**Write a program in C to create and store information in a text file.**

#include <stdio.h>

#include <stdlib.h>

main()

{

char str[1000];

FILE \*fptr;

char fname[20]="test.txt";

printf("\n\n Create a file (test.txt) and input text :\n");

printf("----------------------------------------------\n");

fptr=fopen(fname,"w");

if(fptr==NULL)

{

printf(" Error in opening file!");

exit(1);

}

printf(" Input a sentence for the file : ");

fgets(str, sizeof str, stdin);

fprintf(fptr,"%s",str);

fclose(fptr);

printf("\n The file %s created successfully...!!\n\n",fname);

}

**Write a program in C to read an existing file.**

#include <stdio.h>

#include <stdlib.h>

main()

{

FILE \*fptr;

char fname[20];

char str;

printf("\n\n Read an existing file :\n");

printf("------------------------------\n");

printf(" Input the filename to be opened : ");

scanf("%s",fname);

fptr = fopen (fname, "r");

if (fptr == NULL)

{

printf(" File does not exist or cannot be opened.\n");

exit(0);

}

printf("\n The content of the file %s is :\n",fname);

str = fgetc(fptr);

while (str != EOF)

{

printf ("%c", str);

str = fgetc(fptr);

}

fclose(fptr);

printf("\n\n");

}

**Write a program in C to count the number of words and characters in a file**

#include <stdio.h>

#include <stdlib.h>

void main()

{

FILE \*fptr;

char ch;

int wrd=1,charctr=1;

char fname[20];

printf("\n\n Count the number of words and characters in a file :\n");

printf("---------------------------------------------------------\n");

printf(" Input the filename to be opened : ");

scanf("%s",fname);

fptr=fopen(fname,"r");

if(fptr==NULL)

printf(" File does not exist or can not be opened.");

else

{

ch=fgetc(fptr);

printf(" The content of the file %s are : ",fname);

while(ch!=EOF)

{

printf("%c",ch);

if(ch==' '||ch=='\n')

wrd++;

else

charctr++;

ch=fgetc(fptr);

}

printf("\n The number of words in the file %s are : %d\n",fname,wrd-2);

printf(" The number of characters in the file %s are : %d\n\n",fname,charctr-1);

}

fclose(fptr);

}

**Write a program in C to copy a file to another name.**

#include <stdio.h>

#include <stdlib.h>

void main()

{

FILE \*fptr1, \*fptr2;

char ch, fname1[20], fname2[20];

printf("\n\n Copy a file in another name :\n");

printf("----------------------------------\n");

printf(" Input the source file name : ");

scanf("%s",fname1);

fptr1=fopen(fname1, "r");

if(fptr1==NULL)

{

printf(" File does not found or error in opening.!!");

exit(1);

}

printf(" Input the new file name : ");

scanf("%s",fname2);

fptr2=fopen(fname2, "w");

if(fptr2==NULL)

{

printf(" File does not found or error in opening.!!");

fclose(fptr1);

exit(2);

}

while(1)

{

ch=fgetc(fptr1);

if(ch==EOF)

{

break;

}

else

{

fputc(ch, fptr2);

}

}

printf(" The file %s copied successfully in the file %s. \n",fname1,fname2);

fclose(fptr1);

fclose(fptr2);

getchar();

}

**Write a program in C to remove a file from the disk.**

#include <stdio.h>

void main()

{

int status;

char fname[20];

printf("\n\n Remove a file from the disk :\n");

printf("----------------------------------\n");

printf(" Input the name of file to delete : ");

scanf("%s",fname);

status=remove(fname);

if(status==0)

{

printf(" The file %s is deleted successfully..!!\n\n",fname);

}

else

{

printf(" Unable to delete file %s\n\n",fname);

}

}

**Write a program in C to delete a specific line from a file.**

**#include <stdio.h>**

#include <string.h>

#define MAX 256

main()

