#include <stdio.h>

#include <stdlib.h>

#include <windows.h>

#include <conio.h>

void menu(void);

void add\_student\_data(void);

void remove\_student\_data(void);

void update\_student\_data(void);

void view\_record(void);

void functions(void);

void clear\_record(void);

struct student

{

char name[20];

int id;

char branch[20];

};

struct student s;

int choice;

void menu()

{

system("cls");

Sleep(100);

system("color 0b");

printf("\n\t---Student Management System---\n\n");

printf("1. View record\n");

printf("2. Add student data\n");

printf("3. Remove student data\n");

printf("4. Update student data\n");

printf("5. Clear record\n");

printf("6. Exit\n\n");

printf("Enter your choice : ");

scanf("%d", &choice);

functions();

}

void functions(void)

{

switch (choice)

{

case 1:

system("cls");

view\_record();

break;

case 2:

system("cls");

add\_student\_data();

break;

case 3:

system("cls");

remove\_student\_data();

break;

case 4:

system("cls");

update\_student\_data();

break;

case 5:

system("cls");

clear\_record();

break;

case 6:

system("cls");

system("color 0f");

exit(0);

default:

printf("\n\tInvalid choice...");

}

printf("\n\nPress any key to continue...");

getch();

menu();

}

void add\_student\_data(void)

{

FILE \*ptr;

ptr = fopen("Student\_record.bin", "ab");

if (ptr == NULL)

{

printf("\n\tERROR 404...\n");

return;

}

printf("\n\t---Add Student Record---\n\n");

printf("Name : ");

fflush(stdin);

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

printf("Id : ");

fflush(stdin);

scanf("%d", &s.id);

printf("Branch : ");

fflush(stdin);

scanf("%s", s.branch);

fwrite(&s, sizeof(s), 1, ptr);

fclose(ptr);

}

void view\_record(void)

{

FILE \*ptr;

ptr = fopen("Student\_record.bin", "rb");

int found = 0;

if (ptr == NULL)

{

printf("\n\tERROR 404...\n");

return;

}

printf("\n\t---View Record---\n\n");

while (fread(&s, sizeof(s), 1, ptr) == 1)

{

printf("Name : %s\nId : %d\nBranch : %s\n-----------------------\n", s.name, s.id, s.branch);

found = 1;

}

if (found == 0)

{

printf("no record found...");

}

fclose(ptr);

}

void remove\_student\_data(void)

{

FILE \*ptr1;

FILE \*ptr2;

ptr1 = fopen("Student\_record.bin", "rb");

ptr2 = fopen("temp.bin", "wb");

int id;

int found = 0;

if (ptr1 == NULL || ptr2 == NULL)

{

printf("\n\tERROR 404...\n");

return;

}

printf("\n\t---Remove Student Data---\n\n");

printf("Enter Student ID : ");

scanf("%d", &id);

while (fread(&s, sizeof(s), 1, ptr1) == 1)

{

if (s.id == id)

{

found = 1;

}

else

{

fwrite(&s, sizeof(s), 1, ptr2);

}

}

fclose(ptr1);

fclose(ptr2);

if (found == 0)

{

printf("\nSorry record not found :( ");

return;

}

ptr1 = fopen("Student\_record.bin", "wb");

ptr2 = fopen("temp.bin", "rb");

while (fread(&s, sizeof(s), 1, ptr2) == 1)

{

fwrite(&s, sizeof(s), 1, ptr1);

}

fclose(ptr1);

fclose(ptr2);

ptr2 = fopen("temp.bin", "wb");

fclose(ptr2);

printf("\nRecord is successfully deleted :) \n");

}

void update\_student\_data(void)

{

FILE \*ptr;

ptr = fopen("Student\_record.bin", "rb+");

int found = 0;

int id;

if (ptr == NULL)

{

printf("\n\tERROR 404...\n");

return;

}

printf("\n\t---Update Student Record---\n\n");

printf("Enter Student ID : ");

scanf("%d", &id);

while (fread(&s, sizeof(s), 1, ptr) == 1)

{

if (s.id == id)

{

found = 1;

break;

}

}

if (found == 0)

{

printf("\nSorry record not found :( ");

return;

}

fseek(ptr, -(sizeof(s)), SEEK\_CUR);

printf("\n\tEnter new details\n\n");

printf("Name : ");

fflush(stdin);

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

printf("Id : ");

fflush(stdin);

scanf("%d", &s.id);

printf("Branch : ");

fflush(stdin);

scanf("%s", s.branch);

fwrite(&s, sizeof(s), 1, ptr);

fclose(ptr);

printf("\n Student details updated successfully..\n");

}

void clear\_record(void)

{

FILE \*ptr;

ptr = fopen("Student\_record.bin", "w");

if (ptr == NULL)

{

printf("\n\tERROR 404...\n");

return;

}

printf("\n\t---Clear Record---\n\n");

fclose(ptr);

printf("Record deleted successfully...\n");

}

int main(void)

{

switch (choice)

{

case 1:

view\_record();

break;

case 2:

add\_student\_data();

break;

case 3:

remove\_student\_data();

break;

case 4:

update\_student\_data();

break;

case 5:

clear\_record();

break;

case 6:

system("color 0f");

exit(0);

default:

printf("\n\tInvalid choice...");

}

menu();

return 0;

}