**File Input/Output in C**

A **file** represents a sequence of bytes on the disk where a group of related data is stored. File is created for permanent storage of data. It is a ready made structure.

In C language, we use a structure **pointer of file type** to declare a file.

FILE \*fp;

**Opening a File or Creating a File**

The fopen() function is used to create a new file or to open an existing file.

**General Syntax:**

FILE \*fp = fopen(const char \*filename, const char \*mode);

Here, \*fp is the FILE pointer (FILE \*fp), which will hold the reference to the opened(or created) file.

**filename** is the name of the file to be opened and **mode** specifies the purpose of opening the file. Mode can be of following types,

|  |  |
| --- | --- |
| **mode** | **description** |
| r | opens a text file in reading mode |
| w | opens or create a text file in writing mode. |

**Closing a File**

The fclose() function is used to close an already opened file.

**Program to write content to a file**

Create a file one.txt in same folder as your program exist.

#include <stdio.h>

void main()

{

**FILE \*fptr;**

char name[20];

int age;

float salary;

**fptr = fopen ("one.txt", "w");** /\* open for writing\*/

if (fptr == NULL)

{

printf("File does not exists \n");

return;

}

printf("Enter the name \n");

scanf("%s", name);

**fprintf(fptr, "Name = %s\n", name);**

printf("Enter the age \n");

scanf("%d", &age);

**fprintf(fptr, "Age = %d\n", age);**

printf("Enter the salary \n");

scanf("%f", &salary);

**fprintf(fptr, "Salary = %.2f\n", salary);**

**fclose(fptr);**

}

**Program to read content from a txt file**

#include <stdio.h>

#include <stdlib.h>

void main()

{

**FILE \*fptr;**

char filename[15];

char ch;

printf("Enter the filename to be opened \n");

scanf("%s", filename);

/\* open the file for reading \*/

**fptr = fopen(filename, "r");**

if (fptr == NULL)

{

printf("Cannot open file \n");

exit(0);

}

**ch = fgetc(fptr);**

while (ch != EOF)

{

printf ("%c", ch);

**ch = fgetc(fptr);**

}

**fclose(fptr);**

}