{

int lno, ctr = 0;

char ch;

FILE \*fptr1, \*fptr2;

char fname[MAX];

char str[MAX], temp[] = "temp.txt";

printf("\n\n Delete a specific line from a file :\n");

printf("-----------------------------------------\n");

printf(" Input the file name to be opened : ");

scanf("%s",fname);

fptr1 = fopen(fname, "r");

if (!fptr1)

{

printf(" File not found or unable to open the input file!!\n");

return 0;

}

fptr2 = fopen(temp, "w"); // open the temporary file in write mode

if (!fptr2)

{

printf("Unable to open a temporary file to write!!\n");

fclose(fptr1);

return 0;

}

printf(" Input the line you want to remove : ");

scanf("%d", &lno);

lno++;

// copy all contents to the temporary file except the specific line

while (!feof(fptr1))

{

strcpy(str, "\0");

fgets(str, MAX, fptr1);

if (!feof(fptr1))

{

ctr++;

/\* skip the line at given line number \*/

if (ctr != lno)

{

fprintf(fptr2, "%s", str);

}

}

}

fclose(fptr1);

fclose(fptr2);

remove(fname); // remove the original file

rename(temp, fname); // rename the temporary file to original name

/\*------ Read the file ----------------\*/

fptr1=fopen(fname,"r");

ch=fgetc(fptr1);

printf(" Now the content of the file %s is : \n",fname);

while(ch!=EOF)

{

printf("%c",ch);

ch=fgetc(fptr1);

}

fclose(fptr1);

/\*------- End of reading ---------------\*/

return 0;

}

## Introduction

In this series of C Projects Source Code, we'll look at how to build an Employee Management System in C. We may manage the information of workers working in a firm or organization using this Employee Management System. The file handling technique is used here to save the data in a particular file, and you get the notion of this project as soon as you hear the name.

This project uses the Insert, Edit, and Delete file actions, but the sole constraint is that you can only display the data, not search for any data item in particular. If you have more experience with C, you may alter this program by using the searching strategies.

The source code is available for download under the button below the picture; you may use it to obtain a better understanding of the project.

The following modules are included in this project.

* Add Employee Details
* Edit Employee details
* Modify Employee
* Delete Employee

