

MINI PROJECT LATEX FILE

TOPIC : Restaurant Booking And Food Odering System

```
\documentclass[30pt,a3paper]{article}
\usepackage[hmargin=3.0cm,
vmargin=3cm]{geometry}
\usepackage{listings}
\usepackage{graphicx}
\title{ \huge\textbf{Restaurant Booking And Food Odering System }}
\author{AMAN BAGADIYA ,0801CS211013,\\B.tech 2nd year,SGSITS INDORE}
\date{\today}
\begin{document}
    \maketitle
```

\section{AIM :}

The aim of my project is to make a restaurant management and food odering system via which user can perform following functions :\\

\\

A. USER FUNCTIONS\\

\\

ORDER FOOD\\

BOOK TABLE AT RESTAURANT\\

CANCELL BOOKING\\

GET INFORMATION ABOUT RESTAURANT \\

\\

\\

.\hspace{1cm}B.ADMIN FUNCTIONS\\

\\

ADMIN AUTHENTICATION\\

SET/UPDATE GST RATES\\

GENERATE INVOICE\\

VIEW ALL INVOICES \\

VIEW ALL BOOKINGS \\

\section{Following are the functions used in project:}

\subsection{Welcome_Page() }

This function shows the user functionalities available for him also ask the user about the login choice and then depending upon the user selected choice calls the respective function.

\subsection{Admin_Page() }

This function allows the admin to perform following functions to update GST rate, to view all Invoices, to view all Bookings.

\subsection{Admin_authentication() }

This funtion is used to verify admin by the user entered password nd admin id with the already existing one.

\subsection{User_Registration_Page() }

This funtion takes the user details - name, phone number, address, email id as input and stores it into the structure.

\subsection{options() }

This function displays the options available to the user as follows

1.Get Information About Restaurant

2.Order Food:>

3.Book table:>

4.Cancel your Booking:>

5.Exit:>

and gives the choice to select from the available option and calls the respective function.

```
\subsection{place\_order() }
```

This function displays other 2 functions which are menu and city display.

```
\subsection{city\_display() }
```

This is functions gives the user the choice of the city and ask him to select from the given choices.

```
\subsection{City\_choice(int city\_choice) }
```

This function intakes the parameter as city choice selected by the user and calls the respective city function.

```
\subsection{INDORE() }
```

This function shows the list of the available restaurants of Indore city.

```
\subsection{BHOPAL() }
```

This function shows the list of the available restaurants of Bhopal city,

```
\subsection{MUMBAI() }
```

This function shows the list of the available restaurants of Mumbai city.

```
\subsection{INDORE\_info(int restaurant\_name) }
```

This function shows the information to user about selected restaurant of Indore city.

```
\subsection{BHOPAL\_info(int restaurant\_name) }
```

This function shows the information to user about selected restaurant of Bhopal city.

```
\subsection{MUMBAI\_info(int restaurant\_name) }
```

This function shows the information to user about selected restaurant of Mumbai city.

```
\subsection{void menu() }
```

This function shows the menu and takes the item no. and its quantity as input and calls the bill function for bill generation.

```
\subsection{void Bill() }
```

Its function is to compute the bill and ask the user to add more items and also to call the invoice function to print invoice.

```
\subsection{void Invoice() }
```

Its function to display the invoice on screen in proper format and also to call save_details().

```
\subsection{save\_details() }
```

This function is responsible for creating a file and saving the user details in it.

```
\subsection{Booking() }
```

This function is used to display the list of the vacant and occupied seats and also to book a table and display the successfull booking information.

```
\subsection{save\_booking\_details() }
```

This function is responsible for creating a file and saving the booking details in it.

```
\subsection{Cancel\_Booking() }
```

This function is responsible for cancelling the booking by making that array element as 0.

```
\subsection{void print\_invoice() }
```

This function is used to print the invoice from the file on screen.

```
\subsection{void view\_bookings() }
```

This function is used to print the bookings from the file on screen.

```
\section{Description}
```

