny@DESKTOP-79V69B3:~$ cd compiler/
ing5uny@DESKTOP-79V69B3:~/compiler$ vim Makefile
```

```
1 wc :      wc.l wc.y
2       bison -d wc.y
3       flex wc.l
4       gcc -o $@ wc.tab.c lex.yy.c -ly -lfl
5
6 .PHONY : clean
7 clean :
8       rm -rf *.tab.c *.tab.h *.yy.c wc
```

2) flex와 bison

STEP 2. 프로그램 빌드

```
ing5uny@DESKTOP-79V69B3:~/compiler$ make
bison -d wc.y
flex wc.l
gcc -o wc wc.tab.c lex.yy.c -ly -lfl
ing5uny@DESKTOP-79V69B3:~/compiler$ ls
Makefile  lex.yy.c  wc  wc.l  wc.tab.c  wc.tab.h  wc.y
```

Makefile을 만들면 lex와 yacc의 빌드 과정이 make 명령어 하나로 가능하다.

2) flex와 bison

STEP 3. 빌드 결과물 삭제

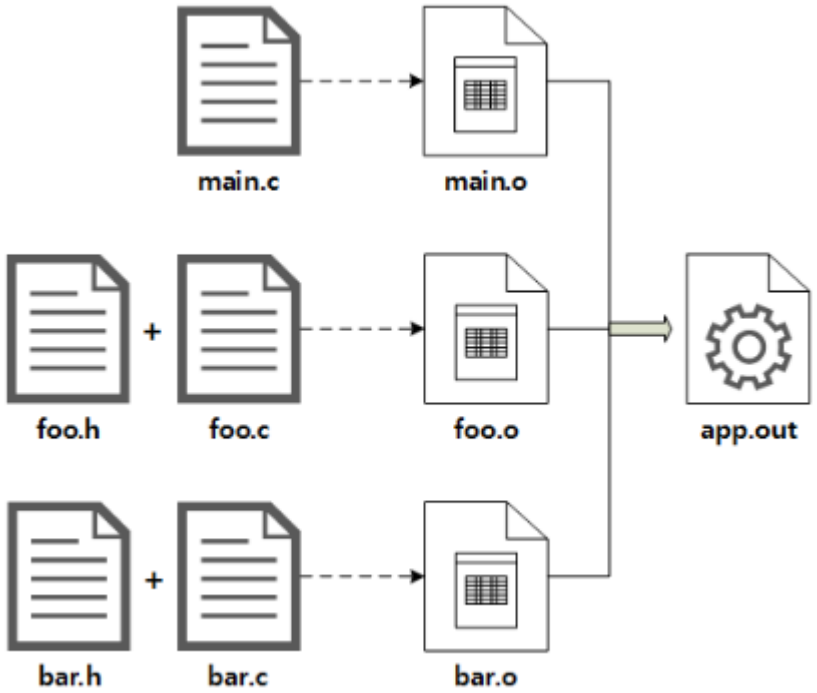
```
ing5uny@DESKTOP-79V69B3:~/compiler$ make clean
rm -rf *.tab.c *.tab.h *.yy.c wc
ing5uny@DESKTOP-79V69B3:~/compiler$ ls
Makefile  wc.l  wc.y
```

Clean을 활용하면 손쉽게 사용하지 않는 파일들을 간단하게 삭제할 수 있다.

2. 3) 여러 파일

본인이 사용하는 텍스트 에디터로 Makefie 만들기

https://www.tuwlab.com/ece/27193 참고.



빌드 상황 예

Makefile

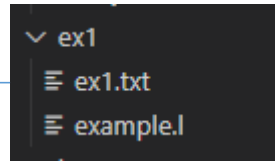
```
1  app.out: main.o foo.o bar.o
2      gcc -o app.out main.o foo.o bar.o
3
4  main.o: foo.h bar.h main.c
5      gcc -c -o main.o main.c
6
7  foo.o: foo.h foo.c
8      gcc -c -o foo.o foo.c
9
10 bar.o: bar.h bar.c
11     gcc -c -o bar.o bar.c
```

app.out파일을 생성하기 위한 Makefile

부록. <, > Redirection

부. <, > Redirection

<, > Redirection을 txt파일을 실행 파일의 입력으로 변경 할 수있다.



ex1.txt

```
Hi
Hello Compiler World!
124 4
```

example.l

```
%{
    #define LETTER 1
    #define DIGIT 2
%}

blank [ \t\n]+
letter [a-zA-Z]
digit [0-9]

%%
{blank} ;
{letter} {return LETTER;}
{digit} {return DIGIT;}
%%

int main(void)
{
    int tok;
    while((tok=yylex())!=0)
        if(tok==LETTER)
            printf("letter! \n");
        else printf("digit!\n");
}
```

```
ing5uny@DESKTOP-79V69B3:~/ex1$ flex example.l
ing5uny@DESKTOP-79V69B3:~/ex1$ ls
ex1.txt  example.l  lex.yy.c
ing5uny@DESKTOP-79V69B3:~/ex1$ cc lex.yy.c -lfl
ing5uny@DESKTOP-79V69B3:~/ex1$ ./a.out < ex1.txt
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
letter!
!digit!
digit!
digit!
digit!
ing5uny@DESKTOP-79V69B3:~/ex1$
```

a.out의 입력으로 ex1.txt를 리다이렉션하여 결과가 출력된 모습

end