

Hotel Management Database System In C Programming

A PROJECT REPORT

Submitted by

HARINI.K.K (RA2111028010200)

Under the guidance of

DR. V.V. Ramalingam

(Associate Professor, Computer Science and Engineering)

S. R. M NAGAR

Kattankulathur

Kancheepuram District

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

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

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

```
//Method Declaration
```

```
void enter_name();
```

```
void find_name();
```

```
void find_room();
```

```
void checkout_guest();
```

```
void view_bookings();
```

```
//Instance Field Declaration
```

```
FILE* view;
```

```
FILE *fp;
```

```
FILE* enter;
```

```
char admin_entry[20] = {'y'};
```

```
char user_entry[20] = {'y'};
```

```
//structure defined to store name, room, bf and wifi required
```

```
struct hotel
```

```
{
```

```
    char name[20];
```

```
    char room[20];
```

```
    char bf[20];
```

```
    char wifi[20];
```

```
}h;
```

```
//main method
```

```
int main(){
```

```
int a,b,c;
```

```
//Welcome screen
```

[illegible]

```
printf("\n>>>>>>>>>>\tFor ADMIN access enter --> 1 \t\t\t<<<<<<<<<<");
```

```
printf("\n>>>>>>>>>>>>\tFor USER access enter --> 2 \t\t\t\t\t<<<<<<<<<<<<");
```

```
printf("\n>> ");
```

```
scanf("%d",&a);
```

```
switch(a){
```

case 1:{

```
while(admin_entry[0] == 'y'){
```

```
printf("\n>>>>>>>>>>\tTo VIEW booking requests --> 1 \t\t\t<<<<<<<<<<");
```

```
printf("\n>>>>>>>>>>\tTo VIEW booked guests details --> 2 \t\t\t<<<<<<<<<<");
```

```

printf("\n>>>>>>>>>>>>>>>\tTo ENTER new guest --> 3 \t\t\t<<<<<<<<<<<<<<<");

printf("\n>>>>>>>>>>>>>>>\tTo FIND room number of existing guest --> 4
\t<<<<<<<<<<<<<<<");

printf("\n>>>>>>>>>>>>>>>\tTo FIND guest of occupied room --> 5 \t<<<<<<<<<<<<<<<");

printf("\n>>>>>>>>>>>>>>>\tTo CHECKOUT room number of existing guest --> 6
<<<<<<<<<<<<<<<\n");

printf(">> ");

scanf("%d",&b);

switch(b){
    case 1:{
        view_bookings();
        break;
    }
    case 2:{
        FILE* fpi;
        fpi = fopen("hotelnew.txt","r");

        while(fscanf(fpi,"%s %s %s",h.name,h.room,h.wifi) != -1){
            fgets(h.bf, 255, (FILE*)fpi);
            printf("%s %s %s %s \n",h.name,h.room,h.wifi,h.bf);
        }
        break;
    }
    case 3:{
        enter_name();
        break;
    }
    case 4:{
        find_name();
        break;
    }
}

```

```

        case 5:{
            find_room();

            break;

        }
    case 6:{
        checkout_guest();

        break;

    }

    default:{
        printf("\n Wrong entry!");

    }

}

//enter again?

printf("Would you like to continue? (y/n)\n");

scanf("%s",admin_entry);


}

if(strcmp(admin_entry,"n") == 0){
    printf("Exiting...\n");
    printf("\e[1;1H\e[2J");
    printf("\n >Exited<\n\n");
    //system("clear");

    return 0;
}

else{
    printf("Wrong entry!\nExiting...\n");

    return 0;
}

break;
}

```

```

case 2:{

    while(user_entry[0] == 'y'){

        printf("\nHello user and welcome to Hotel!");

        printf("\n>>>>>>>>>>>>>\tTo view available rooms --> 1 \t\t\t<<<<<<<<<<<<<");

            printf("\n>>>>>>>>>>>>>\tTo request booking of room --> 2 \t<<<<<<<<<<<<<");

        printf("\n>> ");

        scanf("%d",&c);

        switch(c){

            case 1:{

                FILE* view;

                view = fopen("rooms.txt", "r");

                printf("Available rooms are:\n");

                while(fscanf(view, "%s", h.room) != -1){

                    printf("%s \n", h.room);

                }

                fclose(view);

                break;

            }

            case 2:{

                printf("Enter your name: ");

                scanf("%s", h.name);

                printf("Enter room to book: ");

                scanf("%s", h.room);

                printf("Enter WiFi plan (1GB/5GB): ");

                scanf("%s", h.wifi);

                printf("Enter if bf (yes/no): ");

                scanf("%s", h.bf);

                enter = fopen("bookings.txt", "a");