This mini project program is created with the objective to ease the food odering, restaurant booking and bill generation system it contains about 23 functions all of them perform different different work. It is created in 2 languages C and C++, Here i have majorly used the concept of array ,structures, file handeling,and loops.

```
\section{Profilier report}
```

```
\begin{center}
```

```

\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_124510.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_124528.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_124544.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_124603.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_124623.png}

\end{center}
\vspace{0.8mm}

\section{GBD Activities}

\vspace{0.8mm}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_013525.png}

\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_013558.png}

\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot_20230130_013614.png}

\section{\Huge{Code with Comment}}
\begin{lstlisting}
/*adding all the necessary header files*/
#include<stdio.h>
// #include<conio.h>
#include<stdlib.h>
#include<string.h>
/*Declaration of all the necessary functions */
void save\_booking\_details();
void Booking();
void Cancel\_Booking();
void menu();
void BHOPAL\_info();
void MUMBAI\_info();
void INDORE\_info();
void INDORE();
void City\_choice(int);
void city\_display();
void place\_order();
int Admin\_authentication();
void Admin\_Page();
void User\_Registration\_Page ();
void Welcome\_Page();
void INDORE();
void BHOPAL();
void MUMBAI();
void options();
void save\_details();
void invoice();
void Bill();
void print\_invoice();
void view\_bookings();
/*Declaration of structure*/
struct User {
    char User\_Name[20];
    long long User\_contactdetails;
    char User\_address[50];
    char User\_emial\_id[30];
};
/*Declaration of variable of struct type*/
struct User user1;
/*Declaration of the other variables */
int Table\_no;
int gstRate;
int item;
int quantity;

```

```

float grandTotal;
int restaurant\_name;
int option;
int count=0;
/*Declaration of array to store data*/
char food\_name[50];
int item\_array[31] = {0};
char menu\_array[][35] = {"0", "Paneer Angara", "Paneer Pasasnda", "Paneer Lababdar", "Paneer
Tawa Masala", "Paneer Toofani", "Paneer Handi", "Paneer Kadai", "Paneer Tikka Masala",
"Paneer Butter Masala plus", "Paneer Palak", "Cheese Nan", "Hariyali Nan", "Kashmiri Nan",
"Plain Nan", "Handi Dum Parda Biryani", "Special Veg. Biryani", "Veg. Hyderabad Biryani",
"Veg. Pulao (Plain or Masala)", "Peas Pulao", "Mushroom Onion Pulao", "Cheese Garlic Bread",
"Chesse Corn Chilly Toast", "Veg. Cutlet", "Aloo Tikki", "Veg. Cheese Sandwich", "Mysore
Masala Dosa", "Mysore Plain Dosa", "Hyderabad Masala Dosa", "Hyderabad Plain Dosa", "Sunday
Special Ice-cream", "Almond Carnival Ice-cream", "Kesar Pista Ice-cream", "Kaju Draksh Ice-
cream", "Butter Scotch Ice-cream"};
int quantity\_array[35] = {0};
int Table\_array[10] = {0};
int main (){
    printf ("Welcome to restaurant booking and food odering system developed by Aman
Bagadiya");
    gstRate = 18; //default gst rates
    Welcome\_Page(); //calling of function ----> Welcome\_Page();
}
//defining of the function Welcome\_Page
void Welcome\_Page(){
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    printf ("\t\t\t\t\tWelcome to Restaurant Booking and Food odering system developed by
Aman Bagadiya under the guidance of Prof.Surendra Gupta Sir.\n");
    printf ("\t\t\t\t\t-----\n\n");
    printf ("\t\t\t\t\t
::::::::::::::::::::::::::::::::::::::::::::::::::\n");
    printf ("\t\t\t\t\t:: THIS MINI PROJECT AIMS TO PERFORM
VARIOUS FUNCTIONS ::\n");
    printf ("\t\t\t\t\t
::::::::::::::::::::::::::::::::::::::::::::::::::\n\n");
    printf ("\t\t\t\t\tA.USER FUNCTIONS :\n");
    printf ("\t\t\t\t\t-> ORDER FOOD :\n");
    printf ("\t\t\t\t\t-> BOOK TABLE AT
RESTAURANT :\n");
    printf ("\t\t\t\t\t-> CANCELL BOOKING
:\n");
    printf ("\t\t\t\t\t-> GET INFORMATION
ABOUT RESTAURANT :\n\n");
    printf ("\t\t\t\t\tB.ADMIN FUNCTIONS :\n");
    printf ("\t\t\t\t\t-> ADMIN AUTHENTICATION
:\n");
    printf ("\t\t\t\t\t-> SET/UPDATE GST RATES
:\n");
    printf ("\t\t\t\t\t-> GENERATE INVOICE
:\n");
    printf ("\t\t\t\t\t-> VIEW ALL INVOICES
:\n");
    printf ("\t\t\t\t\t-> VIEW ALL BOOKINGS
:\n");
    int Login\_option; //defining of the variable
    printf ("Login: \n");
    printf ("For User login press 1: \n");
    printf ("For Admin login press 2: \n");
    printf ("To exit press 0: \n");
    scanf ("%d", &Login\_option);
    //HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE USER CHOICE
    switch(Login\_option) {

