/\*\*\*\*\* Elictric Core Service project \*\*\*\*\*/

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <time.h>

#define ProudctFilePath "E:\\technicProj\\Proudcts.bin"

#define UserFilePath "E:\\technicProj\\Users.bin"

#define TechnicianFilePath "E:\\technicProj\\Technician.bin"

#define problemsFilePath "E:\\technicProj\\Problems.bin"

FILE\* userFile;

FILE\* TechnicianFile;

FILE\* ProudctFile;

FILE\* ProblemsFile;

int userCounter = 0;

int productCount = 0;

int numOfTech = 0;

int problemsCounter = 0;

char officerName[] = "nasr";

char officerPassword[] = "12345";

struct PurchseDate {

int day;

int month;

int year;

};

struct product

{

char name[10];

int id;

int prodnum;

};

struct product prod;

struct UserDetails

{

char name[10];

int person\_id;

char address[20];

struct PurchseDate purchseDate;

int productId;

int userNumber = 0;

int Km; //landmark

bool visit;

};

struct UserDetails user;

struct Technician

{

char name[10];

char password[20];

int tech\_id;

int productId;

int placeKm;

int techNumber;

};

struct Technician tech;

struct problems

{

int prodid;

int type; //1:critical 2:regular 3:maintenance

double time;

bool care;

int number;

int userId;

};

struct problems problem;

int printUserDetails()

{

printf("\033[1;33m");

fopen\_s(&userFile, UserFilePath, "rb");

if (userFile == NULL) {

printf("Error! opening file to read\n");

return 0;

}

fread(&user, sizeof(struct UserDetails), 1, userFile);

while (!feof(userFile))

{

printf(" CustomerNumber: %d\n", user.userNumber);

printf(" buy - prod id: %d\n", user.productId);

printf("purchase Date: %d/%d/%d\n", user.purchseDate.day, user.purchseDate.month, user.purchseDate.year);

printf(" customer name:%s\n", user.name);

printf(" person id:%d\n", user.person\_id);

printf(" address:%s\n", user.address);

printf(" Customer Km: %d\n", user.Km);

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

fread(&user, sizeof(struct UserDetails), 1, userFile);

}

fclose(userFile);

}//end print user

int readUserDetails()

{

printf("\033[1;31m");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

userCounter = 0;

fopen\_s(&userFile, UserFilePath, "rb");

if (userFile == NULL) {

printf("Error! opening file to read\n");

printf("this is the first user !!!!\n");

return 0;

}

fread(&user, sizeof(struct UserDetails), 1, userFile);

while (!feof(userFile))

{

fread(&user, sizeof(struct UserDetails), 1, userFile);

if (user.userNumber > 0)

userCounter++;

}

printf("You have %d Customer in your company:\n", userCounter);

fclose(userFile);

}//end read userfile

int addUserDetails()

{

readUserDetails(); //count how many user i have

printf("\033[1;33m");

user.userNumber = userCounter + 1;

printf("enter details to CustomerNumber %d:\n", user.userNumber);

fopen\_s(&userFile, UserFilePath, "ab");

if (userFile == NULL) {

printf("Error! opening file to write");

// Program exits if the file pointer returns NULL.

return 0;

}

printf("enter your prod id:\n");

scanf\_s("%d", &user.productId);

char c2 = getchar();

printf("enter the day of Purchase : \n", user.purchseDate.day);

scanf\_s("%d", &user.purchseDate.day);

getchar();

printf("enter the month of Purchase : \n", user.purchseDate.month);

scanf\_s("%d", &user.purchseDate.month);

getchar();

printf("enter the year of Purchase : \n", user.purchseDate.year);

scanf\_s("%d", &user.purchseDate.year);

getchar();

// fgets(user.PurchaseDate, 11, stdin);

//user.PurchaseDate[strlen(user.PurchaseDate) - 1] = 0;

printf("enter your name: \n");

fgets(user.name, 10, stdin);

user.name[strlen(user.name) - 1] = 0;

printf("enter your person id: \n");

scanf\_s("%d", &user.person\_id);

char c4 = getchar();

printf("enter your address:\n");

fgets(user.address, 20, stdin);

user.address[strlen(user.address) - 1] = 0;

printf("enter your Location in Km\n");

scanf\_s("%d", &user.Km);

char c3 = getchar();

user.visit = 0;

fwrite(&user, sizeof(struct UserDetails), 1, userFile);

