

PPS  
Programming for Problem  
Solving

Mini Project

Paridhi Agarwal

RA2111019010048

## Problem Statement

Write a program in C to make a database for employees which allows the user to add, list, modify and delete records.

### Analysis:

- Accept the User Choice
- Accept input according to user choice
- Store and display the records in a file

## FDT-Function Description Table

Function Name	Return Type	Purpose	Parameter List
main	int	To take user's inputs	-
gotoxy	void	To set the coordinates of the cursor on the console	int x, int y

# Algorithm

I) void gotoxy (int x, int y):

1. Start
2. Set the location of the x-coordinate
3. Set the location of the y-coordinate
4. Set the console cursor position
5. Stop

II) int main ():

1. Start
2. Open database file
3. Accept user choice
4. Repeat the following steps:
  - If the user chooses to add a record, accept the name, age and basic salary of the employee and save it in the database
  - If the user chooses to list records, display all the data
  - If the user chooses to modify a record, accept the name and retake the input and replace it in the database
  - If the user chooses to delete a record, accept the name and remove it from the database
  - If the user chooses to exit, terminate the program
5. Stop

# Source Code

[illegible]

# Database Management

Program allows to add, list, modify and delete records from the database

Paridhi Agarwal RA2111019010048

\*\*\*\*\*  
 \*\*\*\*\*/

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <conio.h>
```

```
#include <windows.h>
```

```
#include <string.h>
```

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

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

$$\{$$
$$\text{coord.X} = \mathbf{x};$$
$$\text{coord.Y} = y;$$

```
SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE),coord);
```

$$\}$$

```
int main()
```

 $\{$ 

FILE \*fp, \*ft;

```
char another, choice;
```

```
struct emp
```

$$\{$$

```
char name[40];
```

```
int age;
```

```
float bs;
```

$$\};$$

```
struct emp e;
```

```
char empname[40];
long int recsize;
fp = fopen("EMP.DAT","rb+");
if(fp == NULL)
{
    fp = fopen("EMP.DAT","wb+");
    if(fp == NULL)
    {
        printf("Connot open file");
        exit(1);
    }
}
recsize = sizeof(e);
while(1)
{
    system("cls");
    gotoxy(30,10);
    printf("1. Add Record");
    gotoxy(30,12);
    printf("2. List Records");
    gotoxy(30,14);
    printf("3. Modify Records");
    gotoxy(30,16);
    printf("4. Delete Records");
    gotoxy(30,18);
    printf("5. Exit");
    gotoxy(30,20);
    printf("Your Choice: ");
    fflush(stdin);
    choice = getche();
}
```

```

switch(choice)
{
case '1':
    system("cls");
    fseek(fp,0,SEEK_END);
    another = 'y';
    while(another == 'y')
    {
        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);
        printf("\nAdd another record(y/n) ");
        fflush(stdin);
        another = getche();
    }
    break;
case '2':
    system("cls");
    rewind(fp);
    while(fread(&e,recsize,1,fp)==1)
    {
        printf("\n%s %d %.2f",e.name,e.age,e.bs);
    }
    getch();
    break;
case '3':

```

```

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)
    {
        if(strcmp(e.name,empname) == 0)
        {
            printf("\nEnter new name,age and bs: ");
            scanf("%s%d%f",e.name,&e.age,&e.bs);
            fseek(fp,-recsize,SEEK_CUR);
            fwrite(&e,recsize,1,fp);
            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");
    rewind(fp); /// move record to starting of file
    while(fread(&e, recsize, 1, fp) == 1)
    {
        if(strcmp(e.name, empname) != 0)
        {
            fwrite(&e, recsize, 1, ft);
        }
    }
    fclose(fp);
    fclose(ft);
    remove("EMP.DAT");
    rename("Temp.dat", "EMP.DAT");
    fp = fopen("EMP.DAT", "rb+");
    printf("Delete another record(y/n)");
    fflush(stdin);
    another = getche();
}
break;
case '5':
    fclose(fp);
    exit(0);
}
}
return 0;
}

```



# VDT-Variable Description Table

## Sample Input Output

Variable Name	Data Type	Purpose	Scope
coord	COORD	To set the coordinates of the console	int main ()
x	int	To store the x-coordinate	int main ()
y	int	To store the y-coordinate	int main ()
*fp	FILE	File Pointers	int main ()
*ft	FILE	File Pointers	int main ()
another	char	To store confirm the choice	int main ()
choice	char	To store the user choice	int main ()
name[40]	char	To store the employee name	int main ()
age	int	To store the employee age	int main ()
bs	float	To store the employee basic salary	int main ()
e	struct	To access structure variables	int main ()
empname[40]	char	To store the employee name	int main ()
recsize	long int	To store the length of each record	int main ()

# Sample Input Output

```
1. Add Record
2. List Records
3. Modify Records
4. Delete Records
5. Exit
Your Choice: _
```

```
Enter name: Paridhi
Enter age: 18
Enter basic salary: 200000
Add another record(y/n) y
Enter name: Aditya
Enter age: 18
Enter basic salary: 300000
Add another record(y/n) y
Enter name: Abhinav
Enter age: 17
Enter basic salary: 100000
Add another record(y/n) y
Enter name: Nishtha
Enter age: 18
Enter basic salary: 150000
Add another record(y/n) y
Enter name:
```