**#include <stdio.h> ///for input output functions like printf, scanf**

**#include <stdlib.h>**

**#include <conio.h>**

**#include <windows.h> ///for windows related functions (not important)**

**#include <string.h> ///string operations**

**/\*\* List of Global Variable \*/**

**COORD coord = {0,0}; /// top-left corner of window**

**/\*\***

**function : gotoxy**

**@param input: x and y coordinates**

**@param output: moves the cursor in specified position of console**

**\*/**

**void gotoxy(int x,int y)**

**{**

**coord.X = x;**

**coord.Y = y;**

**SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE),coord);**

**}**

**/\*\* Main function started \*/**

**int main()**

**{**

**FILE \*fp, \*ft; /// file pointers**

**char another, choice;**

**/\*\* structure that represent a employee \*/**

**struct emp**

**{**

**char name[40]; ///name of employee**

**int age; /// age of employee**

**float bs; /// basic salary of employee**

**};**

**struct emp e; /// structure variable creation**

**char empname[40]; /// string to store name of the employee**

**long int recsize; /// size of each record of employee**

**/\*\* open the file in binary read and write mode**

**\* if the file EMP.DAT already exists then it open that file in read write mode**

**\* if the file doesn't exit it simply create a new copy**

**\*/**

**fp = fopen("EMP.DAT","rb+");**

**if(fp == NULL)**

**{**

**fp = fopen("EMP.DAT","wb+");**

**if(fp == NULL)**

**{**

**printf("Connot open file");**

**exit(1);**

**}**

**}**

**/// sizeo of each record i.e. size of structure variable e**

**recsize = sizeof(e);**

**/// infinite loop continues untile the break statement encounter**

**while(1)**

**{**

**system("cls"); ///clear the console window**

**gotoxy(30,10); /// move the cursor to postion 30, 10 from top-left corner**

**printf("1. Add Record"); /// option for add record**

**gotoxy(30,12);**

**printf("2. List Records"); /// option for showing existing record**

**gotoxy(30,14);**

**printf("3. Modify Records"); /// option for editing record**

**gotoxy(30,16);**

**printf("4. Delete Records"); /// option for deleting record**

**gotoxy(30,18);**

**printf("5. Exit"); /// exit from the program**

**gotoxy(30,20);**

**printf("Your Choice: "); /// enter the choice 1, 2, 3, 4, 5**

**fflush(stdin); /// flush the input buffer**

**choice = getche(); /// get the input from keyboard**

**switch(choice)**

**{**

**case '1': /// if user press 1**

**system("cls");**

**fseek(fp,0,SEEK\_END); /// search the file and move cursor to end of the file**

**/// here 0 indicates moving 0 distance from the end of the file**

**another = 'y';**

**while(another == 'y') /// if user want to add another record**

**{**

**printf("\nEnter name: ");**

**scanf("%s",e.name);**

**printf("\nEnter age: ");**

**scanf("%d", &e.age);**

**printf("\nEnter basic salary: ");**

**scanf("%f", &e.bs);**

**fwrite(&e,recsize,1,fp); /// write the record in the file**

**printf("\nAdd another record(y/n) ");**

**fflush(stdin);**

**another = getche();**

**}**

**break;**

**case '2':**

**system("cls");**

**rewind(fp); ///this moves file cursor to start of the file**

**while(fread(&e,recsize,1,fp)==1) /// read the file and fetch the record one record per fetch**

**{**

**printf("\n%s %d %.2f",e.name,e.age,e.bs); /// print the name, age and basic salary**

**}**

**getch();**

**break;**

**case '3': /// if user press 3 then do editing existing record**

**system("cls");**

**another = 'y';**

**while(another == 'y')**

**{**

**printf("Enter the employee name to modify: ");**

**scanf("%s", empname);**

**rewind(fp);**

**while(fread(&e,recsize,1,fp)==1) /// fetch all record from file**

**{**

**if(strcmp(e.name,empname) == 0) ///if entered name matches with that in file**

**{**

**printf("\nEnter new name,age and bs: ");**

**scanf("%s%d%f",e.name,&e.age,&e.bs);**

**fseek(fp,-recsize,SEEK\_CUR); /// move the cursor 1 step back from current position**

**fwrite(&e,recsize,1,fp); /// override the record**

**break;**

**}**

**}**

**printf("\nModify another record(y/n)");**

**fflush(stdin);**

**another = getche();**

**}**

**break;**

**case '4':**

**system("cls");**

**another = 'y';**

**while(another == 'y')**

**{**

**printf("\nEnter name of employee to delete: ");**

**scanf("%s",empname);**

**ft = fopen("Temp.dat","wb"); /// create a intermediate file for temporary storage**

**rewind(fp); /// move record to starting of file**

**while(fread(&e,recsize,1,fp) == 1) /// read all records from file**

**{**

**if(strcmp(e.name,empname) != 0) /// if the entered record match**

**{**

**fwrite(&e,recsize,1,ft); /// move all records except the one that is to be deleted to temp file**

**}**

**}**

**fclose(fp);**

**fclose(ft);**

**remove("EMP.DAT"); /// remove the orginal file**

**rename("Temp.dat","EMP.DAT"); /// rename the temp file to original file name**

**fp = fopen("EMP.DAT", "rb+");**

**printf("Delete another record(y/n)");**

**fflush(stdin);**

**another = getche();**

**}**

**break;**

**case '5':**

**fclose(fp); /// close the file**

**exit(0); /// exit from the program**

**}**

**}**

**return 0;**

**}**