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

fclose(userFile);

}//end add user

int checkUser(int UserId, int userProdId) {

// printf("please enter the user id for searching:\n");

fopen\_s(&userFile, UserFilePath, "rb");

if (userFile == NULL) {

printf("Error! opening file to read\n");

printf("this is the first user !!!!\n");

return 0;

}

// scanf\_s("%d", &UserId);

// fread(&user, sizeof(struct UserDetails), 1, userFile);

while (!feof(userFile))

{

fread(&user, sizeof(struct UserDetails), 1, userFile);

if ((UserId == user.person\_id) && (userProdId == user.productId))

{

printf("user name:%s\n", user.name);

printf("---person id:%d\n", user.person\_id);

printf("---address:%s\n", user.address);

printf("---CustomerNumber: %d\n", user.userNumber);

printf("---prod id: %d\n", user.productId);

printf("purchase Date: %d/%d/%d\n", user.purchseDate.day, user.purchseDate.month, user.purchseDate.year);

printf("---Location Km: %d\n", user.Km);

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

fclose(userFile); return 1;

}

}

printf("user not found OR product id is wrong!!!\n");

fclose(userFile);

return 0;

}//end ceck user

int readTechDetails()

{

printf("\033[1;31m");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

numOfTech = 0;

fopen\_s(&TechnicianFile, TechnicianFilePath, "rb");

if (TechnicianFile == NULL) {

printf("Error! opening file to read\n");

printf("This is the first technician !!!!\n");

return 0;

}

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

while (!feof(TechnicianFile))

{

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

if (tech.techNumber > 0)

numOfTech++;

}

printf("You have %d technicians in your company:\n", numOfTech);

fclose(TechnicianFile);

}//end read tech file

int printTechDetails()

{

printf("\033[1;33m");

fopen\_s(&TechnicianFile, TechnicianFilePath, "rb");

if (TechnicianFile == NULL) {

printf("Error! opening file to read\n");

return 0;

}

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

while (!feof(TechnicianFile))

{

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

printf("Tecnician's Number: %d\n", tech.techNumber);

printf("Tecnician's ID %d\n", tech.tech\_id);

printf("Tecnician's name:%s\n", tech.name);

printf("Tecnician's password:%s\n", tech.password);

printf("Tecnician can fix product with ID %d \n", tech.productId);

printf("Tecnician's service area in Km: %d\n", tech.placeKm);

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

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

}

fclose(TechnicianFile);

}//end print tech

int addTechnicianDetails()

{

readTechDetails(); //count how many technician we have

printf("\033[1;33m");

tech.techNumber = numOfTech + 1;

printf("Enter details to technician number %d:\n", tech.techNumber);

//add technicians details

fopen\_s(&TechnicianFile, TechnicianFilePath, "ab");

if (TechnicianFile == NULL) {

printf("Error! opening file to write");

// Program exits if the file pointer returns NULL.

return 0;

}

printf("Enter Technician name:\n");

fgets(tech.name, 10, stdin);

tech.name[strlen(tech.name) - 1] = 0;

printf("Enter technical password:\n");

fgets(tech.password, 20, stdin);

tech.password[strlen(tech.password) - 1] = 0;

printf("Enter Technician ID:\n");

scanf\_s("%d", &tech.tech\_id);

char c2 = getchar();

printf("Enter which product\_id you work: \n");

scanf\_s("%d", &tech.productId);

char c4 = getchar();

printf("Enter your service area in Km:\n");

scanf\_s("%d", &tech.placeKm);

char c3 = getchar();

fwrite(&tech, sizeof(struct Technician), 1, TechnicianFile);

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

fclose(TechnicianFile);

return 0;

}//end add tech

int searchTechByID() {

int techID;

fopen\_s(&TechnicianFile, TechnicianFilePath, "rb");

if (TechnicianFile == NULL) {

printf("Error! opening file to read\n");

printf("This is the first user !!!!\n");

return 0;

}

printf("Please enter the technician's ID:\n");

scanf\_s("%d", &techID);

int a = getchar();

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

while (!feof(TechnicianFile))

{

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

if (techID == tech.tech\_id) {

printf("\036[1;32m");

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

printf("---Technician's Number is: %d\n", tech.techNumber);

printf("---Technician's name is: %s\n", tech.name);

printf("---Technician's Id is: %d\n", tech.tech\_id);

printf("---Technician works on prod ID: %d\n", tech.productId);

printf("---Technician works on area:%d\n", tech.placeKm);

