FITNESS CLUB MANAGEMENT SYSTEM

Computer Programming

CSE101

SUBMITTED TO: Dr.Jyoti

SUBMITTED BY: ASWATHI (12211795), JAIDEEP (12213130), UMESH (12212197), BHUDEV (12213521)

ROLLNO: 25,26,27,28

FITNESS CLUB MANAGEMENT SYSTEM

OVERVIEW:

The Fitness Club Management System is a program developed using the C programming language that allows club owners to manage the customer's records in their fitness clubs. The program includes several modules, each responsible for a specific task, such as adding new customers, deleting existing customers, displaying the records of all customers, searching for specific customers by their mobile number or name, and displaying available slots for customers to book

INTRODUCTION:

The Fitness Club Management System is a project that aims to help club owners manage their customers' records with ease. The system allows the club owner to keep track of important information such as the customer's date of joining, package details, and training period. Additionally, the system enables the owner to store other details like the customer's name, mobile number, BMI, address, and package details.

MODULES:

- **Add Record of Customer to File:** This module allows the club owner to add the customer's record to the system. When this module is executed, it prompts the owner must input the customer's details such as name, mobile number, BMI, address,

package details, and date of joining. The program generates a unique ID for each customer, which is used to identify them in the record file. Once the owner inputs the customer's details, the program writes the information to the record file.

- **Delete the Record of Customer:** This module allows the club owner to remove customer details from the system. When this module is executed, the owner is prompted to enter the customer’s number. The program then reads the record file, searches for the customer with the corresponding number, and deletes their record from the file.

- **Display Record of All Customers:** This module enables the club owner to view all the customer records in the system. When executed, the program reads the record file and displays the name, mobile number, BMI,

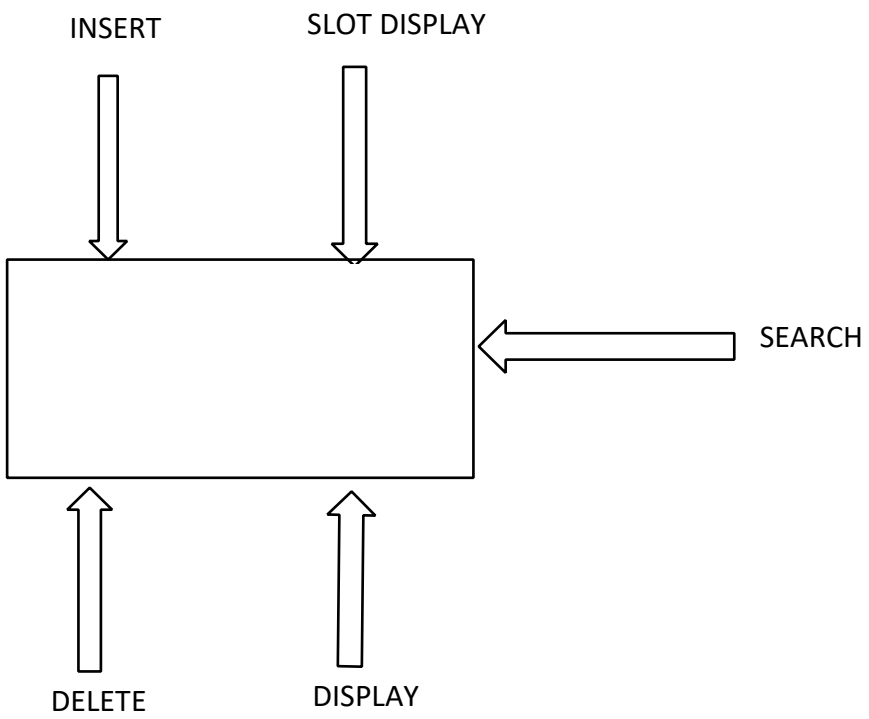
address, date of joining, and package details of each customer.

- **Search the Record of Customer by Mobile Number, Name or id:** This module allows the club owner to search for a customer's record by their mobile number, name or ID. When executed, the program prompts the user to enter the mobile number, name, or ID of the customer they wish to search for. The program then reads the record file, searches for the corresponding customer, and displays their details if they exist in the file.

- **Display slot timings of the club and user must be able to select any slot. (Check if slot if full.):** The "Display slot timings of the club and user must be able to select any slot" module displays the available slots for customers to book. When executed, the program reads the record file and checks for

any slots that are already booked. It then displays the available slots to the user, who can select any slot that is not full. If the user selects a slot that is already booked, the program displays an appropriate message.

ZERO-LEVEL DFD:



PROJECT CODE:

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>

// Structure to store customer details
struct customer {
    char name[50];
    char mobile[11];
    float bmi;
    char address[100];
    char package[20];
    char join_date[11];
    int train_period;
};

// Function to add customer record to file
void add_record() {
    struct customer c;
    FILE *f;
    f = fopen("customers.dat", "ab+");
    if (f == NULL) {
        printf("Error in opening file!");
        exit(1);
    }
```

```
    printf("\nEnter name: ");
    scanf("%s", c.name);
    printf("\nEnter mobile number: ");
    scanf("%s", c.mobile);
    printf("\nEnter BMI: ");
    scanf("%f", &c.bmi);
    printf("\nEnter address: ");
    scanf("%s", c.address);
    printf("\nEnter package details: ");
    scanf("%s", c.package);
    printf("\nEnter joining date (DD/MM/YYYY): ");
    scanf("%s", c.join_date);
    printf("\nEnter training period (in months): ");
    scanf("%d", &c.train_period);
    fwrite(&c, sizeof(c), 1, f);
    printf("\nRecord added successfully!\n");
    fclose(f);
}

// Function to delete customer record from file
void delete_record() {
    char mobile[11];
    int found = 0;
    struct customer c;
    FILE *f, *t;
    f = fopen("customers.dat", "rb");
    if (f == NULL){
```