```

```

        case 1: User\_Registration\_Page(); //calling of function ---->
User\_Registration\_Page()
            break; // break statement is used to terminate switch statement execution,

        case 2: Admin\_Page(); //calling of function ----> Admin\_Page();
            break; // break statement is used to terminate switch statement execution,

        case 0: printf ("You have successfully exited from program \n\n ");
            printf ("This project is made by Aman Bagadiya under the guidance of Mr
Surendra Gupta Sir \n\n");
            break; // break statement is used to terminate switch statement execution,

        default: printf("You have entered an invalid choice...Exiting...\n \n");
            exit(0); //exit(0) this function is used to successfully terminate and exit from the
program
    }
}
//defining of the function Admin\_Page()
void Admin\_Page(){
    if (Admin\_authentication() == 1) { // calling of the Admin\_authentication() function
to verify the admin
        system("cls"); //this is the function which is defined in stdlib.h and it is used to
clear terminal screen
        printf("Current GST rate %d percent : \n", gstRate);
        int select;//defining of the variable
        printf("Press 1 to update GST rate:\n");
        printf("Press 2 to view all Invoices : \n");
        printf("Press 3 to view all Bookings : \n");
        printf("To exit press 0: \n");
        scanf("%d",&select);
        //HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE ADMIN
CHOICE
        switch(select) {
            case 1:
                printf("Enter New GST rates (in percentage): \n");
                scanf("%d",&gstRate);
            case 2:
                print\_invoice(); //calling of function ----> print\_invoice()
            case 3:
                view\_bookings(); //calling of function ----> view\_bookings()
        }
    }
}
//defining of the function Admin\_authentication()
int Admin\_authentication(){
    system("cls"); //This is the function which is defined in stdlib.h and it is used to clear
terminal screen
    int admin\_check = 1;//defining of the variable
    char loginID[] = "Aman";
    char password[] = "Aman";
    char user\_entered\password[11]; //Declaration of the array user\_entered\password to
store login id
    char user\_entered\login[10]; //Declaration of the array user\_entered\login to
store password

    printf("*****\t Welcome to Admin Login Page \t*****\n\n");
    printf("Kindly enter your credentials below : \n\n");
    printf("Enter Admin login ID: ");
    scanf("%s", user\_entered\login);
    printf("Enter Admin password: ");
    scanf("%s", user\_entered\password);
    if(strcmp(loginID , user\_entered\login) != 0) { //Using strcmp function to compare 2
strings loginID and user\_entered\login
        admin\_check = 0;
    }
}

