Assignment 1:  
---------------------  
Mr. XYZ has started a new company few months back. So far, he stored the  
employees records in a file named 'Employee.txt' and maintained it  
manually. But with the increase in number of employees he finds it very  
tedious to maintain. So, he asked IITP 2nd Yr CSE students to develop a C  
program to 'Select', 'Insert', 'Update' and 'Delete' an employee details  
in employee database file i.e. 'Employee.txt'.  
  
Format of employee database i.e. 'Employee.txt'  
  
<Num of employees>  
<EmpId> <Name>  <Dept Name>     <Skills>        <Salary>  
<EmpId> <Name>  <Dept Name>     <Skills>        <Salary>  
..  
..  
<EmpId> <Name>  <Dept Name>     <Skills>        <Salary>  
  
  
Use following command syntax for Selecting, Inserting, Updating and  
Deleting a record.  
Select:  
a. Select \* from Employee;  
b. Select \* from Employeewee where <Field>=<User input value>;  
c. Select \* from Employee where Salary > <User input value>;  
  
Insert:  
Insert into Employee values(<EmpId>,<Name>,<Dept Name>,<Skills>,<Salary>);  
  
Update:  
Update Employee set <Field>=<User input value>, ..., <Field>=<User input  
value> where <Field>=<User input value>;  
  
Delete:  
Delete from Employee where <Field>=<User input value>;  
  
  
Note:  
1. EmpId must be unique and un-editable.  
2. Use structure for temporarily storing the employees details. At  
initialization, program should read records from 'Employee.txt' (if  
present otherwise create from the code itself) and stores it in structure.  
Then after all the operations must be performed on structure only. On  
program termination modified records in structure should be reflected into  
database file in the same format.

#include <stdio.h>

typedef struct {

int empId;

char name[20];

char deptName[20];

char skills[20];

double salary;

} employee;

employee employees[100];

int i, num;

void initialize()

{

FILE \*ifp= fopen("Employee.txt", "r");

if (ifp == NULL) {

printf(“Couldn’t open file!”);

exit(1);

}

fscanf(ifp, "%d", &num);

for(i=0; i<num; i++)

{

(fscanf(ifp, "%d %s %s %s %lf ", &employees[i].empId, &employees[i].name, &employees[i].deptName, &employees[i].skills, &employees[i].salary);

}

fclose(ifp);

}

void options()

{

int option;

printf("\n Please enter a number from 1-5 to choose the following:");

printf("\n 1. Select");

printf("\n 2. Insert");

printf("\n 3. Update");

printf("\n 4. Delete");

printf("\n 5. Exit");

scanf("%d", &option);

switch(option)

{

case 1: Select(); break;

case 2: Insert(); break;

case 3: Update(); break;

case 4: Delete(); break;

case 5: break;

default: printf("\n wrong input! Try again."); options();

}

}

void insert()

{ int id;

printf("\n Please enter Employee id:");

scanf("%d", &id);

int flag= checkId(id);

if(flag==0)

{

employees[num].empId=id;

printf("\n ID Unique! Enter other details-");

printf("\n Name:");

scanf("%s", &employee)

}

else if(flag==1)

{

printf("\n ID not unique!Enter again ");

insert();

}

}

int checkId(int id)

{

int i; int flag=0;

for (i=0; i<num; i++)

{

if (employees[i].empId==id)

flag = 1;

}

return (flag);

}

int main()

{

initialize();

options();

return 0;

}

#include <stdio.h>

#include <string.h>

#include <conio.h>

typedef struct {

int empId;

char name[20];

char deptName[20];

char skills[20];

double salary;

} employee;

employee employees[100];

int i, num;

void initialize()

{

FILE \*ifp= fopen("Employee.txt", "r");

if (ifp == NULL) {

printf(“Couldn’t open file!”);

exit(1);

}

fscanf(ifp, "%d", &num);

for(i=0; i<num; i++)

{

(fscanf(ifp, "%d %s %s %s %lf ", &employees[i].empId, &employees[i].name, &employees[i].deptName, &employees[i].skills, &employees[i].salary);

}

fclose(ifp);

}

void options()

{

int option;

printf("\n Please enter a number from 1-5 to choose the following:");

printf("\n 1. Select");

printf("\n 2. Insert");

printf("\n 3. Update");

printf("\n 4. Delete");

printf("\n 5. Exit");

scanf("%d", &option);

switch(option)

{

case 1: Select(); break;

case 2: Insert(); break;

case 3: Update(); break;

case 4: Delete(); break;

case 5: break;

default: printf("\n wrong input! Try again."); options();

}

}

void insert()

{ int id;

printf("\n Please enter Employee id:");

scanf("%d", &id);

int flag= checkId(id);

if(flag==0)

{

employees[num].empId=id;

printf("\n ID Unique! Enter other details-");

printf("\n Name:");

scanf("%s", &employees[num].name);

printf("\n Dept Name:");

scanf("%s", &employees[num].deptName);

printf("\n Skills:");

scanf("%s", &employees[num].skills);

printf("\n Salary:");

scanf("%lf", &employees[num].salary);

num++;

updateFile();

options();

}

else if(flag==1)

{

printf("\n ID not unique!Enter again ");

insert();

}

}

int checkId(int id)

{

int i; int flag=0;

for (i=0; i<num; i++)

{

if (employees[i].empId==id)

flag = 1;

}

return (flag);

}

void select()

{

printf("\n Please enter a number from 1-5: ");

printf("1. Employee ID");

printf("2. Name");

printf("3. Department Name");

printf("4. Skills");

printf("5. Salary");

int option;

scanf("%d", &option);

if(option==1)

{

int id, flag=0;

printf("\n Enter ID:");

scanf("%d", &id);

for(i=0; i<num; i++)

{

if(employees[i].empId==id)

{

flag=1;

printf("\n EmpId: %d", employees[i].empId);

printf("\n Name: %s", employees[i].name);

printf("\n Department Name: %s", employees[i].deptName);

printf("\n Skills: %d", employees[i].skills);

printf("\n Salary: %lf", employees[i].salary);

}

}

if(flag==0)

printf("Entered Employee ID doesn't exist!");

}

options();

}

int main()

{

initialize();

options();

return 0;

}