```
        printf("Error in opening file!");
        exit(1);
    }
    t = fopen("temp.dat", "wb");
    if (t == NULL) {
        printf("Error in opening file!");
        fclose(f);
        exit(1);
    }
    printf("Enter mobile number of customer to be deleted: ");
    scanf("%s", mobile);
    while (fread(&c, sizeof(c), 1, f) == 1) {
        if (strcmp(c.mobile, mobile) != 0)
            fwrite(&c, sizeof(c), 1, t);
        else
            found = 1;
    }
    if (found == 1)
        printf("Record deleted successfully!\n");
    else
        printf("Record not found!\n");
    fclose(f);
    fclose(t);
    remove("customers.dat");
    rename("temp.dat", "customers.dat");
}
```

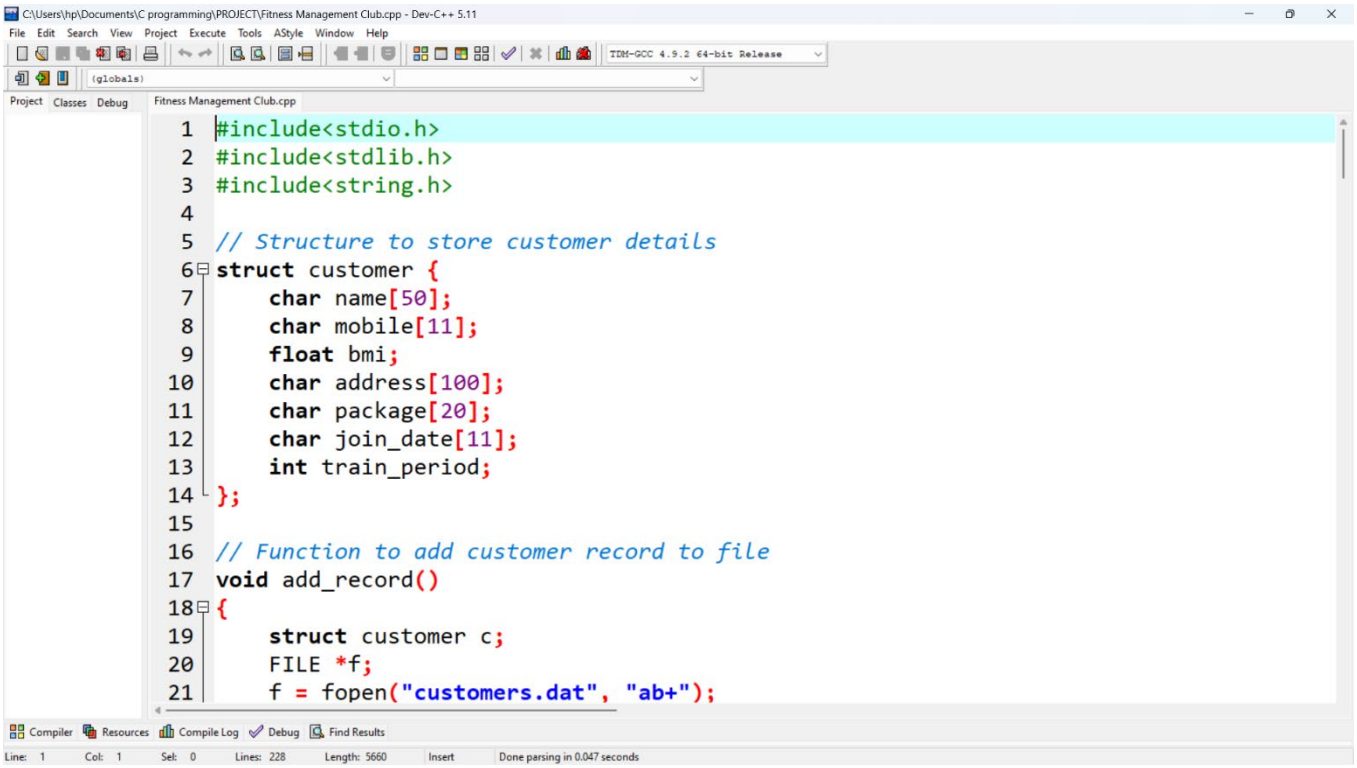
```
// Function to display all customer records from file
void display_records() {
    struct customer c;
    FILE *f;
    f = fopen("customers.dat", "rb");
    if (f == NULL) {
        printf("Error in opening file!");
        exit(1);
    }
    while (fread(&c, sizeof(c), 1, f) == 1) {
        printf("Name: %s\nMobile Number: %s\nBMI:
%f\nAddress: %s\nPackage Details: %s\nJoining Date:
%s\nTraining Period: %d months\n\n", c.name, c.mobile, c.bmi,
c.address, c.package, c.join_date, c.train_period);
    }
    fclose(f);
}

// Function to search for customer record by mobile number,
name, or id
void search_record() {
    char search_query[50];
    int found = 0;
    struct customer c;
    FILE *f;
    f = fopen("customers.dat", "rb");
    if (f == NULL) {
```

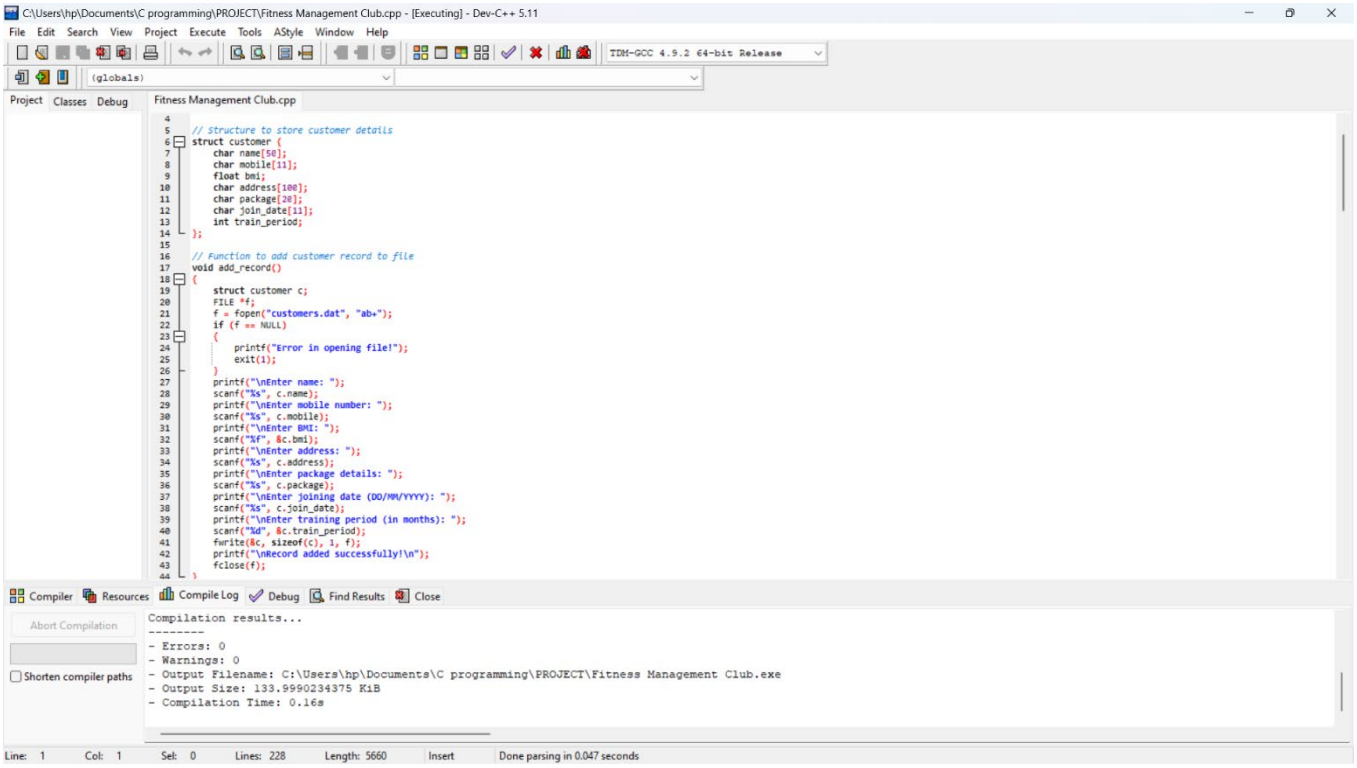
```
        printf("Error in opening file!");
        exit(1);

}
printf("Enter mobile number, name or ID to search: ");
scanf("%s", search_query);
while (fread(&c, sizeof(c), 1, f) == 1) {
   if (strcmp(c.mobile, search_query) == 0 || strcmp(c.name,
search_query) == 0) {
        printf("Name: %s\tMobile Number: %s\tBMI: %f\tAddress:
%s\tPackage Details: %s\tJoining Date: %s\tTraining Period: %d
months\n\n", c.name, c.mobile, c.bmi, c.address, c.package,
c.join_date, c.train_period);
        found = 1;
   }
}
if (found == 0)
   printf("Record not found!\n");
fclose(f);
}