```

```

if(strcmp(password, user\_entered\_password) != 0) { //Using strcmp function to compare 2
strings password and user\_entered\_password
    admin\_check = 0;
}
if(admin\_check == 0) {
    printf("sorry You Have Entered Wrong credentials.\n");
    printf("Try Again\n");
    return 0;
}else {
    return 1; // if admin is successfully verified it will return 1
}
}

void User\_Registration\_Page(){
    //defining of the function User\_Registration\_Page()
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    printf("Enter Your Name\n");
    fflush(stdin); //it is used to clear the buffer
    scanf("%[^\n]s", &user1.User\_Name);
    printf("Enter Your Phone Number\n");
    scanf("%lld", &user1.User\_contactdetails);
    printf("Enter Your Email-id\n");
    fflush(stdin); //it is used to clear the buffer
    scanf("%[^\n]s", &user1.User\_emial\_id);
    printf("Enter Address\n");
    fflush(stdin); //it is used to clear the buffer
    scanf("%[^\n]s", &user1.User\_address);
    printf("Your details registered successfully\n");
    options(); //calling of function ----> options
}

void options(){
    do{
        system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
        printf("\n
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::");
        printf("\n
::                                     ::             !!!!!!!!!!!!!!!welcome!!!!!!!!!!!!!!
::");
        printf("\n
::                                     ::
::");
        printf("\n
::                                     ::      ~> Please select from below options:
::");
        printf("\n
::                                     ::
::");
        printf("\n
::                                     ::          1.Get Information About Restaurant :>
::");
        printf("\n
::                                     ::          2.Order Food:>
::");
        printf("\n
::                                     ::          3.Book table:>
::");
        printf("\n
::                                     ::          4.Cancel your Booking:>
::");
        printf("\n
::                                     ::          5.Exit:>
::");
        printf("\n
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n\n");
        printf("PLEASE SELECT FROM THE ABOVE OPTIONS:\n");
        scanf("%d", &option);
        //HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE ADMIN
CHOICE
        switch (option){
            case 1: city\_display();
                    break; // break statement is used to terminate switch statement
execution,

```

```

        case 2 :
            place\_order();
            break; // break statement is used to terminate switch statement
execution,
        case 3 : {
            city\_display();
            Booking();
        }
        case 4 : {
            city\_display();
            Cancel\_Booking();
        }
        case 5 : {
            printf("\nExiting...\n");
            exit(0); //exit(0) this function is used to successfully terminate and
exit from the program
        }
        default :{
            printf("Please Enter A Valid Choice");
        }
    }
}while(1);
}
void place\_order(){
    city\_display();
    menu();
}
void city\_display(){
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    printf("::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n");
    printf("::                                                                    ::\n");
    printf("::      ~   Please select your city from below list: ::\n");
    printf("::                                                                    ::\n");
    printf("::      >          1.INDORE                                ::\n");
    printf("::      >>         2.BHOPAL                                ::\n");
    printf("::      >>>        3.MUMBAI                                ::\n");
    printf("::                                                                    ::\n");
    printf("::                                                                    ::\n");
    printf("::                                                                    ::\n");
    printf("::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n\n");
    //getchar();
    printf("Enter your choice by entering the serial number of city : ");
    int city\_choice;
    scanf("%d",&city\_choice);
    City\_choice(city\_choice); //calling of function ----> City\_choice(city\_choice)
}
void City\_choice(int city\_choice){
    switch (city\_choice){
        //HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE
ADMIN CHOICE
        case 1 : {
            INDORE();
            break; // break statement is used to terminate switch statement execution,
        }
        case 2 : {
            BHOPAL();
            break; // break statement is used to terminate switch statement execution,
        }
        case 3 : {
            MUMBAI();
            break; // break statement is used to terminate switch statement execution,
        }
    }
}
}

