

```

#include<stdio.h>
#include<string.h>
#include<stdlib.h>
typedef struct
{
    char name[100];
    int age;
    float sal;
    int id;
}emp;
void inputdata(emp *ep)
{
    printf("enter employee's name:");
    fflush(stdin);
    gets(ep->name);
    printf("enter employee's age:");
    scanf("%d",&ep->age);
    printf("enter employee's salary:");
    scanf("%f",&ep->sal);
    printf("enter employee's id:");
    scanf("%d",&ep->id);
}
void savedata(FILE *fp,emp *ep)
{
    fprintf(fp,"%s\n%d\n%.2f\n%d\n",ep->name,ep->age,ep->sal,ep->id);
}
int readdata(FILE *fp,emp *ep)
{
    char temp[31];
    if(fgets(ep->name,99,fp)==NULL)
        return 0;

    ep->name[strlen(ep->name)-1]='\0';

    fgets(temp,30,fp);//replaces enter by null fgets//
    temp[strlen(temp)-1]='\0';
    ep->age=atoi(temp);
    fgets(temp,10,fp);
    temp[strlen(temp)-1]='\0';
    ep->sal=atof(temp);
    fgets(temp,10,fp);
    temp[strlen(temp)-1]='\0';
    ep->id=atoi(temp);
    return 1;
}
void printdata(emp *e,int i)
{
    printf("%d# employee's name is %s\nage is %d\nsalary is %.2f\nid is %d\n",i,e->name,e->age,e->sal,e->id);
}
void addrecord(FILE *fp)
{
    emp e;
    fp=fopen("c:\\users\\lenovo\\desktop\\emp.txt","a");

```

```

        inputdata(&e);
        savedata(fp,&e);
        fclose(fp);
    }
void displayrecord(FILE *fp)
{
    int i=1;
    char temp[100];
    emp e;
    fp=fopen("c:\\users\\lenovo\\desktop\\emp.txt","r");
    if(fp==NULL)
        puts("file not found");
    else
        puts("file found");
    while(readdata(fp,&e))
    {

        printdata(&e,i);
        i++;

    }
    fclose(fp);
}
void searchbyid(FILE *fp)
{
    emp e;
    fp=fopen("c:\\users\\lenovo\\desktop\\emp.txt","r");
    int id,found=0,i=1;
    printf("enter the id you want to search:");
    scanf("%d",&id);
    while(readdata(fp,&e))
    {
        if(e.id==id)
        {
            printdata(&e,i);
            found=1;
            break;
        }
        i++;
    }
    if(!found)
        printf("id not found");
    fclose(fp);
}
void searchbyname(FILE *fp)
{
    emp e;
    fp=fopen("c:\\users\\lenovo\\desktop\\emp.txt","r");
    int found=0,i=1;
    char name[50];
    printf("enter the name you want to search:");
    fflush(stdin);
    gets(name);
    while(!feof(fp))
    {
        readdata(fp,&e);
    }
}

```

```

        if(!strcmp(e.name,name))
        {
            printdata(&e,i);
            found=1;
            break;
        }
        i++;
    }
    if(!found)
        printf("name not found");
    fclose(fp);
}
void edit(FILE *fp)
{
    emp e;
    int id;int found=0;
    FILE *t;
    t=fopen("c:\\users\\lenovo\\desktop\\temp.txt","w");
    fp=fopen("c:\\users\\lenovo\\desktop\\emp.txt","r");
    printf("enter the id you want to edit:");
    scanf("%d",&id);
    while(readdata(fp,&e))
    {
        if(e.id==id)
        {
            found=1;
            inputdata(&e);
        }
        savedata(t,&e);
    }
    if(!found)
        printf("id not found");
    fclose(fp);
    fclose(t);
    unlink("c:\\users\\lenovo\\desktop\\emp.txt");
    rename("c:\\users\\lenovo\\desktop\\temp.txt","emp.txt");
}

```

```

void deleterecord(FILE *fp)
{
    emp e;
    int id;int found=0;
    FILE *t;
    t=fopen("c:\\users\\lenovo\\desktop\\temp.txt","w");
    fp=fopen("c:\\users\\lenovo\\desktop\\emp.txt","r");
    printf("enter the id you want to delete:");
    scanf("%d",&id);
    while(readdata(fp,&e))
    {
        if(e.id==id)
            found=1;
        else

```

```

        savedata(t,&e);
    }
    if(!found)
        printf("id not found");
    fclose(fp);
    fclose(t);
    unlink("c:\\users\\lenovo\\desktop\\emp.txt");
    rename("c:\\users\\lenovo\\desktop\\temp.txt","emp.txt");
}

int main()
{
    int n,i,ch;
    FILE *fp;
    while(1)
    {
        printf("1.Add\n2.Display\n3.Search by id\n4.Search by
name\n5.Edit\n6.Delete\n7.Exit\n");
        printf("enter choice:");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1: addrecord(fp);
                    break;
            case 2: displayrecord(fp);
                    break;
            case 3: searchbyid(fp);
                    break;
            case 4: searchbyname(fp);
                    break;
            case 5: edit(fp);
                    break;
            case 6: deleterecord(fp);
                    break;
            case 7: exit(1);
        }
    }
}

```