// Function to display available slots and let user select one
void select_slot() {
int slots[5] = {0, 0, 0, 0, 0};
int slot, found = 0;
struct customer c;
FILE *f;
```

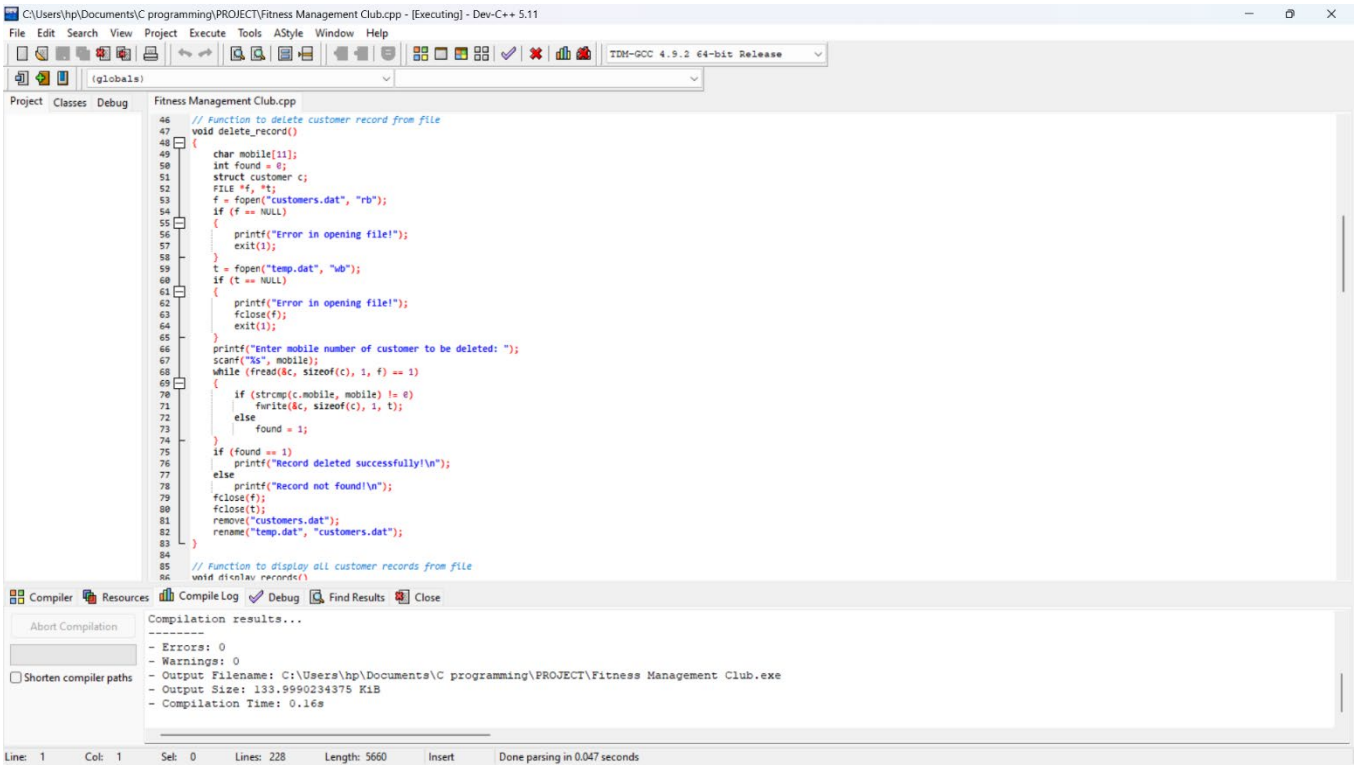
```
f = fopen("customers.dat", "rb");
if (f == NULL) {
printf("Error in opening file!");
exit(1);
}
while (fread(&c, sizeof(c), 1, f) == 1) {
if (c.train_period > 6 && c.train_period <= 12) {
if (strcmp(c.join_date, "01/01/2023") == 0) {
slots[0] = 1;
}
else if (strcmp(c.join_date, "01/02/2023") == 0) {
slots[1] = 1;
}
else if (strcmp(c.join_date, "01/03/2023") == 0) {
slots[2] = 1;
}
else if (strcmp(c.join_date, "01/04/2023") == 0) {
slots[3] = 1;
}
else if (strcmp(c.join_date, "01/05/2023") == 0) {
slots[4] = 1;
}
}
}
printf("Available slots:\n");
for (int i = 0; i < 5; i++) {
if (slots[i] == 0) {
```

```
printf("%d. 01/%02d/2023\n", i+1, i+1);
found = 1;
}
}
if (found == 0) {
printf("No slots available!\n");
return;
}
printf("Enter slot number: ");
scanf("%d", &slot);
if (slots[slot-1] == 1) {
printf("Slot already booked!\n");
return;
}
printf("Slot booked successfully!\n");
fclose(f);
}