```

```
void INDORE() {
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear terminal screen
    printf("::::::::::::::::::::::::::::::::::::::::::::\n");
    printf(":: ~> Please select restaurant below list: ::\n");
    printf("::\n");
    printf(":: Name of Restaurant Ratings ::\n");
    printf("::\n");
    printf(":: 1.Kebabsville ** ::\n");
    printf(":: 2.Ginger Ganesha *** ::\n");
    printf(":: 3.Shree Chotiwalla * ::\n");
    printf(":: 4.Village ** ::\n");
    printf(":: 5.Vidorra * ::\n");
    printf("::\n");
    printf("::::::::::::::::::::::::::::::::::::::::::::\n\n");
    //getchar();
    printf("Enter your choice by entering the serial number of restaurant : \n");
    int restaurant_name;
    scanf("%d",&restaurant_name);
    if (option == 1)
        INDORE_info(restaurant_name); //calling of function ----> INDORE_info
    printf("\n\n");
}

void BHOPAL() {
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear terminal screen
    printf("::::::::::::::::::::::::::::::::::::::::::::\n");
    printf(":: ~> Please select restaurant below list: ::\n");
    printf("::\n");
    printf(":: Name of Restaurant Ratings ::\n");
    printf("::\n");
    printf(":: 1.Manohar ** ::\n");
    printf(":: 2.Cafe Chokolade ** ::\n");
    printf(":: 3.Greek Food & Beyond * ::\n");
    printf(":: 4.Bapu Ki Kutia ** ::\n");
    printf(":: 5.Al-Beik *** ::\n");
    printf("::\n");
    printf("::::::::::::::::::::::::::::::::::::::::::::\n\n");
    //getchar();
    printf("Enter your choice by entering the serial number of restaurant : \n");
    int restaurant_name;
    scanf("%d",&restaurant_name);
    if (option == 1)
        BHOPAL_info(restaurant_name); //calling of function ----> BHOPAL_info
    printf("\n\n");
}

void MUMBAI() {
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear terminal screen
    printf("::::::::::::::::::::::::::::::::::::::::::::\n");
    printf(":: ~> Please select restaurant below list: ::\n");
    printf("::\n");
    printf(":: Name of Restaurant Ratings ::\n");
    printf("::\n");
    printf(":: 1.Bayroute Restaurant ** ::\n");
    printf(":: 2. Hakkasan *** ::\n");
    printf(":: 3.Dome Intercontinental * ::\n");
    printf(":: 4.Yauatcha Restaurant ** ::\n");
    printf(":: 5.Pali Village cafe * ::\n");
    printf("::\n");
    printf("::::::::::::::::::::::::::::::::::::::::::::\n\n");
}
```



```

//getchar();
printf("Enter your choice by entering the serial number of restaurant : \n");

scanf("%d",&restaurant\_name);
if (option == 1)
MUMBAI\_info(restaurant\_name); //calling of function ----> MUMBAI\_info
printf("\n\n");

}

void INDORE\_info(int restaurant\_name){
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    switch (restaurant\_name){
        case 1 : {
            printf("LOCATION : Sayaji Indore, H/1 , Scheme No.54, Vijay Nagar, Indore,
Madhya Pradesh - 452010\n\nPHONE :0731 4006666, 73899 48912\n\nMORE INFORMATION: PRICE RANGE
rs 805 - rs 1,208\nCUISINES =Indian, Barbecue, Asian\nSPECIAL DIETS =Vegetarian Friendly,
Vegan Options, Halal, Gluten Free Options Dinners can now become even more special with a
laid back and relaxed time relishing the buffet at a Kebabsville! Sayaji Indore's buffet
style restaurant- Kebabsville is a class apart! Set alongside the pool with lavish seating
arrangement, you can have a relaxing buffet full of mouth-watering kebabs, grilled veggies
and North Indian food dishes. Variety of kebabs in vegetarian and non-vegetarian options,
made by our expert chefs are sure to win your hearts. You can eat the kebabs to your heart's
content and also enjoy the North India dishes alongside. Relish in our lavishly spread buffet
service and make your evenings worth remembering.\n");
            break; // break statement is used to terminate switch statement execution,
        }
        case 2 : {
            printf("LOCATION LG-12,13, DM Tower, 21/1, Lala Banarasilal Dawar Marg, Race
Course Road, Near Zanjeerwala Chowraha, New Palasia, Indore, Madhya Pradesh 452001,
India\n\nPhone=+91-731-3301441\n\nMore information-A good and lively ambience with amazing
South Indian and Chinese dishes, Ginger Ganesha might just steal your heart with the sheer
simplicity of its. If you want to have excellent dosas and super delicious vegetarian biryani
then this is the place to be. Do try the Bisebele Bath and the Mexican Roll Dosa. Not only
the food, but the restaurant is also known for its quick service which is an added star. This
is also a popular place to dine at with the family because of the reasonable price at which
you can eat with your heart and the cozy environment that gives you the comfort of being in
the company of your loved ones\n\n");
            break; // break statement is used to terminate switch statement execution,
        }
        case 3 : {
            printf("LOCATION: 8 B, Raunak Plaza, Opposite Nath Mandir, South Tukoganj,
Indore, Madhya Pradesh 452001, India\n\nPHONE: +91-731-2528586\n\nMore Information: One of
the most popular eating joints in the city, Shree Chotiwalla serves you with delicious veg
Indian food. The wide variety of dishes in the menu might confuse you a bit. That's a good
thing, right? There is a booth seating and a family table seating arrangement that are
available for different purposes and guests. Not to miss is the veg thali that has a huge
range of dishes that are served on a single platter. It's quite a treat in every sense of the
word. There are also dishes for the stricity\_display followers of Jainism available at this
restaurant which essentially means vegetarian dishes that do not even have onion or garlic in
it\n\n");
            break; // break statement is used to terminate switch statement execution,
        }
        case 4 : {
            printf("LOCATION: 5th Floor, Central Mall, RNT Marg, Indore, Madhya Pradesh
452001, India\n\nPHONE: +91-731-4006617|+91-78699-10313\n\nMore Information: Sit on a chair,
on a khatiya or with a cozy takiya, Village is a sure shot dream to dine at. Serving
Rajasthani and Gujarati cuisines as their specialty, the place ensures a fulfilling
experience for every foodie who enters the place. The thatched stalls serving street food
from various parts of the country add another world charm to the place. The truck with a
buffet arrangement and the puppet show are amazing. If you are the kind of person who does
not get satisfied with just the food then the overall experience here is going to stay with
you for a long time.\n");
            break; // break statement is used to terminate switch statement execution,
        }
    }
}

