

파일 입출력 (1)

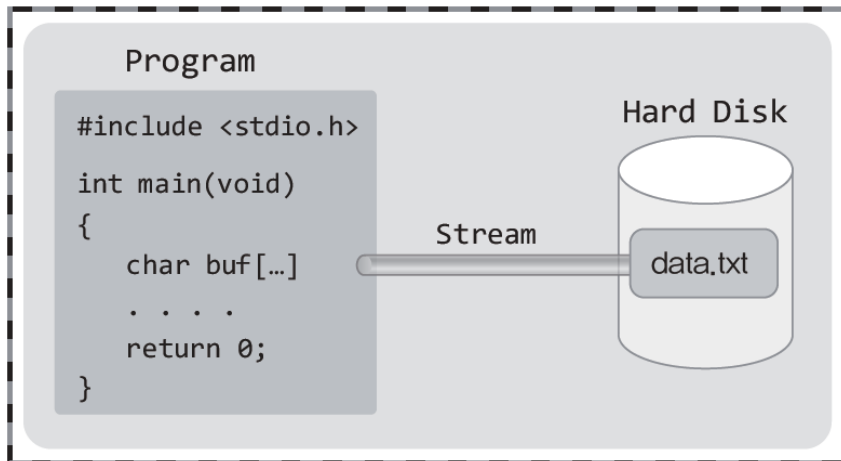
- 파일 스트림 생성과 분류 -

성공회대학교 IT융합자율학부
소프트웨어공학전공
홍 성 준

파일 스트림

◎ 스트림(stream)

- '데이터 이동의 경로가 되는 다리' 또는 '한 방향으로 흐르는 데이터 흐름'
 - 프로그램과 입력 장치 (stdin)
 - 프로그램과 출력 장치 (stdout, stderr)
- 콘솔 입출력과 마찬가지로 파일로부터의 데이터 입출력을 위한 스트림이 필요
 - 운영체제에 따라 파일의 구조가 결정되고 관리되기 때문에, 실제 파일 스트림 생성도 운영체제의 몫