int main() {
int choice;
while (1) {
printf("Fitness Club Management System\n");
printf("1. Add customer record to file\n2. Delete customer record from file\n3. Display all customer records\n4. Search for customer record\n5. Select slot\n6. Exit\n");
printf("Enter your choice: ");
scanf("%d", &choice);
```

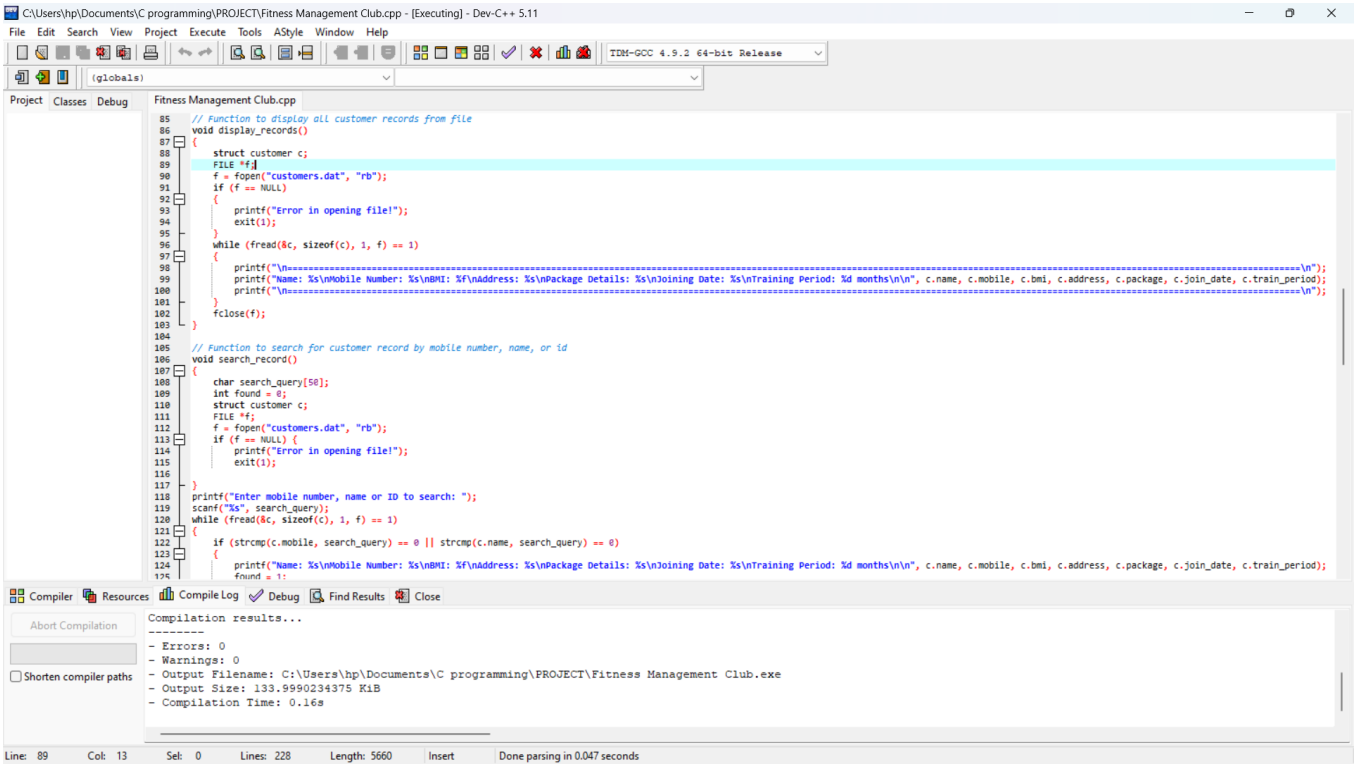
```
switch (choice) {
            case 1:
                add_record();
                break;
            case 2:
                delete_record();
                break;
            case 3:
                display_records();
                break;
            case 4:
                search_record();
                break;
            case 5:
                select_slot();
      case 6:
         exit(0);
      default:
         printf("Invalid choice!\n");
   }
}
return 0;
}
```


C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
// Structure to store customer details
struct customer {
char name[50];
char mobile[11];
float bmi;
char address[100];
char package[20];
char join_date[11];
int train_period;
};
// Function to add customer record to file
void add_record()
{
struct customer c;
FILE *f;
f = fopen("customers.dat", "ab+");
Compiler Resources Compile Log Debug Find Results
Line: 1 Col: 1 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds

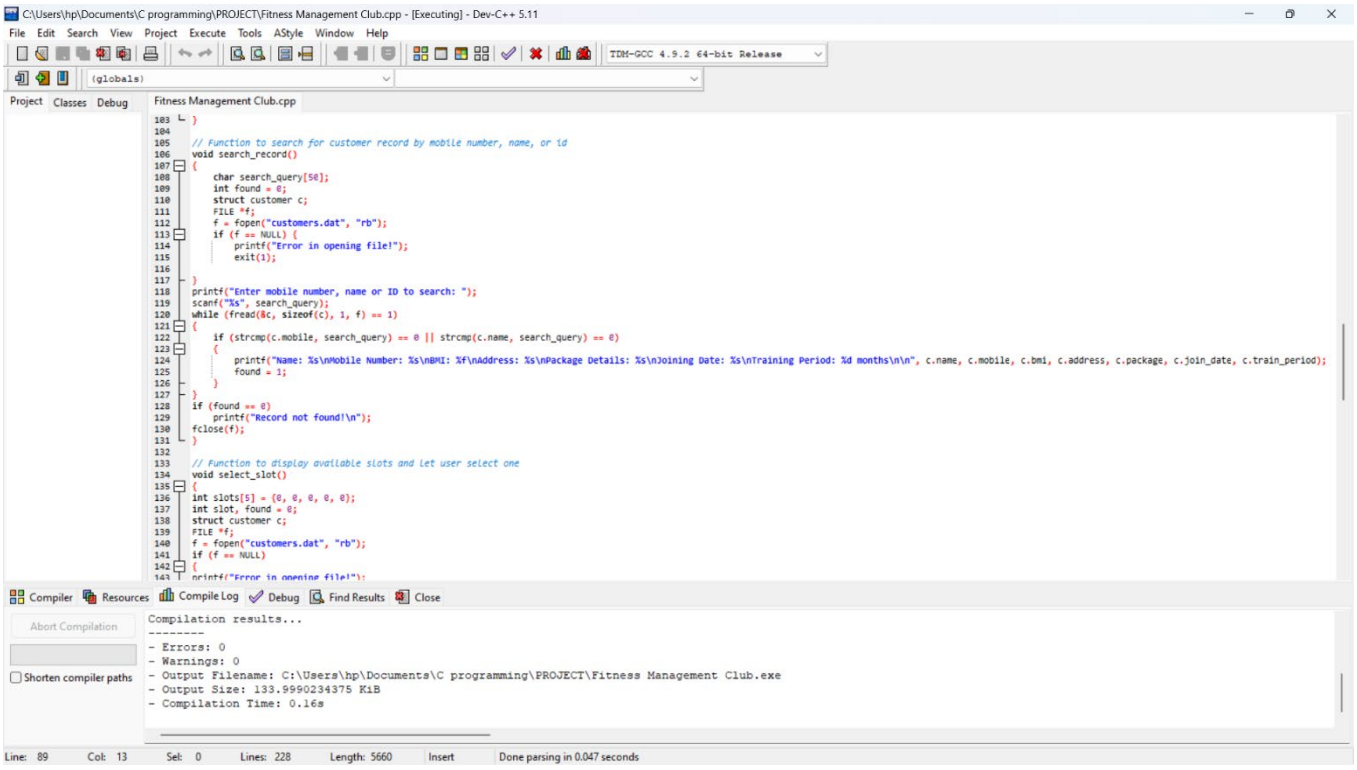

C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
// Structure to store customer details
struct customer {
char name[50];
char mobile[11];
float bmi;
char address[100];
char package[20];
char join_date[11];
int train_period;
};
// Function to add customer record to file
void add_record()
{
struct customer c;
FILE *f;
f = fopen("customers.dat", "ab+");
if (f == NULL)
{
printf("Error in opening file!");
exit(1);
}
printf("\nEnter name: ");
scanf("%s", c.name);
printf("\nEnter mobile number: ");
scanf("%s", c.mobile);
printf("\nEnter BMI: ");
scanf("%f", &c.bmi);
printf("\nEnter address: ");
scanf("%s", c.address);
printf("\nEnter package details: ");
scanf("%s", c.package);
printf("\nEnter joining date (DD/MM/YYYY): ");
scanf("%s", c.join_date);
printf("\nEnter training period (in months): ");
scanf("%d", &c.train_period);
fwrite(&c, sizeof(c), 1, f);
printf("\nRecord added successfully!\n");
fclose(f);
}
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.exe
- Output Size: 133.9990234375 KiB
- Compilation Time: 0.16s
Line: 1 Col: 1 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds

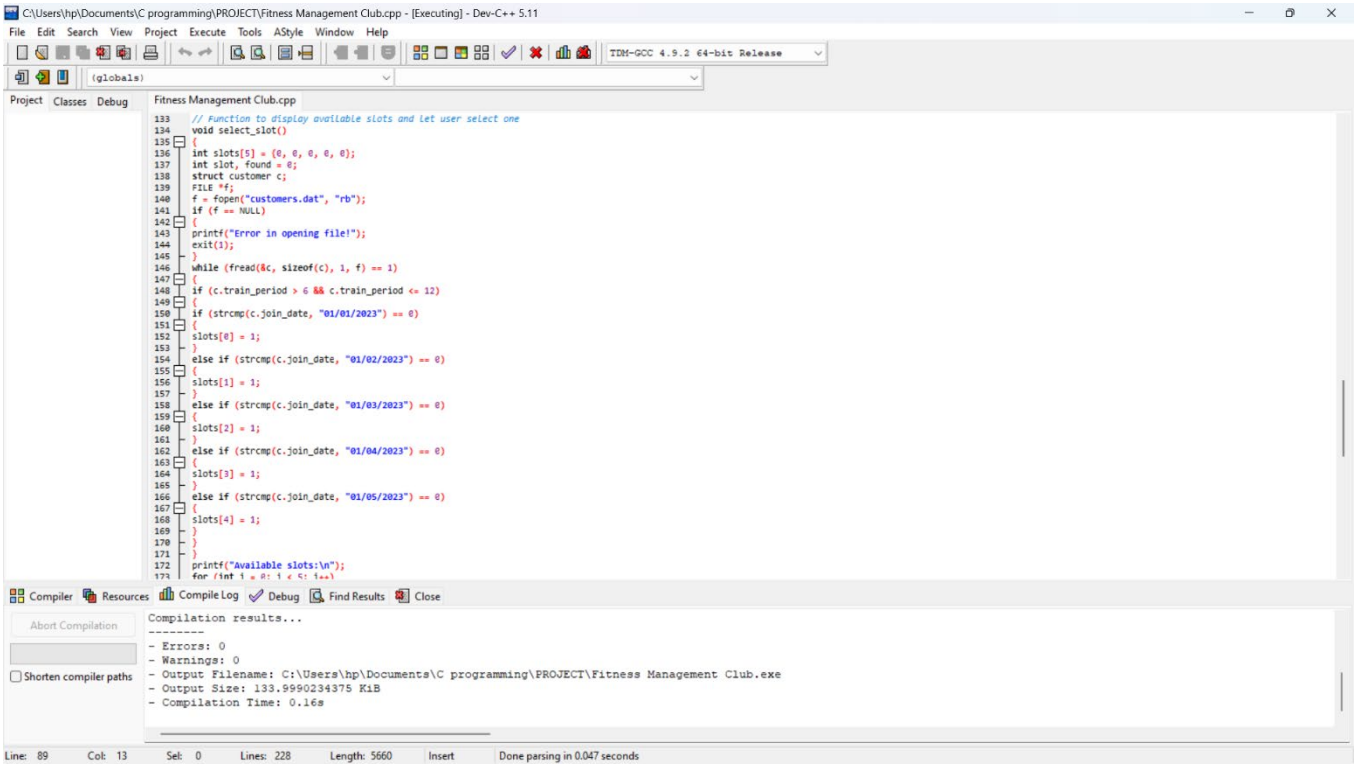

C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
// Function to delete customer record from file