```

```

        case 5 : {
            printf("LOCATION : New Palasia, Indore\n\nCONTACT DETAILS:12345678 \n\nMORE
INFORMATION :CUISINES Indian, Asian SPECIAL DIETS Vegan Options, Vegetarian Friendly MEALS
Dinner, Lunch.\n");
            break; // break statement is used to terminate switch statement execution,
        }
    }
    printf("\nPress any key to continue...");
    //getch(); //Here //getch will take any character input and then this function will be
    terminated and the next line of the calling function will execute
}
void BHOPAL\_info(int restaurant\_name){
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
    terminal screen
    //HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE USER
    CHOICE
    switch (restaurant\_name){
        case 1 : {
            printf("132, Zone 1, Maharana Pratap Nagar, Bhopal\nStreet Food, South
Indian, Fast Food, Desserts, North Indian, Chinese, Bakery, Mithai \n8:30am to 10:30pm (Mon-
Sun) \n300\n");
        }
        case 2 : {
            printf("\n206, Zone 2, Near Arya Bhavan, Maharana Pratap Nagar, Bhopal,\n
India Cafe, Bakery \n11 AM to 11 PM \nINR 450\n \n");
            break; // break statement is used to terminate switch statement execution,
        }
        case 3 : {
            printf("Third Floor, DB City Mall, Maharana Pratap Nagar, Bhopal, \nIndia
Greek, Mediterranean \n12 Noon to 10:30 PM \nINR 1000\n");
            break; // break statement is used to terminate switch statement execution,
        }

        case 4 : {
            printf(" 123, Jyoti Shopping Centre, Behind Jyothi Talkies Zone-I, Maharana
Pratap Nagar,Bhopal, Madhya Pradesh 462011\n North-Indian, Chinese, South-Indian, Snacks \n7
AM to 11 PM \nINR 500 \n");
            break; // break statement is used to terminate switch statement execution,
        }

        case 5 : {
            printf("Regiment Rd,Badabagh, Shahjahanabad,Bhopal, Madhya Pradesh 462001\n
Snacks \n1:30 PM to 10 PM\n INR 300\n");
            break; // break statement is used to terminate switch statement execution,
        }
    }
    printf("\nPress any key to continue...");
    //getch(); //Here //getch will take any character input and then this function will be
    terminated and the next line of the calling function will execute
}
void MUMBAI\_info(int restaurant\_name){
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
    terminal screen
    //HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE USER
    CHOICE
    switch (restaurant\_name){
        case 1 : {
            printf("Location: Bayroute Juhu, 14, Silver Beach Estate Opposite Juhu Post
Office, AB Nair Road, Juhu, Mumbai.\n Foods to try: Harissa Chicken Kebab, Kebab Istanbul,
Baklava, Gold Souk are some of the must-try dishes.\n Price for two: INR 3000 \n");
            break; // break statement is used to terminate switch statement execution,
        }
        case 2 : {

