

[13주차 1강] **파일 입출력(1)**

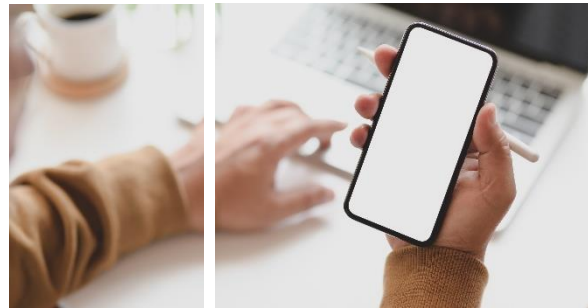
14.1 파일 입출력 개요





학습 목표

📍 14.1 파일의 개념을 이해한다.





1. 파일 입출력 개요

2. 파일 입출력 절차

3. 텍스트 파일 입출력

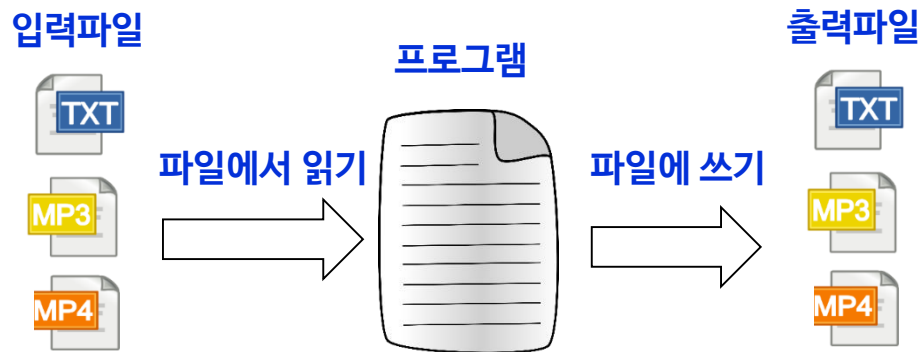
1. 파일 입출력 개요

지금까지 사용한 입출력 방식

- 키보드(표준입력장치)를 통해 데이터를 입력하고, 모니터(표준출력장치)로 결과 출력
- 프로그램 실행할 때마다 매번 입력해야 하고, 결과는 프로그램을 종료하면 사라짐

프로그램의 실행/종료 여부와 무관하게 데이터를 보존하고 싶다면??

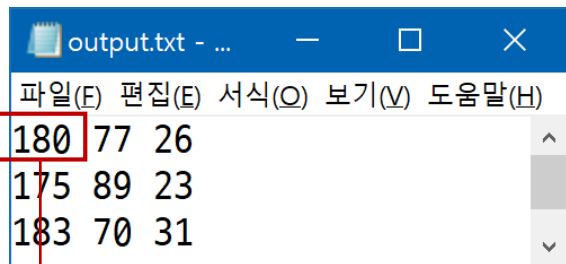
- 데이터를 파일로부터 입력받고, 결과를 파일로 저장 (**파일 입출력**)



1. 파일 입출력 개요

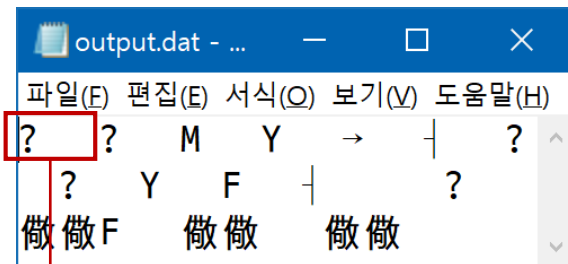
저장된 데이터 형식에 따른 파일 종류

● 텍스트 파일



0X 31 38 30
→ '1' '8' '0'

● 이진 파일



0X B4 00 00 00
→ 16x11 + 4
= 180

1. 파일 입출력 개요



텍스트 파일과 이진 파일 비교

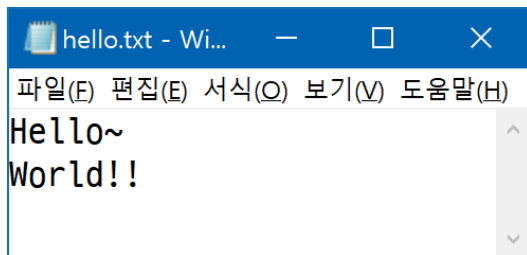
	텍스트(text) 파일	이진(binary) 파일
특성	<ul style="list-style-type: none">✓ 사람이 인식할 수 있는 문자를 담고 있는 파일✓ 특별한 응용 프로그램 없이 내용 볼 수 있음✓ 메모장 프로그램을 통해 파일을 열었을 때, 읽을 수 있는 문자들로 표현됨✓ 모든 데이터가 문자 형식으로 변환되어 기록됨 (예, 정수 10 → 문자 '1' '0' 으로 변환되어 저장)✓ 이진파일보다 공간 많이 차지하고, 입출력 속도 느림	<ul style="list-style-type: none">✓ 컴퓨터가 인식할 수 있는 데이터를 가진 파일✓ 특정 응용 프로그램을 이용해야 액세스할 수 있는 파일(예, ppt 파일)✓ 메모장 프로그램을 통해 파일을 열었을 때, 알아볼 수 없는 이상한 문자들로 표시✓ 데이터가 문자로 변환되지 않고 있는 그대로 저장 (예, 정수 10 → 10을 나타내는 이진 비트값 그대로 저장)✓ 텍스트 파일보다 저장 공간을 적게 차지하고, 입출력 속도 빠름