void delete_record()
{
char mobile[11];
int found = 0;
struct customer c;
FILE *f, *t;
f = fopen("customers.dat", "rb");
if (f == NULL)
{
printf("Error in opening file!");
exit(1);
}
t = fopen("temp.dat", "wb");
if (t == NULL)
{
printf("Error in opening file!");
fclose(f);
exit(1);
}
printf("Enter mobile number of customer to be deleted: ");
scanf("%s", mobile);
while (fread(&c, sizeof(c), 1, f) == 1)
{
if (strcmp(c.mobile, mobile) != 0)
fwrite(&c, sizeof(c), 1, t);
else
found = 1;
}
if (found == 1)
printf("Record deleted successfully!\n");
else
printf("Record not found!\n");
fclose(f);
fclose(t);
remove("customers.dat");
rename("temp.dat", "customers.dat");
}
// Function to display all customer records from file
void display_records()
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...

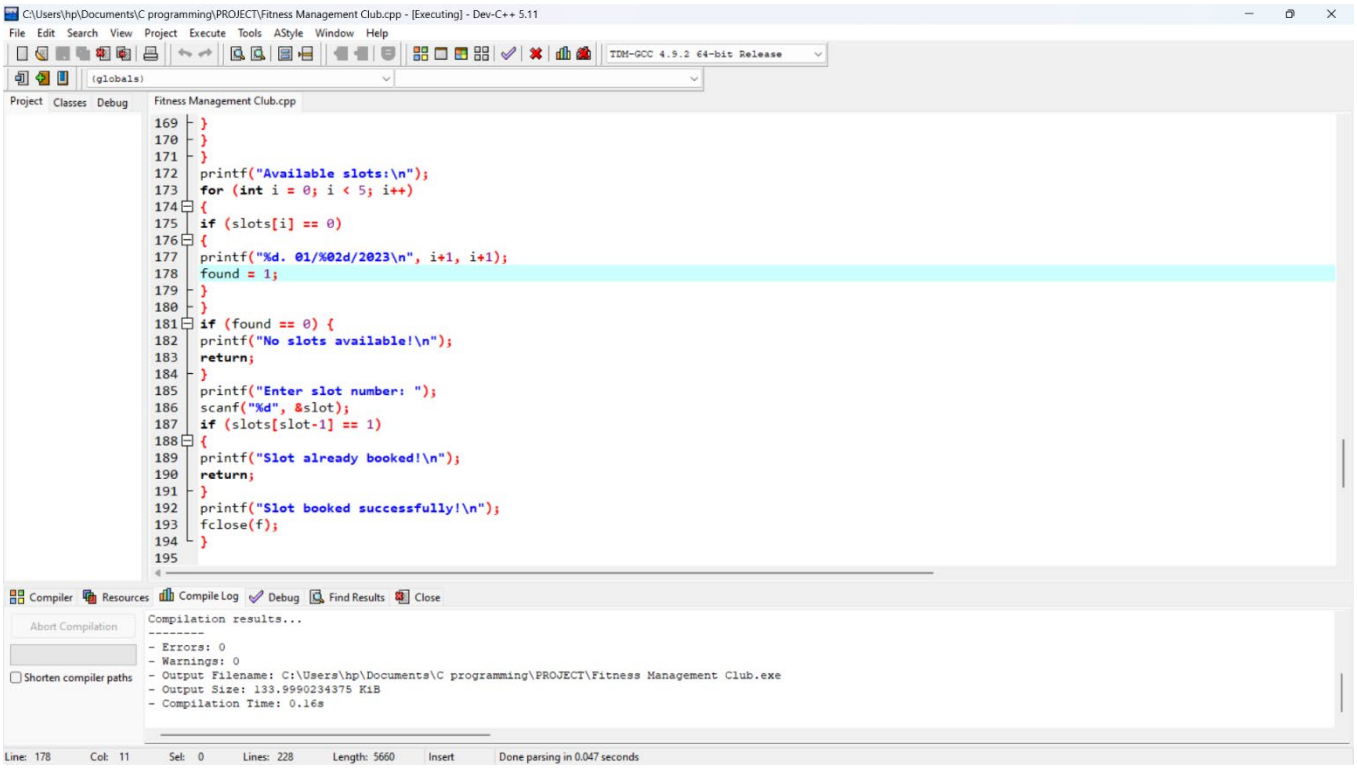
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.exe
- Output Size: 133.9990234375 KiB
- Compilation Time: 0.16s
Line: 1 Col: 1 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds


C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
// Function to display all customer records from file
void display_records()
{
struct customer c;
FILE *f;
f = fopen("customers.dat", "rb");
if (f == NULL)
{
printf("Error in opening file!");
exit(1);
}
while (fread(&c, sizeof(c), 1, f) == 1)
{
printf("\n==========\n");
printf("Name: %s\nMobile Number: %s\nBMI: %f\nAddress: %s\nPackage Details: %s\nJoining Date: %s\nTraining Period: %d months\n\n", c.name, c.mobile, c.bmi, c.address, c.package, c.join_date, c.train_period);
printf("\n==========\n");
}
fclose(f);
}
// Function to search for customer record by mobile number, name, or id
void search_record()
{
char search_query[50];
int found = 0;
struct customer c;
FILE *f;
f = fopen("customers.dat", "rb");
if (f == NULL) {
printf("Error in opening file!");
exit(1);
}
printf("Enter mobile number, name or ID to search: ");
scanf("%s", search_query);
while (fread(&c, sizeof(c), 1, f) == 1)
{
if (strcmp(c.mobile, search_query) == 0 || strcmp(c.name, search_query) == 0)
{
printf("Name: %s\nMobile Number: %s\nBMI: %f\nAddress: %s\nPackage Details: %s\nJoining Date: %s\nTraining Period: %d months\n\n", c.name, c.mobile, c.bmi, c.address, c.package, c.join_date, c.train_period);
found = 1;
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...

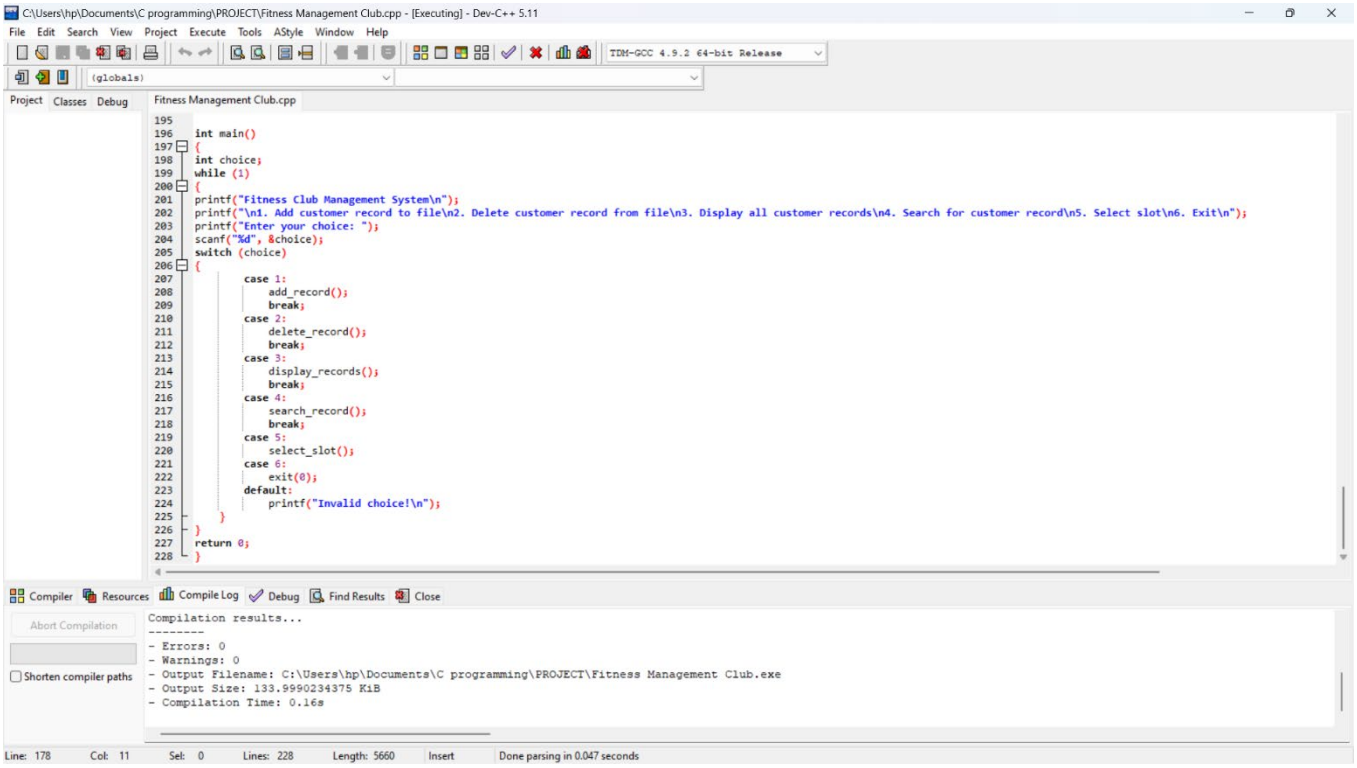
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.exe
- Output Size: 133.9990234375 KiB
- Compilation Time: 0.16s
Line: 89 Col: 13 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds


C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