파일 스트림

◎ 파일 스트림 생성

- fopen 함수를 호출하여 프로그램 상에서 파일 스트림을 생성

스트림을 형성할 파일의 이름

형성할 스트림의 종류

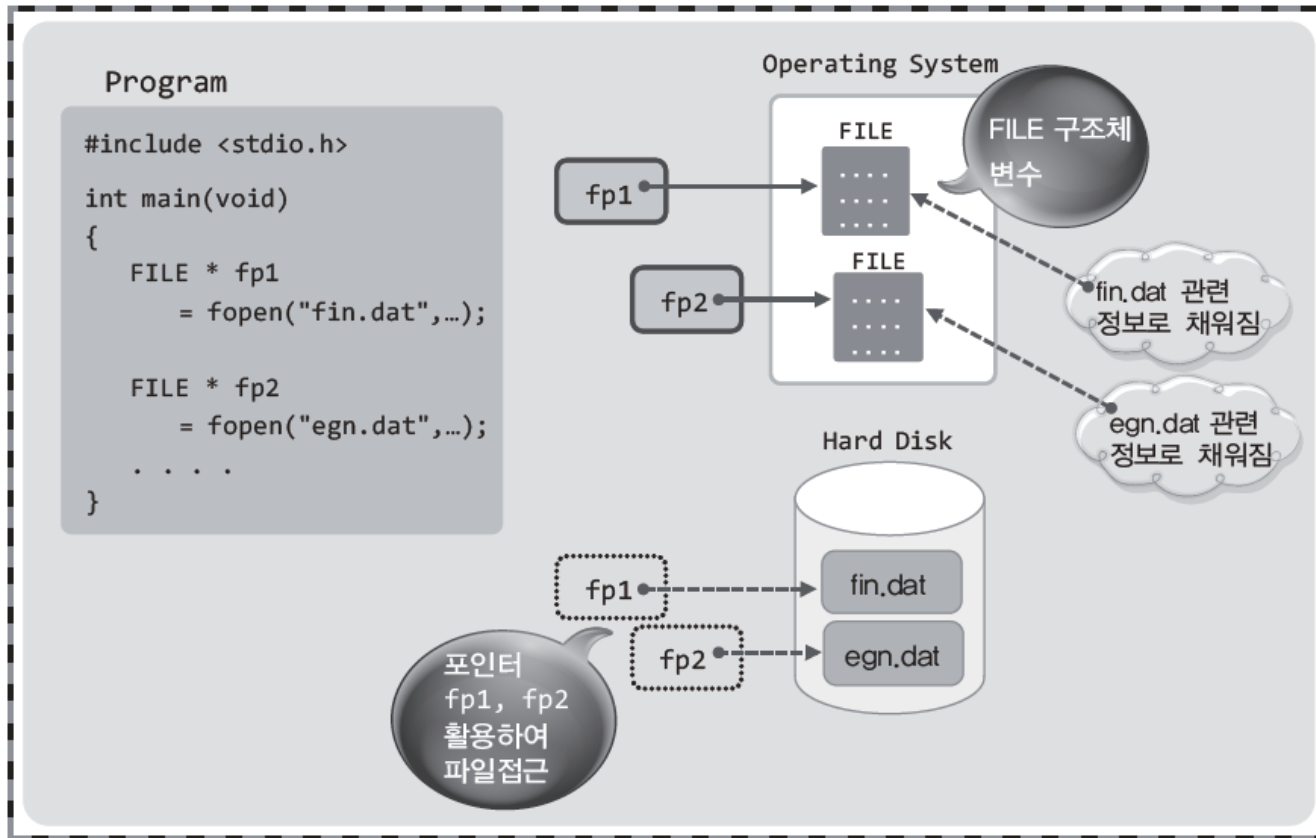
```
#include <stdio.h>
FILE * fopen(const char * filename, const char * mode);
```

➔ 성공 시 해당 파일의 FILE 구조체 변수의 주소 값, 실패 시 NULL 포인터 반환

- fopen 함수가 호출되면 FILE 구조체 변수가 생성
- 생성된 FILE 구조체 변수에는 파일에 대한 정보가 담겨 있음
- fopen 함수는 FILE 구조체 변수의 포인터를 반환하며 사실 상 파일을 가리키는 '지시자' 역할을 함

파일 스트림

◎ 파일 스트림 생성



파일 스트림

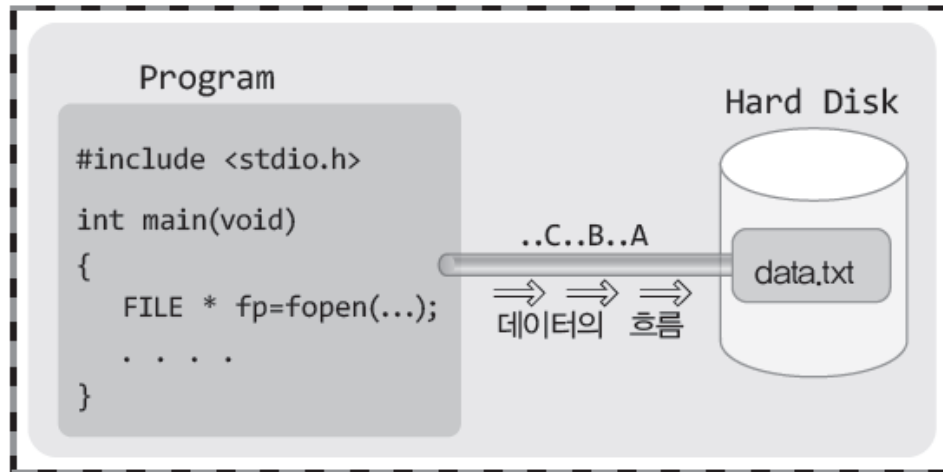
◎ 파일 입출력 스트림 생성: fopen 함수

- 출력 스트림

- 데이터를 파일에 쓰기 위한 스트림

```
FILE * fp = fopen("data.txt", "wt");
```

“파일 data.txt와 스트림을 형성하되 wt 모드로 스트림을 형성해라!”