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

return 0;

}

}

printf("Technician ID isn't found\n");

fclose(TechnicianFile);

return 0;

}

int checkTech(int techId, int techProdId) {

// printf("please enter the tech id for searching:\n");

fopen\_s(&TechnicianFile, TechnicianFilePath, "rb");

if (TechnicianFile == NULL) {

printf("Error! opening file to read\n");

printf("this is the first user !!!!\n");

return 0;

}

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

while (!feof(TechnicianFile))

{

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

if ((techId == tech.tech\_id) && (techProdId == tech.productId))

{

printf("technical name:%s\n", tech.name);

printf("---tech id:%d\n", tech.tech\_id);

printf("---tech password:%s\n", tech.password);

printf("---techNumber: %d\n", tech.techNumber);

printf("---prod id %d\n", tech.productId);

printf("---TechPlace Km: %d\n", tech.placeKm);

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

fclose(TechnicianFile); return 1;

}

}

printf("techinical not found OR product id is wrong!!!\n");

fclose(TechnicianFile);

return 0;

}

int SearchProductId(int ProductId) {

fopen\_s(&ProudctFile, ProudctFilePath, "rb");

if (ProudctFile == NULL) {

printf("Error! opening file to read\n");

printf("this is the first Product !!!!\n");

return 0;

}

// printf("please enter the Product id:\n");

//scanf\_s("%d", &ProductId);

fread(&prod, sizeof(struct product), 1, ProudctFile);

while (!feof(ProudctFile))

{

fread(&prod, sizeof(struct product), 1, ProudctFile);

if (ProductId == prod.id) {

printf("\033[1;33m");

printf("Product name:%s\n", prod.name);

printf("--- product id:%d\n", prod.id);

printf("--- product number %d\n", prod.prodnum);

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

fclose(ProudctFile);

return 1;

}

}

printf("Proudct not found\n");

fclose(ProudctFile);

}

int printProdDetails()

{

printf("\033[1;33m");

fopen\_s(&ProudctFile, ProudctFilePath, "rb");

if (ProudctFile == NULL) {

printf("Error! opening file to read\n");

return 0;

}

fread(&prod, sizeof(struct product), 1, ProudctFile);

while (!feof(ProudctFile))

{

printf("Proudct number: %d\n", prod.prodnum);

printf("Product name: %s\n", prod.name);

printf("Proudct ID: %d\n", prod.id);

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

fread(&prod, sizeof(struct product), 1, ProudctFile);

}

fclose(ProudctFile);

}

int readProdDetails()

{

printf("\033[1;31m");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

productCount = 0;

fopen\_s(&ProudctFile, ProudctFilePath, "rb");

if (ProudctFile == NULL) {

printf("Error! opening file to read\n");

printf("this is the first Product !!!!\n");

return 0;

}

// n = 0;

fread(&prod, sizeof(struct product), 1, ProudctFile);

while (!feof(ProudctFile))

{

fread(&prod, sizeof(struct product), 1, ProudctFile);

if (prod.prodnum > 0)

productCount++;

}

printf("You have %d Proudcts in your company:\n", productCount);

fclose(ProudctFile);

}

int addProdDetails()

{

readProdDetails(); //count how many Products i have

printf("\033[1;33m");

prod.prodnum = productCount + 1;

printf("enter details to Product Number %d:\n", prod.prodnum);

//\*add user details

fopen\_s(&ProudctFile, ProudctFilePath, "ab");

if (ProudctFile == NULL) {

printf("Error! opening file to write");

// Program exits if the file pointer returns NULL.

return 0;

}

printf("enter Product name:\n");

fgets(prod.name, 11, stdin);

prod.name[strlen(prod.name) - 1] = 0;

printf("enter Product Id:\n");

scanf\_s("%d", &prod.id);

char c2 = getchar();

prod.prodnum = productCount + 1;

fwrite(&prod, sizeof(struct product), 1, ProudctFile);

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

fclose(ProudctFile);

}

int printProblems()

{

printf("\033[1;33m");

fopen\_s(&userFile, problemsFilePath, "rb");

if (userFile == NULL) {

printf("Error! opening userfile to read\n");

return 0;

