#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define FILENAME "contacts.txt"

#define MAX\_NAME\_LEN 100

#define MAX\_PHONE\_LEN 20

#define MAX\_EMAIL\_LEN 100

typedef struct {

char name[MAX\_NAME\_LEN];

char phone[MAX\_PHONE\_LEN];

char email[MAX\_EMAIL\_LEN];

} Contact;

void load\_contacts(Contact \*\*contacts, int \*count) {

FILE \*file = fopen(FILENAME, "r");

if (!file) return;

\*count = 0;

\*contacts = NULL;

char line[256];

while (fgets(line, sizeof(line), file)) {

\*contacts = realloc(\*contacts, (\*count + 1) \* sizeof(Contact));

sscanf(line, "%[^,],%[^,],%s", (\*contacts)[\*count].name, (\*contacts)[\*count].phone, (\*contacts)[\*count].email);

(\*count)++;

}

fclose(file);

}

void save\_contacts(Contact \*contacts, int count) {

FILE \*file = fopen(FILENAME, "w");

if (!file) return;

for (int i = 0; i < count; i++) {

fprintf(file, "%s,%s,%s\n", contacts[i].name, contacts[i].phone, contacts[i].email);

}

fclose(file);

}

void add\_contact(Contact \*\*contacts, int \*count) {

\*contacts = realloc(\*contacts, (\*count + 1) \* sizeof(Contact));

printf("Enter name: ");

scanf(" %[^\n]", (\*contacts)[\*count].name);

printf("Enter phone number: ");

scanf(" %[^\n]", (\*contacts)[\*count].phone);

printf("Enter email address: ");

scanf(" %[^\n]", (\*contacts)[\*count].email);

(\*count)++;

save\_contacts(\*contacts, \*count);

printf("Contact added successfully!\n");

}

void view\_contacts(Contact \*contacts, int count) {

if (count == 0) {

printf("No contacts found.\n");

return;

}

for (int i = 0; i < count; i++) {

printf("%d. Name: %s, Phone: %s, Email: %s\n", i + 1, contacts[i].name, contacts[i].phone, contacts[i].email);

}

}

void edit\_contact(Contact \*contacts, int count) {

if (count == 0) {

printf("No contacts to edit.\n");

return;

}

view\_contacts(contacts, count);

int index;

printf("Enter the contact number to edit: ");

scanf("%d", &index);

if (index < 1 || index > count) {

printf("Invalid contact number.\n");

return;

}

index--;

printf("Enter new name: ");

scanf(" %[^\n]", contacts[index].name);

printf("Enter new phone number: ");

scanf(" %[^\n]", contacts[index].phone);

printf("Enter new email address: ");

scanf(" %[^\n]", contacts[index].email);

save\_contacts(contacts, count);

printf("Contact updated successfully!\n");

}

void delete\_contact(Contact \*\*contacts, int \*count) {

if (\*count == 0) {

printf("No contacts to delete.\n");

return;

}

view\_contacts(\*contacts, \*count);

int index;

printf("Enter the contact number to delete: ");

scanf("%d", &index);

if (index < 1 || index > \*count) {

printf("Invalid contact number.\n");

return;

}

index--; // Convert to zero-based index

for (int i = index; i < \*count - 1; i++) {

(\*contacts)[i] = (\*contacts)[i + 1];

}

(\*count)--;

\*contacts = realloc(\*contacts, (\*count) \* sizeof(Contact));

save\_contacts(\*contacts, \*count);

printf("Contact deleted successfully!\n");

}

int main() {

Contact \*contacts = NULL;

int count = 0;

load\_contacts(&contacts, &count);

while (1) {

printf("\nContact Manager\n");

printf("1. Add Contact\n");

printf("2. View Contacts\n");

printf("3. Edit Contact\n");

printf("4. Delete Contact\n");

printf("5. Exit\n");

printf("Select an option: ");

int choice;

scanf("%d", &choice);

switch (choice) {

case 1:

add\_contact(&contacts, &count);

break;

case 2:

view\_contacts(contacts, count);

break;

case 3:

edit\_contact(contacts, count);

break;

case 4:

delete\_contact(&contacts, &count);

break;

case 5:

free(contacts);

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

return 0;

default:

printf("Invalid choice. Please try again.\n");

}

}

}