- FILE 구조체 포인터 fp에 저장된 값이 파일 스트림에 데이터를 전송하는 도구가 됨

파일 스트림

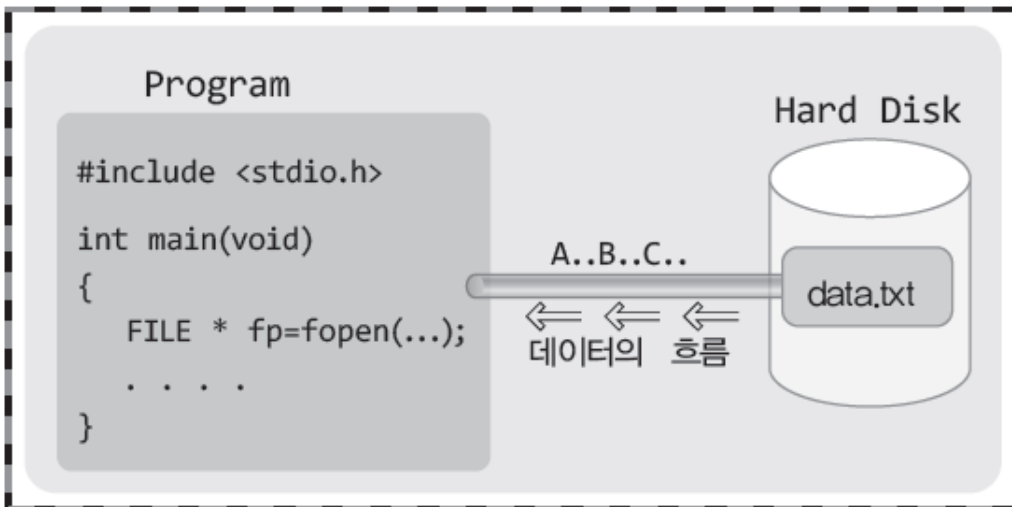
◎ 파일 입출력 스트림 생성: fopen 함수

- 입력 스트림

- 데이터를 파일로부터 읽어 들이기 위한 스트림

```
FILE * fp = fopen("data.txt", "rt");
```

“파일 data.txt와 스트림을 형성하되 rt 모드로 스트림을 형성해라!”



- FILE 구조체 포인터 fp에 저장된 값이 파일의 스트림으로부터 데이터를 수신하는 도구가 됨

파일 스트림

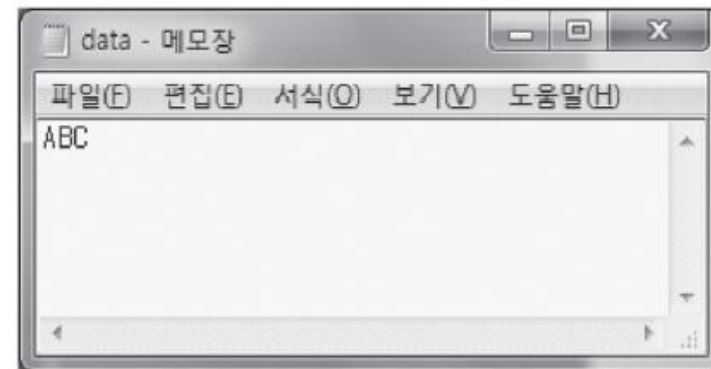
◎ 파일에 데이터 기록하기

```
int main(void)
{
    FILE * fp=fopen("data.txt", "wt");
    if(fp==NULL) {
        puts("파일오픈 실패!");
        return -1;    // 비정상적 종료를 의미하기 위해서 -1을 반환
    }

    fputc('A', fp);
    fputc('B', fp);
    fputc('C', fp);
    fclose(fp);    // 스트림의 종료
    return 0;
}
```