```

```

        fprintf(enter,"%s %s %s %s",h.name,h.room,h.wifi,h.bf);
        fprintf(enter,"\n");

        printf("Succesfully requested booking\n");
        fclose(enter);

    }
}

printf("Would you like to continue? (y/n)\n");
scanf("%s",user_entry);

}

if(strcmp(user_entry,"n") == 0){
    printf("Exiting...\n");
    printf("\e[1;1H\e[2J");
    printf("\n >Exited<\n\n");
    //system("clear");
    return 0;
}
else{
    printf("Wrong entry!\nExiting...\n");
    return 0;
}
break;
}
}
}

//method declare to enter name

```

```

void enter_name(){

    FILE *tmp1;

    FILE *tmp2;


    printf("\nEnter guest name --> ");
    scanf("%s",h.name);
    printf("\nEnter their room number --> ");
    scanf("%s",h.room);
    printf("\n Enter wifi plan (1GB/5GB) --> ");
    scanf("%s",h.wifi);
    printf("\nEnter if breakfast is included --> ");
    scanf("%s",h.bf);


    //entering guests
    fp = fopen("hotelnew.txt", "a");

    if( fp == NULL){
        printf("\nFile not found");
        exit(1);
    }
    else{
        fprintf(fp,"%s %s %s %s",h.name,h.room,h.wifi,h.bf);
        printf("\nDone\n");
    }
    fprintf(fp,"\n");


    //
    char remove_guest[20];
    strcpy(remove_guest,h.name);
    char remove_room[20];

```



```

strcpy(remove_room,h.room);

//removing booking
tmp1 = fopen("tmp1.txt","a");
enter = fopen("bookings.txt","r");

while(fscanf(enter,"%s %s %s",h.name,h.room,h.wifi) != -1){
    fgetc(enter);
    fgets(h.bf, 20, (FILE*)enter);
    if(strcmp(h.name,remove_guest) != 0){
        fprintf(tmp1,"%s %s %s %s",h.name,h.room,h.wifi,h.bf);
    }
}
fclose(enter);
fclose(tmp1);
remove("bookings.txt");
rename("tmp1.txt","bookings.txt");
fclose(fp);

//removing room
tmp2 = fopen("tmp2.txt","a");
view = fopen("rooms.txt","r");

while(fscanf(view,"%s",h.room) != -1){
    if(strcmp(h.room,remove_room) != 0){
        fprintf(tmp2,"%s",h.room);
        fprintf(tmp2,"\n");
    }
}
fclose(view);

```

```

fclose(tmp2);

remove("rooms.txt");

rename("tmp2.txt","rooms.txt");


fclose(fp);
}


//method defined to find quest name
void find_name(){
    char buffer[20];
    char entered_name[20];
    int guestFound =0;

    printf("\nEnter guest name to find -->");
    scanf("%s",entered_name);

    fp = fopen("hotelnew.txt","r");
    while(fscanf(fp,"%s %s %s",h.name,h.room,h.wifi) != -1){
        fgets(h.bf, 255, (FILE*)fp);
        if(strcmp(h.name,entered_name) == 0){
            guestFound = 1;
            printf("\n Guest found!");
            printf("\n Name is %s",h.name);
            printf("\n Room number is %s",h.room);
            printf("\n WiFi plan is %s",h.wifi);
            printf("\n Breakfast included %s",h.bf);

        }
    }
    if(guestFound == 0){