// Function to search for customer record by mobile number, name, or id
void search_record()
{
char search_query[50];
int found = 0;
struct customer c;
FILE *f;
f = fopen("customers.dat", "rb");
if (f == NULL) {
printf("Error in opening file!");
exit(1);
}
printf("Enter mobile number, name or ID to search: ");
scanf("%s", search_query);
while (fread(&c, sizeof(c), 1, f) == 1)
{
if (strcmp(c.mobile, search_query) == 0 || strcmp(c.name, search_query) == 0)
{
printf("Name: %s\nMobile Number: %s\nBMI: %f\nAddress: %s\nPackage Details: %s\nJoining Date: %s\nTraining Period: %d months\n\n", c.name, c.mobile, c.bmi, c.address, c.package, c.join_date, c.train_period);
found = 1;
}
}
if (found == 0)
printf("Record not found!\n");
fclose(f);
}
// Function to display available slots and let user select one
void select_slot()
{
int slots[5] = {0, 0, 0, 0, 0};
int slot, found = 0;
struct customer c;
FILE *f;
f = fopen("customers.dat", "rb");
if (f == NULL)
{
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.exe
- Output Size: 133.9990234375 KiB
- Compilation Time: 0.16s
Line: 89 Col: 13 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds


C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
// Function to display available slots and let user select one
void select_slot()
{
int slots[5] = {0, 0, 0, 0, 0};
int slot, found = 0;
struct customer c;
FILE *f;
f = fopen("customers.dat", "rb");
if (f == NULL)
{
printf("Error in opening file!");
exit(1);
}
while (fread(&c, sizeof(c), 1, f) == 1)
{
if (c.train_period > 6 && c.train_period <= 12)
{
if (strcmp(c.join_date, "01/01/2023") == 0)
{
slots[0] = 1;
}
else if (strcmp(c.join_date, "01/02/2023") == 0)
{
slots[1] = 1;
}
else if (strcmp(c.join_date, "01/03/2023") == 0)
{
slots[2] = 1;
}
else if (strcmp(c.join_date, "01/04/2023") == 0)
{
slots[3] = 1;
}
else if (strcmp(c.join_date, "01/05/2023") == 0)
{
slots[4] = 1;
}
}
}
printf("Available slots:\n");
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.exe
- Output Size: 133.9990234375 KiB
- Compilation Time: 0.16s
Line: 89 Col: 13 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds

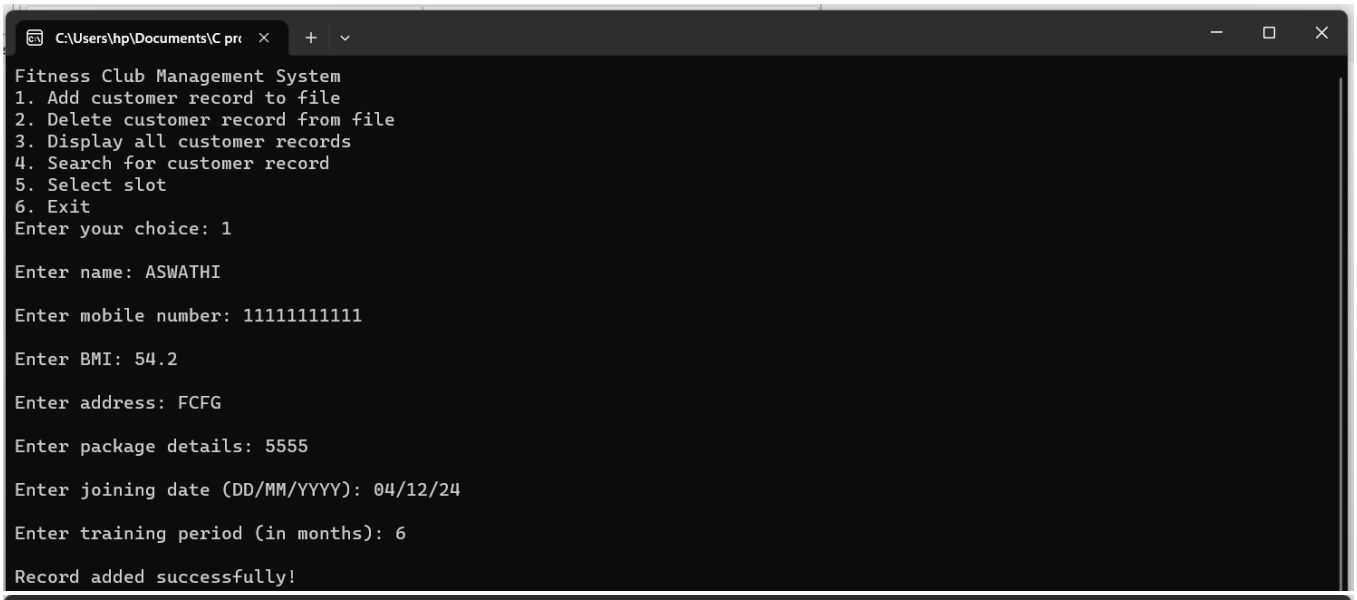

C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
169 }
170 }
171 }
172 printf("Available slots:\n");
173 for (int i = 0; i < 5; i++)
174 {
175 if (slots[i] == 0)
176 {
177 printf("%d. 01/%02d/2023\n", i+1, i+1);
178 found = 1;
179 }
180 }
181 if (found == 0) {
182 printf("No slots available!\n");
183 return;
184 }
185 printf("Enter slot number: ");
186 scanf("%d", &slot);
187 if (slots[slot-1] == 1)
188 {