- fopen 함수 호출 시 절대경로를 명시할 수 있음

```
FILE * fp = fopen("C:\\Project\\data.txt", "wt");
```



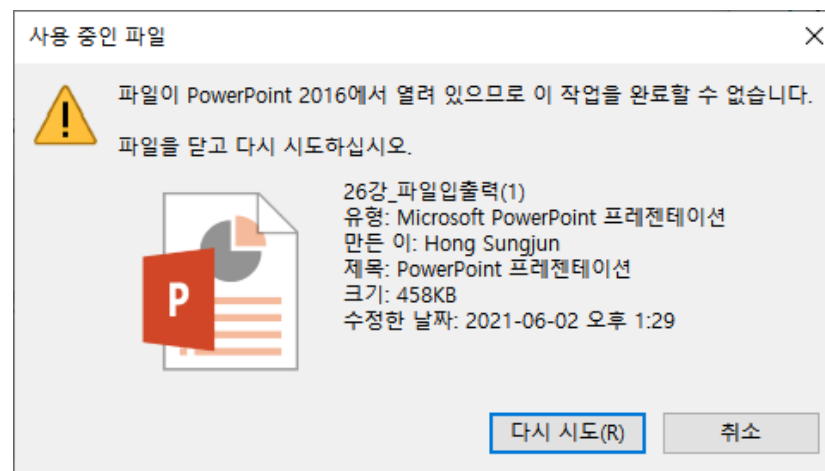
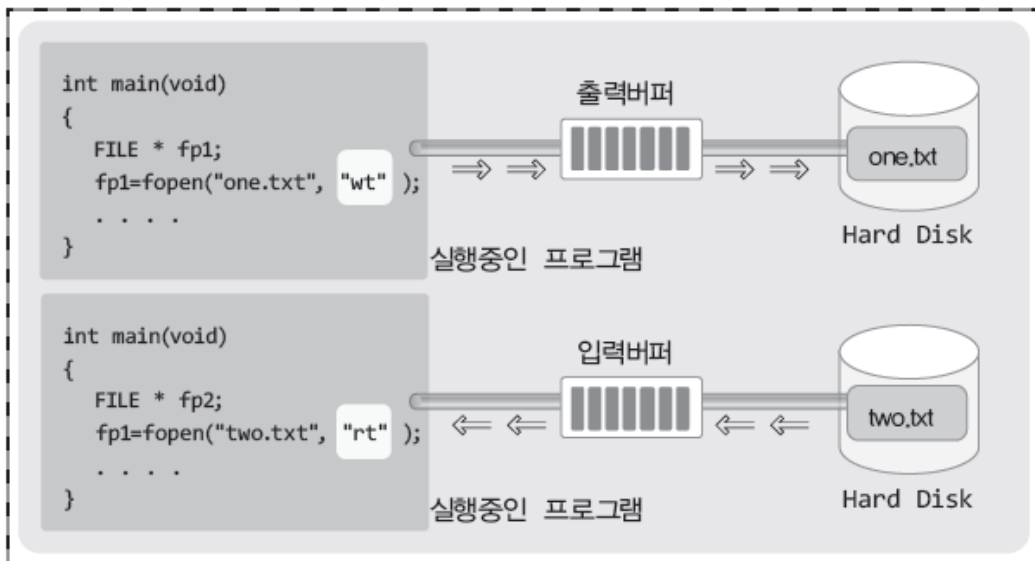
파일 스트림

