

# File I/O

# File

➤ FILE \*fopen (const char \*filename, const char \*mode)

Mode	Description
"r"	읽기 모드로 파일을 연다.
"w"	쓰기 모드로 파일을 생성한다.
"a"	추가 모드로 파일을 연다.

```

~/FileIO
1 #include <stdio.h>
2
3 int main(void){
4     FILE *test;
5
6     test = fopen("test.txt", "r");
7
8     fclose(test);
9
10    return 0;
11 }
~
~
1,1    모두

```

# File

➤ 파일 입력 라이브러리 함수

종류	설명	입력 함수
문자 단위	문자 단위로 입력	<code>int fgetc(FILE *fp)</code>
문자열 단위	문자열 단위로 입력	<code>char *fgets(FILE *fp)</code>
서식화된 입출력	형식 지정 입력	<code>int fscanf(FILE *fp, ...)</code>
이진 데이터	이진 데이터 입력	<code>fread()</code>

# File

- size\_t fread (void \*buffer, size\_t size, size\_t count, FILE \* stream)
  - stream에서 buffer로 size\*count 만큼의 데이터를 복사
  - Return 값으로 실제 읽어 들인 데이터의 길이를 Return

```

~/FileIO
1 #include <stdio.h>
2 #define SIZE 100
3
4 int main(void){
5     FILE *test;
6     char text[SIZE];
7     int read_length = 0;
8
9     if((test = fopen("test.txt", "r")) != NULL){
10         read_length = fread(text, sizeof(char), SIZE, test);
11         printf("read length = %d \n", read_length);
12         printf("read string = %s", text);
13     }
14
15     fclose(test);
16
17     return 0;
18 }
  
```

# File

➤ int fseek (FILE \*stream, long offset, int origin)

- 대상 파일의 stream의 origin에서 부터 offset 만큼의 위치에 커서를 위치
- **origin**

상수	값	설명
SEEK_SET	0	파일의 시작
SEEK_CUR	1	현재 위치
SEEK_END	2	파일의 끝

# File

- int fseek (FILE \*stream, long offset, int origin)

```

~/FileIO
1 #include <stdio.h>
2 #define SIZE 100
3
4 int main(void){
5     FILE *test;
6     char text[SIZE];
7     char text2[SIZE];
8     int result = 0;
9     int read_length = 0;
10
11     if((test = fopen("test.txt", "r")) != NULL){
12         read_length = fread(text, sizeof(char), SIZE, test);
13         printf("read length = %d \n", read_length);
14         printf("read string = %s \n", text);
15
16         result = fseek(test, 10L, SEEK_SET);
17         read_length = fread(text2, sizeof(char), SIZE, test);
18         printf("read length = %d \n", read_length);
19         printf("read string = %s\n", text2);
20     }
21     fclose(test);
22
23     return 0;
24 }
25 }

```

# File

➤ long ftell (FILE \*stream)

- 대상 파일 stream의 현재 커서까지의 크기를 반환

```

~/FileIO
1 #include <stdio.h>
2 #define SIZE 100
3
4 int main(void){
5     FILE *test;
6     int result = 0;
7     long len = 0;
8
9     if((test = fopen("test.txt", "r")) != NULL){
10        len = ftell(test);
11        printf("1 : %ld \n", len);
12        result = fseek(test, 10L, SEEK_SET);
13        len = ftell(test);
14        printf("2 : %ld \n", len);
15        result = fseek(test, 0, SEEK_END);
16        len = ftell(test);
17        printf("3 : %ld \n", len);
18    }
19
20    fclose(test);
21
22    return 0;
23 }
~
~
"file_ftell.c" 23L, 404C
17, 3-9
모 두

```