189 printf("Slot already booked!\n");
190 return;
191 }
192 printf("Slot booked successfully!\n");
193 fclose(f);
194 }
195
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...

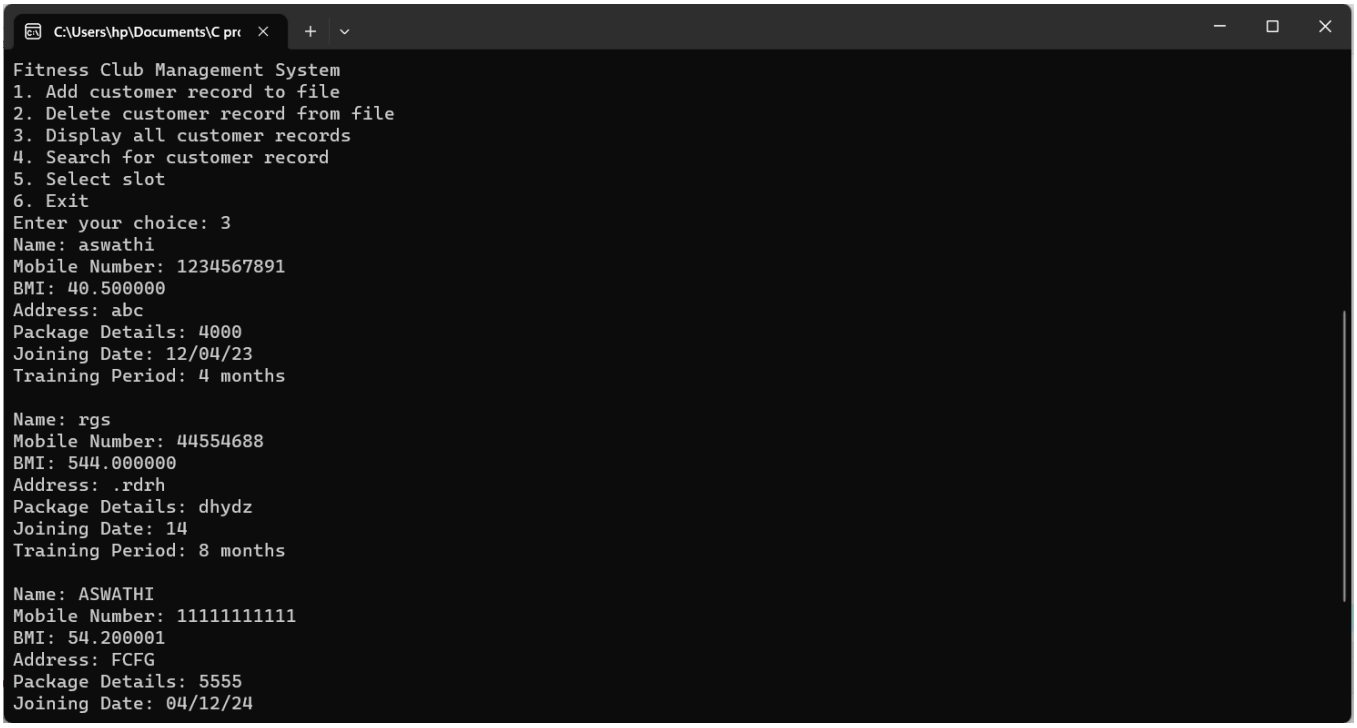
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.exe
- Output Size: 133.9990234375 KiB
- Compilation Time: 0.16s
Line: 178 Col: 11 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds

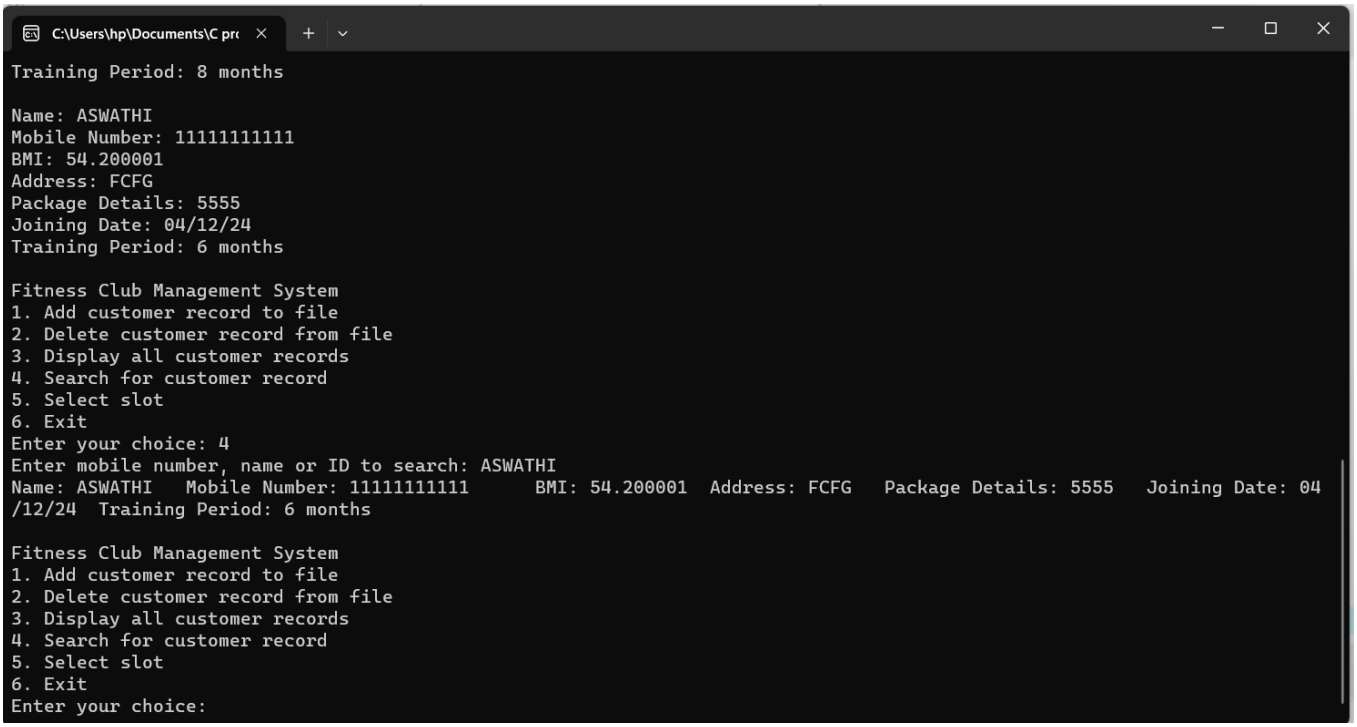

C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Fitness Management Club.cpp
195
196 int main()
197 {
198 int choice;
199 while (1)
200 {
201 printf("Fitness Club Management System\n");
202 printf("\n1. Add customer record to file\n2. Delete customer record from file\n3. Display all customer records\n4. Search for customer record\n5. Select slot\n6. Exit\n");
203 printf("Enter your choice: ");
204 scanf("%d", &choice);
205 switch (choice)
206 {
207 case 1:
208 add_record();
209 break;
210 case 2:
211 delete_record();
212 break;
213 case 3:
214 display_records();
215 break;
216 case 4:
217 search_record();
218 break;
219 case 5:
220 select_slot();
221 case 6:
222 exit(0);
223 default:
224 printf("Invalid choice!\n");
225 }
226 }
227 return 0;
228 }
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...

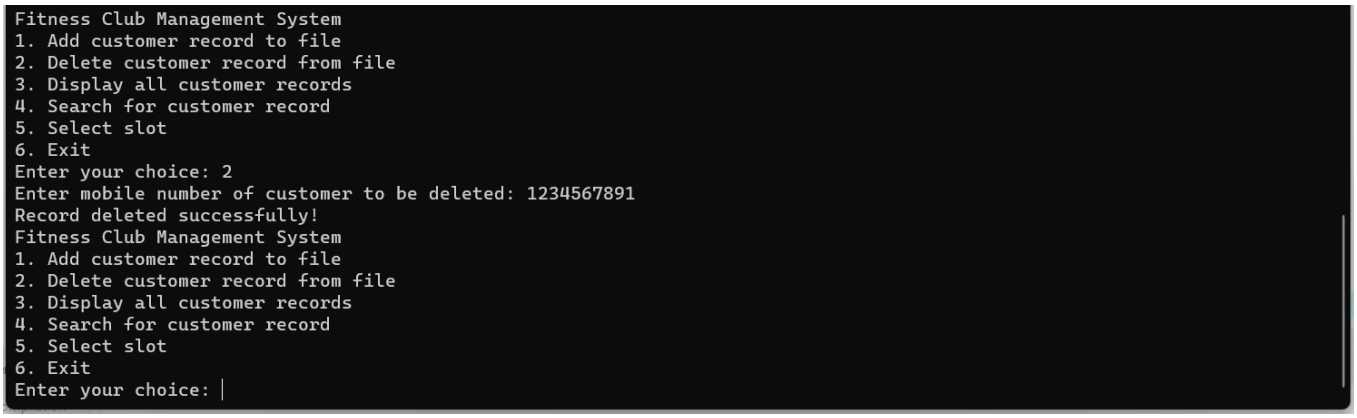
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\hp\Documents\C programming\PROJECT\Fitness Management Club.exe
- Output Size: 133.9990234375 KiB
- Compilation Time: 0.16s
Line: 178 Col: 11 Sel: 0 Lines: 228 Length: 5660 Insert Done parsing in 0.047 seconds

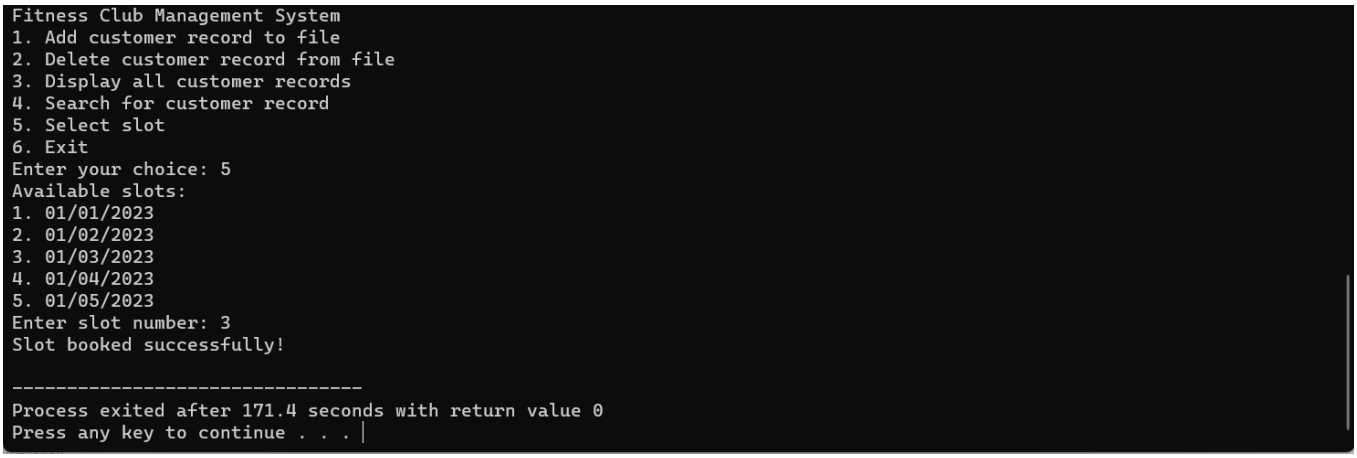
OUTPUT:


C:\Users\hp\Documents\C pr
Fitness Club Management System
1. Add customer record to file
2. Delete customer record from file
3. Display all customer records
4. Search for customer record
5. Select slot
6. Exit
Enter your choice: 1
Enter name: ASWATHI
Enter mobile number: 11111111111
Enter BMI: 54.2
Enter address: FCFG
Enter package details: 5555
Enter joining date (DD/MM/YYYY): 04/12/24
Enter training period (in months): 6
Record added successfully!


C:\Users\hp\Documents\C pr
Fitness Club Management System
1. Add customer record to file
2. Delete customer record from file
3. Display all customer records
4. Search for customer record
5. Select slot
6. Exit
Enter your choice: 3
Name: aswathi
Mobile Number: 1234567891
BMI: 40.500000
Address: abc
Package Details: 4000
Joining Date: 12/04/23
Training Period: 4 months
Name: rgs
Mobile Number: 44554688
BMI: 544.000000
Address: .rdrh
Package Details: dhydz
Joining Date: 14
Training Period: 8 months
Name: ASWATHI
Mobile Number: 11111111111