}

fopen\_s(&ProblemsFile, problemsFilePath, "rb");

if (ProblemsFile == NULL) {

printf("Error! opening Problemfile to read\n");

return 0;

}

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

while (!feof(ProblemsFile))

{

printf("problem number: %d\n", problem.number);

printf("problem user id: %d\n", problem.userId);

printf("problem Product id: %d\n", problem.prodid);

if (problem.type == 1)

printf("problem type is Maintenance and have num=: %d\n", problem.type);

if (problem.type == 2)

printf("problem type is Regular and have num=: %d\n", problem.type);

if (problem.type == 3)

printf("problem type is Critical and have num=: %d\n", problem.type);

printf("problem time :%.2lf\n ", problem.time);

printf("problem care :%i\n ", problem.care);

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

}

fclose(ProblemsFile);

}

int readProblems()

{

printf("\033[1;31m");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

problemsCounter = 0;

fopen\_s(&ProblemsFile, problemsFilePath, "rb");

if (ProblemsFile == NULL) {

printf("Error! opening file to read\n");

printf("this is the first problem !!!!\n");

return 0;

}

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

while (!feof(ProblemsFile))

{

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

problemsCounter++;

}

printf("You have %d problems in your company:\n", problemsCounter);

fclose(ProblemsFile);

}

int maintancecheck() {

int now;

time\_t t;

struct tm now1;

time(&t);

now = localtime\_s(&now1, &t);

struct Date {

int d, m, y;

};

struct Date d;

printf("\033[1;33m");

fopen\_s(&userFile, UserFilePath, "rb");

if (userFile == NULL) {

printf("Error! opening userfile to read\n");

return 0;

}

/\* fopen\_s(&ProblemsFile, problemsFilePath, "rb");

if (ProblemsFile == NULL) {

printf("Error! opening Problemfile to read\n");

return 0;

}\*/

int current\_day = now1.tm\_mday;

int current\_month = now1.tm\_mon + 1;

int current\_year = now1.tm\_year + 1900;

int purchse\_day = user.purchseDate.day;

int purchse\_month = user.purchseDate.month;

int purchse\_year = user.purchseDate.year;

//printf("current Date %d/%d/%d\n"), now1.tm\_mday, now1.tm\_mon + 1, now1.tm\_year + 1900);

//printf("purchase Date %d/%d/%d\n"), user.purchseDate.day, user.purchseDate.month, user.purchseDate.year);

int month[] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

if (user.purchseDate.day > current\_day)

{

current\_day = current\_day + month[purchse\_month - 1];

current\_month = current\_month - 1;

}

if (user.purchseDate.month > current\_month)

{

current\_year = current\_year - 1;

current\_month = current\_month + 12;

}

int calculated\_date = current\_day - purchse\_day;

int calculated\_month = current\_month - purchse\_month;

int calculated\_year = current\_year - purchse\_year;

printf("bought date \nYears: %d Months: %d Days:" " %d\n", calculated\_year, calculated\_month, calculated\_date);

printf("current date \nYears: %d Months: %d Days:" " %d\n", current\_day, current\_month, current\_year);

if (calculated\_year == 0 && calculated\_month < 6) {

if (user.visit = false) {

// fclose(ProblemsFile);

fclose(userFile);

return 1;

}

}

else {

// fclose(ProblemsFile);

fclose(userFile);

return 2;

}

}

int addProblems()

{

readProblems(); //count how many Products i have

printf("\033[1;33m");

problem.number = problemsCounter + 1;

printf("enter details to problem Number %d:\n", problem.number);

fopen\_s(&ProblemsFile, problemsFilePath, "ab+");

if (ProblemsFile == NULL) {

printf("Error! opening file to write");

// Program exits if the file pointer returns NULL.

return 0;

}

printf("enter user Id:\n");

scanf\_s("%d", &problem.userId);

char c2 = getchar();

printf("enter product id: \n");

scanf\_s("%d", &problem.prodid);

char c4 = getchar();

if (!checkUser(problem.userId, problem.prodid))

{

fclose(ProblemsFile);

return 0;

}

int x = maintancecheck();

switch (x) {

case 1:

printf("user must have maintance visit\n");

problem.type = 1;

break;

case 2:

printf(" the product is out of the maintance date ,or the user already had a maintance visit\n");

printf(" please choose Problem type\n2.regular\n3.critical\n");

scanf\_s("%d", &problem.type);

char c1 = getchar();

printf("Please enter problem time\n");

scanf\_s("%lf", &problem.time);

char c3 = getchar();

break;