◎ 파일 스트림 소멸시키기: fclose 함수

```
#include <stdio.h>
int fclose(FILE * stream);
```

➔ 성공 시 0, 실패 시 EOF를 반환

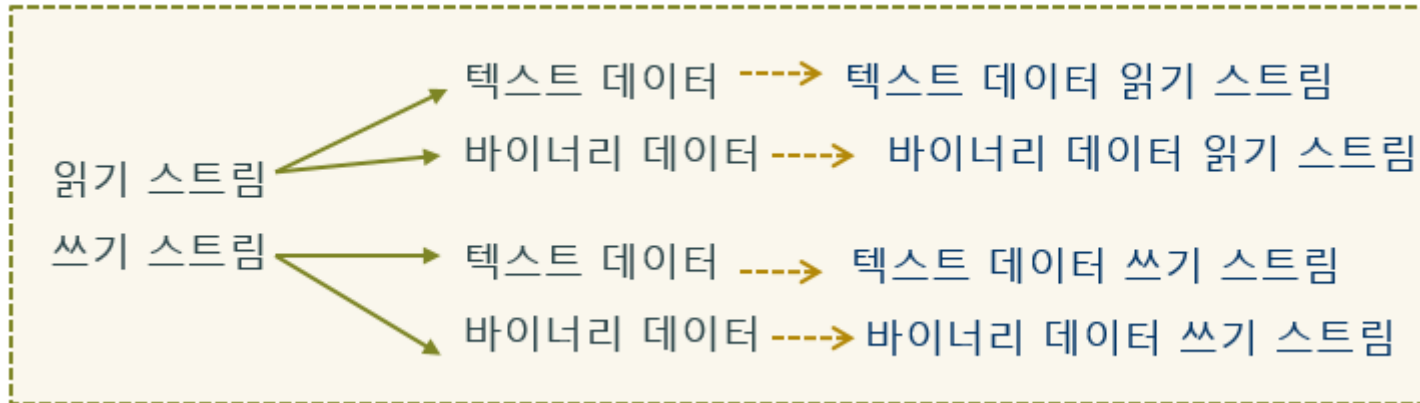
- 운영체제가 스트림에 할당한 자원을 반환하여 소멸시킴
- 출력 버퍼에 버퍼링된 데이터를 출력시킴



파일 열기 모드

◎ 파일 스트림 분류 기준

- 데이터의 입력 또는 데이터의 출력
 - 파일에서 데이터를 읽는데 사용하는 스트림과 파일에 데이터를 쓰는데 사용하는 스트림을 구분
- 텍스트 데이터 또는 바이너리 데이터
 - 입출력의 대상이 되는 데이터의 종류에 따라 스트림을 구분



파일 열기 모드

◎ 파일 스트림 분류 기준 1: 읽기 또는 쓰기

모드(mode)	스트림의 성격	파일이 없으면?
r	읽기 가능	에러
w	쓰기 가능	생성
a	파일의 끝에 덧붙여 쓰기 가능	생성
r+	읽기/쓰기 가능	에러
w+	읽기/쓰기 가능	생성
a+	읽기/덧붙여 쓰기 가능	생성

- 파일 모드에 'a'는 덧붙여 쓰기(append)가 가능한 스트림의 생성을 의미
 - 미리 기록된 파일의 EOF 다음부터 파일에 데이터를 기록
- 파일 모드에 '+'는 읽기, 쓰기가 모두 가능한 스트림의 생성을 의미
 - FP가 꼬일 위험이 있어 반드시 필요한 경우를 제외하곤 권장하지 않음