BMI: 54.200001
Address: FCFG
Package Details: 5555
Joining Date: 04/12/24


C:\Users\hp\Documents\C pr
Training Period: 8 months
Name: ASWATHI
Mobile Number: 11111111111
BMI: 54.200001
Address: FCFG
Package Details: 5555
Joining Date: 04/12/24
Training Period: 6 months
Fitness Club Management System
1. Add customer record to file
2. Delete customer record from file
3. Display all customer records
4. Search for customer record
5. Select slot
6. Exit
Enter your choice: 4
Enter mobile number, name or ID to search: ASWATHI
Name: ASWATHI Mobile Number: 11111111111 BMI: 54.200001 Address: FCFG Package Details: 5555 Joining Date: 04
/12/24 Training Period: 6 months
Fitness Club Management System
1. Add customer record to file
2. Delete customer record from file
3. Display all customer records
4. Search for customer record
5. Select slot
6. Exit
Enter your choice:


Fitness Club Management System
1. Add customer record to file
2. Delete customer record from file
3. Display all customer records
4. Search for customer record
5. Select slot
6. Exit
Enter your choice: 2
Enter mobile number of customer to be deleted: 1234567891
Record deleted successfully!
Fitness Club Management System
1. Add customer record to file
2. Delete customer record from file
3. Display all customer records
4. Search for customer record
5. Select slot
6. Exit
Enter your choice:


Fitness Club Management System
1. Add customer record to file
2. Delete customer record from file
3. Display all customer records
4. Search for customer record
5. Select slot
6. Exit
Enter your choice: 5
Available slots:
1. 01/01/2023
2. 01/02/2023
3. 01/03/2023
4. 01/04/2023
5. 01/05/2023
Enter slot number: 3
Slot booked successfully!

Process exited after 171.4 seconds with return value 0
Press any key to continue . . .

CONCLUSION:

The Fitness Club Management System is an essential tool for club owners who want to manage their customers' records efficiently. It is an efficient and user-friendly program that simplifies the process of managing the customer's records in a fitness club. The modular design of the program makes it easy to modify and update to meet specific requirements. With the help of this program, club owners can keep track of their customers' records and provide them with better services. With its five modules, the system enables the club owner to add and delete customer records, display customer records, search for specific customer records, and view the club's slot timings. Ultimately, this project provides a user-friendly interface for club owners to manage their fitness club effectively.

THANK YOU