}

problem.care = 0;

fwrite(&problem, sizeof(struct problems), 1, ProblemsFile);

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

fclose(ProblemsFile);

return 0;

}

int updateProblems()

{

fopen\_s(&ProblemsFile, problemsFilePath, "ab");

if (ProblemsFile == NULL) {

printf("Error! opening file to write");

// Program exits if the file pointer returns NULL.

return 0;

}

printf("enter user Id:\n");

scanf\_s("%d", &problem.userId);

char c2 = getchar();

printf("enter product id: \n");

scanf\_s("%d", &problem.prodid);

char c4 = getchar();

if (!checkUser(problem.userId, problem.prodid))

{

fclose(ProblemsFile);

return 0;

}

printf("enter problem type //1:critical 2:regular 3:maintenance:\n");

scanf\_s("%d", &problem.type);

char c1 = getchar();

printf("Please enter problem time\n");

scanf\_s("%lf", &problem.time);

char c3 = getchar();

fwrite(&problem, sizeof(struct problems), 1, ProblemsFile);

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

fclose(ProblemsFile);

return 0;

}

int modifyProblemStatus() {

int option = 0;

int type;

int problemNumber;

char ch;

// checkTech(int techId, int techProdId);

printf("\033[1;33m");

printf("Please enter Probelm number you want to modify: \n");

scanf\_s("%d", &problemNumber);

printf("\033[1;33m");

fopen\_s(&ProblemsFile, problemsFilePath, "rb+");

if (ProblemsFile == NULL) {

printf("Error! opening file to read\n");

return 0;

}

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

while (!feof(ProblemsFile))

{

if (problemNumber == problem.number)

{

option = problem.type;

}

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

}

/\* printf("Please enter technical id:\n");

int tech\_id;

scanf\_s("%d", &tech\_id);

if (!checkTech(tech\_id, problem.prodid))

{

fclose(ProblemsFile);

return 0;

}\*/

printf("\033[1;33m");

switch (option)

{

case 1:

{ printf("problem id %d is maintenance\n", problem.number);

printf("Did the visit done? y,n: ");

getchar();

scanf\_s("%c", &ch);

getchar();

if (ch == 'n')

{

printf("Visit did not done yet\n");

problem.care = false;

}

else {

printf("is there any other problems? y,n: \n");

getchar();

scanf\_s("%c", &ch);

getchar();

if (ch == 'y') {

printf("which type: 2,3\n");

scanf\_s("%d", &type);// מחזירה אותו למסך שיבוץ

problem.type = type;

printf("type updated.....\n");

}

else

{

problem.care = true;

printf("status updated.....\n");

}

}

break;

}

case 2:

{

printf("problem id %d is Regular\n", problem.number);

printf("have the problem solved ?y,n:\n ");

getchar();

scanf\_s("%c", &ch);

getchar();

if (ch == 'y')

{

printf("status updated.....\n");

problem.care = true;

}

else

{

printf("status not updated.....\n");

problem.care = false;

}

break;

}

case 3:

{ printf("problem id %d is criceital\n", problem.number);

printf("have the criceital problem solved ?y,n:\n");

getchar();

scanf\_s("%c", &ch);

getchar();

if (ch == 'y')

{

problem.care = true;

printf("status updated.....\n");

}

else

{

problem.type = 2;

printf("type updated.....\n");

}

break;

}

}

fwrite(&problem, sizeof(struct problems), 1, ProblemsFile);

fclose(ProblemsFile);

}//end

int jobPlacement()

{

char TechName[10];

printf("Welcome to your renovation paper....\n");

printf("please enter your technical name:\n");

fgets(TechName, 10, stdin);

TechName[strlen(TechName) - 1] = 0;

fopen\_s(&TechnicianFile, TechnicianFilePath, "rb");

if (TechnicianFile == NULL) {

printf("Error! opening Technician file to read\n");

return 0;

}

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

while (!feof(TechnicianFile))

{

if (strcmp(TechName, tech.name) == 0)

{

printf("Technician number:%d\n", tech.techNumber);

printf("Technician name:%s\n", tech.name);

break;

}

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

}

fclose(TechnicianFile);

fopen\_s(&ProblemsFile, problemsFilePath, "rb");

if (ProblemsFile == NULL) {

printf("Error! opening Problems file to read\n");

return 0;