```
Paridhi 18 200000.00
Aditya 18 300000.00
Abhinav 17 100000.00
Nishtha 18 150000.00
Hannah 18 250000.00
```

```
Enter new name,age and bs: Paridhi
19
3000000
Modify another record(y/n)
```

```
Enter new name,age and bs: Paridhi
19
3000000
Modify another record(y/n)
```

```
Paridhi 19 3000000.00
Aditya 18 300000.00
Abhinav 17 100000.00
Nishtha 18 150000.00
```

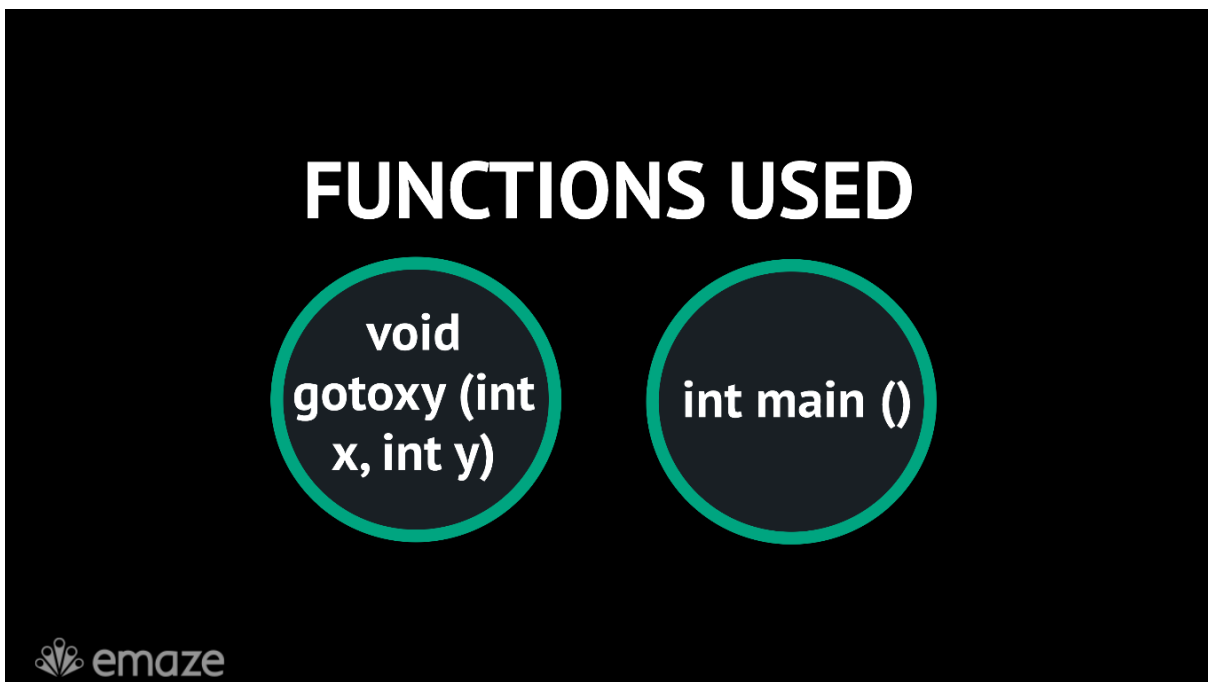
# Presentation



A presentation slide with a dark background. It features a faint world map and several large, overlapping plus signs in light blue, teal, and purple. The text 'DATABASE MANGANEMENT RECORDING EMPLOYEE DATA' is written in white, bold, uppercase letters. The 'emaze' logo is in the bottom left corner.

**DATABASE  
MANGANEMENT  
RECORDING EMPLOYEE DATA**

 emaze




A presentation slide with a dark background. It features two large, overlapping circles with teal outlines. The left circle contains the text 'void gotoxy (int x, int y)' and the right circle contains 'int main ()'. The text 'FUNCTIONS USED' is written in white, bold, uppercase letters at the top. The 'emaze' logo is in the bottom left corner.

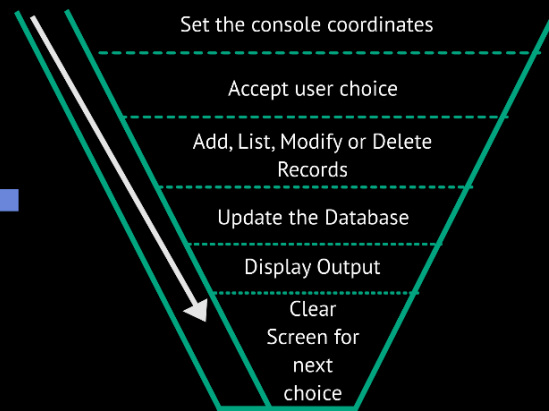
**FUNCTIONS USED**

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

**int main ()**

 emaze

## ALGORITHM



# THANK YOU

Paridhi Agarwal  
RA2111019010048