```

```

        printf("\nGuest %s not found!\n", entered_name);
    }
    fclose(fp);
}

//method definition for guest checkout
void checkout_guest(){
    char buffer2[20];
    char checkout_name[20];
    char add_room[20];

    FILE *tmp;

    printf("Enter guest too checkout ");
    scanf("%s",checkout_name);

    fp = fopen("hotelnew.txt","r");
    tmp = fopen("tmp.txt","w");

    if(fp == NULL){
        printf("File not found");
        exit(1);
    }
    else{
        while(fscanf(fp,"%s %s %s",h.name,h.room,h.wifi) != -1){
            fgetc(fp);
            fgets(h.bf, 20, (FILE*)fp);
            if(strcmp(h.name,checkout_name) != 0){
                fprintf(tmp,"%s %s %s %s",h.name,h.room,h.wifi,h.bf);
            }
            else{

```

```

        strcpy(add_room,h.room);
    }
}
printf("Room checked out!");
}

```

```

fclose(fp);
fclose(tmp);
remove("hotelnew.txt");
rename("tmp.txt","hotelnew.txt");

```

```

//enter room back into available rooms list
view = fopen("rooms.txt","a");
printf(" add is %s",add_room);
fprintf(view,"%s",add_room);
fclose(view);

}

```

```

//method definition for finding rooms available
void find_room(){
    char buffer[20];
    char entered_room[20];
    int roomFound =0;
    FILE *fp;
    printf("\nEnter room number to find -->");
    scanf("%s",entered_room);

    fp = fopen("hotelnew.txt","r");
    while(fscanf(fp,"%s %s %s",h.name,h.room,h.wifi) != -1){
        fgets(h.bf, 255, (FILE*)fp);
    }
}

```

```

if(strcmp(h.room,entered_room) == 0){
    roomFound = 1;
    printf("\n Room found!");
    printf("\n Name  is %s",h.name);
    printf("\n Room number is %s",h.room);
    printf("WiFi plan is %s",h.wifi);
    printf("\n Breakfast included %s",h.bf);
}
}
if(roomFound == 0){
    printf("\nRoom %s not found!\n", entered_room);
}
fclose(fp);
}

//method defined to view bookings
void view_bookings(){

    enter = fopen("bookings.txt","r");

    while(fscanf(enter,"%s %s %s",h.name,h.room,h.wifi) != -1){
        fgets(h.bf, 255, (FILE*)enter);
        printf("%s %s %s %s \n",h.name,h.room,h.wifi,h.bf);
    }
}

```

```

*****
*                                     *
* -----
*      WELCOME TO HOTEL MUJ
* -----
*                                     *
*                                     *
*                                     *
*      ARNAV AGARWAL
*      199301***
*      SECTION *, EL PROJECT
*****

>>>>>>>>>> For ADMIN access enter --> 1 <<<<<<<<<<<<
>>>>>>>>>> For USER access enter --> 2 <<<<<<<<<<<<
>> 2

Hello user and welcome to Hotel!
>>>>>>>>>> To view available rooms --> 1 <<<<<<<<<<<<
>>>>>>>>>> To request booking of room --> 2 <<<<<<<<<<<<
>> 2
Enter your name: xxxxxxxx
Enter room to book: 2
Enter WiFi plan (1GB/5GB): 1GB
Enter if bf (yes/no): Yes
Succesfully requested booking
Would you like to continue? (y/n)
y

Hello user and welcome to Hotel!
>>>>>>>>>> To view available rooms --> 1 <<<<<<<<<<<<
>>>>>>>>>> To request booking of room --> 2 <<<<<<<<<<<<
>> _

```

```

*****
*                                     *
* -----
*      WELCOME TO HOTEL MUJ
* -----
*                                     *
*                                     *
*                                     *
*      ARNAV AGARWAL
*      199301***
*      SECTION *, EL PROJECT
*****

>>>>>>>>>> For ADMIN access enter --> 1 <<<<<<<<<<<<
>>>>>>>>>> For USER access enter --> 2 <<<<<<<<<<<<
>> 1

>>>>>>>>>> To VIEW booking requests --> 1 <<<<<<<<<<<<
>>>>>>>>>> To VIEW booked guests details --> 2 <<<<<<<<<<<<
>>>>>>>>>> To ENTER new guest --> 3 <<<<<<<<<<<<
>>>>>>>>>> To FIND room number of existing guest --> 4 <<<<<<<<<<<<
>>>>>>>>>> To FIND guest of occupied room --> 5 <<<<<<<<<<<<
>>>>>>>>>> To CHECKOUT room number of existing guest --> 6 <<<<<<<<<<<<

```