}

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

while (!feof(ProblemsFile))

{

if (problem.prodid == tech.productId)

{

printf("problem number: %d\n", problem.number);

printf("problem user id: %d\n", problem.userId);

printf("problem Product id: %d\n", problem.prodid);

if (problem.type == 1)

printf("problem type is Critical and have num=: %d\n", problem.type);

if (problem.type == 2)

printf("problem type is Regular and have num=: %d\n", problem.type);

if (problem.type == 3)

printf("problem type is Maintenance and have num=: %d\n", problem.type);

printf("problem time :%.2lf\n ", problem.time);

printf("problem care :%i\n ", problem.care);

}

fread(&problem, sizeof(struct problems), 1, ProblemsFile);

}

fclose(TechnicianFile);

fclose(ProblemsFile);

}

int technicalLogin() {

int n;

char username[10];

char password[20];

printf("\033[1;35m");

printf("Please Enter your login credentials below\n");

printf("Enter Technician name: \n");

fgets(username, 10, stdin);

username[strlen(username) - 1] = 0;

printf("Enter Technician Password: \n");

fgets(password, 20, stdin);

password[strlen(password) - 1] = 0;

fopen\_s(&TechnicianFile, TechnicianFilePath, "rb");

if (TechnicianFile == NULL) {

printf("Error! opening file to read\n");

printf("This is the first user !!!!\n");

return 0;

}

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

while (!feof(TechnicianFile))

{

if (strcmp(username, tech.name) == 0)

{

if (strcmp(password, tech.password) == 0)

{

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

printf(".....WELCOME.....\n");

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

fclose(TechnicianFile);

printf("What do you want to do???\n");

printf("1:show job Placement \n");

printf("2:update status.\n");

scanf\_s("%d", &n);

char c = getchar();

if (n == 1)

jobPlacement();

else if (n == 2)

modifyProblemStatus();

else

printf("Sorry!!! it's a wrong option\n");

return 1;

}

else

{

printf("Sorry...technical paswword is wrong..\n"); return 0;

}

}

fread(&tech, sizeof(struct Technician), 1, TechnicianFile);

}

printf("Technician isn't found\n");

fclose(TechnicianFile);

return 0;

}

int officeWork()

{

for (;;)

{

printf("\033[1;33m");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("Please choose what you want to do\n");

printf("1. Add User Details \n");

printf("2. Add Product Details\n");

printf("3. Add Technician\n");

printf("4. Add problems\n");

printf("5. print User Details\n");

printf("6. print Technician Details\n");

printf("7. print Product Details\n");

printf("8. print Problems\n");

printf("9. Exit\n"); // exist

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

int option;

// option= getchar();

scanf\_s("%d", &option);

char c = getchar();

switch (option)

{

case 1:

addUserDetails();

break;

case 2:

addProdDetails();

break;

case 3:

addTechnicianDetails();

break;

case 4:

addProblems();

break;

case 5:

readUserDetails();

printUserDetails();

break;

case 6:

readTechDetails();

printTechDetails();

break;

case 7:

readProdDetails();

printProdDetails();

break;

case 8:

printProblems();

break;

case 9:

return 0;

default:

break;

}

}

return 0;

}

void loginOfficeWork()

{

char username[10];

char password[20];

printf("\033[1;34m");

printf("Wlecome to the office work..\n");

printf("Please enter officer name:\n");

fgets(username, 10, stdin);

username[strlen(username) - 1] = 0;

printf("Please enter officer password:\n");

fgets(password, 20, stdin);

password[strlen(password) - 1] = 0;

if (strcmp(officerName, username) == 0)

{

if (strcmp(officerPassword, password) == 0)

officeWork();

else

printf("Sorry...Password is wrong!!!\n");

}

else

printf("Sorry...Officer name is wrong!!!\n");

}

int main()

{

for (;;)

{

printf("\033[1;32m");

printf("\*\*\*\*\*\*\*\*\*\*\*\n");

printf("Please chose option: \n");

printf("\n1.Technician\n");

printf("\n2.office menu\n");

printf("\n9.exit\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\n");

int option;

scanf\_s("%d", &option);

char c;

c = getchar();

switch (option)

{

case 1:technicalLogin();

break;

case 2:

loginOfficeWork();

break;

case 9:return 0;

}

}

return 0;

}