**Binary VS Character Streams**

Differences between a binary and a character or text stream:

1. The data from a text file is divided into lines by \n and each can have at most 254 characters (in most systems). In case of binary stream, it is a single string of values.
2. Character streams can only work with ASCII codes.
3. In some systems, spaces following a (newline) "\n" is omitted in character streams.
4. In character streams there is one-to-one correspondence between each character read or written.

Using binary stream, only the following functions are different:

size\_t fread(void \*buffer, size\_t size, size\_t num, FILE \*fp);  
size\_t fwrite(void \*buffer, size\_t size, size\_t num, FILE \*fp);  
size\_t fread *// returns number of inputs*size\_t fwrite *// returns number of outputs*void \*buffer *// an address to a memory (without a specific data type)*size\_t size *// size of byte being read*size\_t num *// number of inputs/outputs*

C

#include<stdio.h>  
#include<stdlib.h>  
  
int main()  
{  
 FILE \*fp; int i;  
 if((fp = fopen("myfile", "wb")) == **NULL**)  
 {  
 printf("Error");  
 exit(1);  
 }  
 i = 100;  
 if(fwrite(&i, 2, 1, fp) != 1)

*//take 1 input of size 2 bytes; if not possible print error* printf("Write error"); exit(1);  
 close(fp);  
  
 if((fp = fopen("myfile", "rb")) == **NULL**)  
 {  
 printf("Error");  
 exit(1);  
 }  
 if(fread(&i, 2, 1, fp) != 1)

*//give 1 output of size 2 bytes; if not possible print error* printf("Read error"); exit(1);  
 printf("%d", i); close(fp);  
}

C