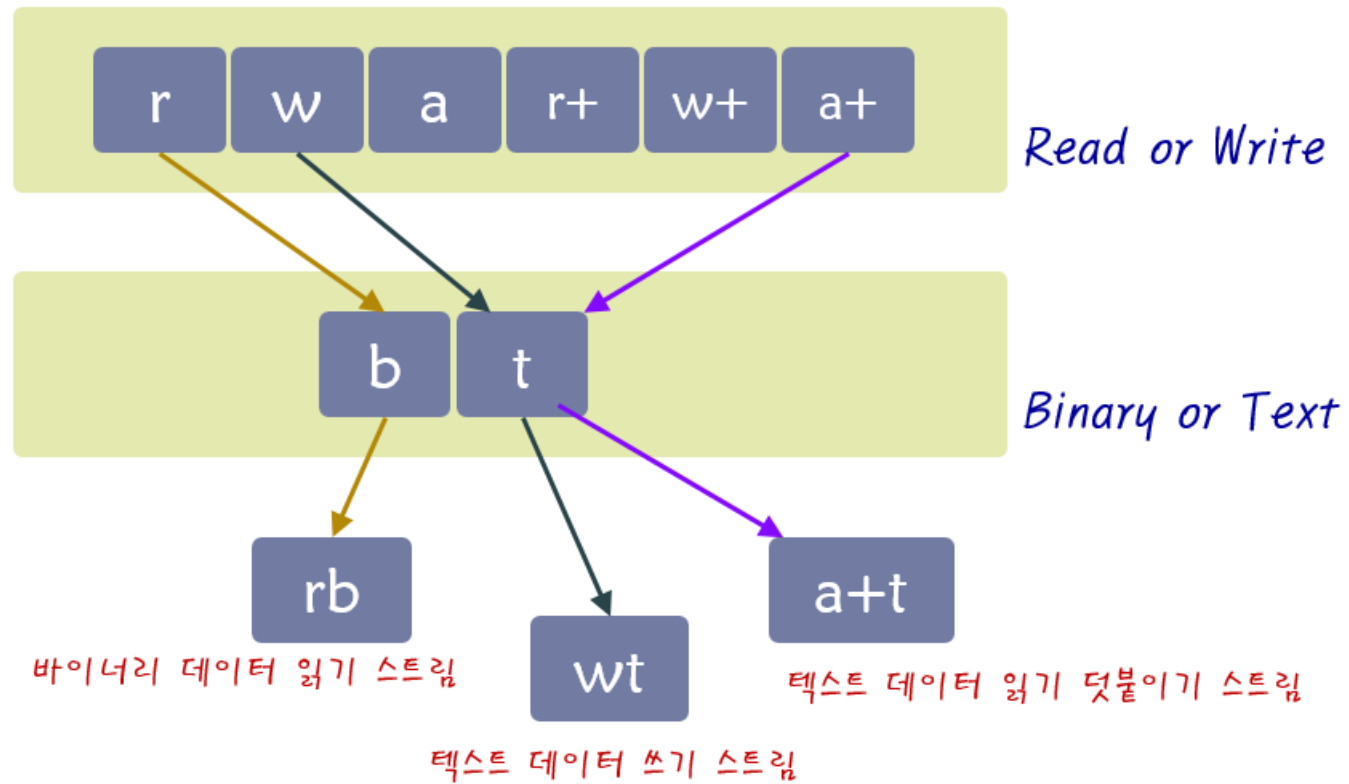
파일 열기 모드

◎ 파일 스트림 분류 기준 2: 텍스트 또는 바이너리

- 데이터의 종류에 따라 스트림을 구분
 - 텍스트 모드 스트림 (t) : 문자 데이터를 저장하는 스트림
 - 바이너리 모드 스트림 (b) : 바이너리 데이터를 저장하는 스트림
- 텍스트 데이터
 - 사람이 인식할 수 있는 유형의 문자로 이루어진 데이터
 - 도서 목록, 물품 가격, 전화번호, 주민등록번호
- 바이너리 데이터
 - 컴퓨터가 인식할 수 있는 유형의 데이터
 - 음악 파일, 이미지 파일 등

파일 열기 모드

◎ 파일 열기 모드의 조합



- 입출력 데이터 유형을 생략하면 텍스트 모드로 파일 스트림을 생성