FILE * is a pointer to a FILE data type

ALWAYS save
the return value
from fopen()
into a FILE * variable

fopen()
Purpose: open a file.
Parameters: fopen(char *filename, char *access);
Return: FILE *fp (upon successful) NULL if not successful
Access Code: string "r" - read, "rb" - read in binary

"w" - write, "wb" - write in binary (overwrites a file)

"a" - append (does not overwrite)

:

```
FILE * fp;

fp = fopen ("file.txt", "w");

if (fp == NULL)
{
    printf("file.txt does not exist, exiting program"
    return ERROR;
}
```

Any read or write operation generally happens in a loop.

For instance, as long as we are not at the end of the file, keep reading / keep writing.

while (!feof(ifp))

fread() : read binary data
Prototype:
size_t fread(void *a_ptr, size_t element_size, size_t n, FILE *fp);

- **ptr** This is the pointer to a block of memory with a minimum size of *size*nmemb* bytes.
- **size** This is the **size in bytes** of each element to be read.
- **nmemb** This is the number of elements, each one with a size of **size** bytes.
- **stream** This is the pointer to a FILE object that specifies an input stream.

Return Value
The total number of elements successfully read are returned as a size_t object

```
#define kBuf 100

FILE* ifp = NULL
// fopen code
char buffer[kBuf0 = {};

fread(buffer, sizeof(char), kBuf, ifp);
```

fwrite() : used to write up to count items, each of size bytes
Prototype : size_t fwrite(const void *ptr, size_t size, size_t nmemb, FILE *stream
Return Value : return the number of full items succesfully written

- **ptr** This is the pointer to the array of elements to be written.
- **size** This is the size in bytes of each element to be written.
- **nmemb** This is the number of elements, each one with a size of **size** bytes.
- **stream** This is the pointer to a FILE object that specifies an output stream.

```
FILE *ofp = NULL
// fopen code to open this file for writing

fwrite(buffer, sizeof(char), kBuf, ofp):
```

fgets()
Reads a line of text until a '\n' is seen or the (numOfCharacters-1) is read. The read string is stored in the string argument given.
Prototype: fgets(char* string, int numOfCharacters, FILE* stream);
Return Value: a pointer to the string argument.

fclose(): used to close files opened with fopen()
prototype: fclose(fp);
parameters: file pointer
return value: 0 (false) if opened file was successfully closed else true

```
if (fclose(fp) != 0)
{
    //error handling code here
}
```