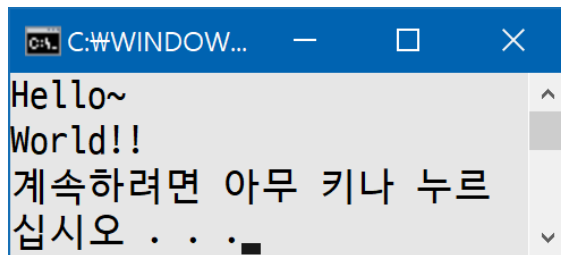
1. 파일 입출력 개요

파일(텍스트)과 화면 출력 비교

● 텍스트 파일



● 화면 출력



```
printf("Hello~\nWorld!!\n");
```

● 내부적으로는 둘 다 문자의 연속적인 나열 (일차원)

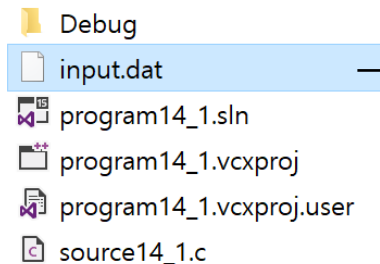
H	e	l	l	o	~	\n	W	o	r	l	d	!	!	\n
---	---	---	---	---	---	----	---	---	---	---	---	---	---	----



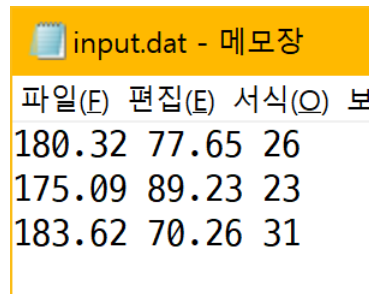
1. 파일 입출력 개요

파일 입출력 따라 해보기 (1)

- ✓ input.dat 파일로부터 키, 몸무게, 나이 정보를 입력받아
→ 이 내용을 그대로 output.dat 파일에 출력하기



〈현재 작업 폴더 내용 - 실행 전〉
?



메모장으로
작성

〈input.dat 파일의 내용〉



1. 파일 입출력 개요

파일 입출력 따라 해보기 (2)

- ✓ input.dat 파일로부터 키, 몸무게, 나이 정보를 입력받아
→ 이 내용을 그대로 output.dat 파일에 출력하기

(새로 생성)

Debug
input.dat
output.dat
program14_1.sln
program14_1.vcxproj
program14_1.vcxproj.user
source14_1.c

〈현재 작업 폴더 내용 - 실행 후〉

output.dat - 메모장

파일(F)	편집(E)	서식(O)	도움말(H)
180.32	77.65	26	
175.09	89.23	23	
183.62	70.26	31	

〈output.dat 파일의 내용〉



※ 실습하기



예제 프로그램 14.1

```
double height, weight;
int age, i;
FILE *fp1, *fp2;           // FILE 구조체 포인터 선언

fp1 = fopen("input.dat", "r"); // input.dat 파일 열기
fp2 = fopen("output.dat", "w"); // output.dat 파일 열기

for (i = 0; i < 3; i++) {
    fscanf(fp1, "%lf %lf %d", &height, &weight, &age); // 입력
    fprintf(fp2, "%.2f %.2f %d\n", height, weight, age); // 출력
}

fclose(fp1);                // input.dat 파일 닫기
fclose(fp2);                // output.dat 파일 닫기
```



1. 파일 입출력 개요



예제 프로그램 14.1과 표준 입출력 비교

- 입출력은 유사

```
fscanf(fp1, "%lf %lf %d", &height, &weight, &age);  
fprintf(fp2, "%.2f %.2f %d\n", height, weight, age);
```

- 파일을 열고 닫는 과정 필요

```
fp1 = fopen("input.dat", "r");  
fp2 = fopen("output.dat", "w");  
...  
fclose(fp1);  
fclose(fp2);
```



1. 파일 입출력 개요



입력 파일과 출력 파일은 어느 폴더에?

- 파일의 위치는 현재 작업 폴더 기준
- 작업 폴더란? 작업이 이루어지는 폴더
 - ✓ VS에서 실행하는 경우에는 프로젝트 폴더
 - ✓ 앞의 예제를 실행해서 확인해보자.
 - ✓ 실행 파일이 위치한 폴더
 - ✓ CMD 모드에서 확인해보자 (다음 슬라이드)



1. 파일 입출력 개요



CMD (명령프롬프트)

- 일종의 text 기반 사용자 interface (명령창에서 cmd 실행)
- 폴더 경로 및 파일 확인해보기
- 기본 명령어
 - ✓ cd : 폴더 이동을 위한 명령어 (cd 경로이름, cd \w, cd ..)
 - ✓ dir : 폴더의 파일 목록을 보여주는 명령어 (dir /w)
 - ✓ 탐색기와 비슷한 역할
- CMD에서 앞 예제 프로그램의 실행 파일을 실행시켜보자.





학습 정리

- **파일 입출력**이란 파일의 자료를 읽거나 파일로 저장하는 처리 방법을 말함
- **텍스트 파일**은 사람이 인식할 수 있는 문자를 담고 있는 파일로, 데이터가 텍스트로 변환되어 기록됨
- **이진 파일**은 컴퓨터 프로그램이 인식할 수 있는 데이터를 가진 파일로, 데이터가 컴퓨터 내부 표현 방식 그대로 기록됨
- **작업 폴더**란 프로그램의 작업이 이루어지는 폴더를 의미함