```

```

        printf("Location: Hakkasan, 2nd Floor, Krystal Building, Waterfield Road,
Bandra West, Mumbai. \nFoods to try: Everything in Cantonese cuisine menu is so yum and do
not forget to try the dumplings. \nPrice for two: INR 2500\n");
        break; // break statement is used to terminate switch statement execution,
    }
    case 3 : {
        printf("Location: Dome Intercontinental 135, Marine Drive, Mumbai, \nFoods to
try: Afghani Chicken, Brownie and cocktails.\n Price for two: INR 3,500\n");
        break; // break statement is used to terminate switch statement execution,
    }
    case 4 : {
        printf("Location: Yauatcha, Raheja Towers, Bandra Kurla Complex, Bandra East,
Mumbai\n Foods to try: Crispy Prawns Cheung Fun, Pork char sui bun are extraordinary. \nPrice
for two: INR 3,500\n");
        break; // break statement is used to terminate switch statement execution,
    }
    case 5 : {
        printf("Location: Pali Village Cafe, 602, Ambedkar Road, Pali Naka, Pali
Hill, Bandra West, Mumbai\n Foods to try: Risotto, Salads, Panna Cotta, Pizzas are worth a
try. \nPrice for two: INR 3000\n");
        break; // break statement is used to terminate switch statement execution,
    }
}
printf("\nPress any key to continue...");
//getch(); //Here //getch will take any character input and then this function will be
terminated and the next line of the calling function will execute
}
void menu(){
    //defining of the function menu()
    printf("\n ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::");
    printf("\n\n
ITEMS.....PRICE \n1 Paneer
Angara.....220 \n2 Paneer Pasasnda
.....210 \n3 Paneer
Lababdar.....215 \n4 Paneer Tawa
Masala.....215 \n5 Paneer Toofani
.....220 \n6 Paneer
Handi.....200 \n7 Paneer
Kadai..... 200 \n8 Paneer Tikka
Masala.....195 \n9 Paneer Butter
Masala.....195 \n10 Paneer
Palak.....188 \n*ROTIES - TANDURI \n11
Cheese Nan .....88 \n12 Hariyali
Nan.....78 \n13 Kashmiri Nan
.....98 \n14 Plain
Nan.....48 \nINDIAN RICE & BIRYANI
\n15 Handi Dum Parida Biryani .....230 \n16 Special
Veg. Biryani.....195 \n17 Veg. Hyderabadi
Biryani.....205 \n18 Veg. Pulao (Plain or
Masala).....180 \n19 Peas Pulao
.....180 \n20 Mushroom Onion
Pulao.....195 \nSNACKS \n21 Cheese Garlic
Bread.....130 \n22 Chesse Corn Chilly
Toast.....130 \n23 Veg. Cutlet
.....85 \n24 Aloo Tikki
.....100 \n25 Veg. Cheese Sandwich
.....90 \nSOUTH INDIAN \n26 Mysore Masala
Dosa.....140 \n27 Mysore Plain
Dosa.....130 \n28 Hyderabadi Masala Dosa
.....140 \n29 Hyderabadi Plain Dosa
.....130 \n*DESSERTS & SWEETS \n30 Sunday
Special Ice-cream.....170 \n31 Almond Carnival
Ice-cream .....55 \n32 Kesar Pista Ice-cream
.....55 \n33 Kaju Draksh Ice-

```

```

cream.....45 \n34 Butter Scotch Ice-
cream.....45\n");
printf("\n::::::::::::::::::::::::::::::::::::::::::::\n\n");
//getchar();
printf("ENTER THE S.NO. OF THE ITEM FROM THE ABOVE LIST :\n");
scanf("%d",&item);
printf("ENTER THE QUANTITY: \n");
scanf("%d",&quantity);
Bill();
}
void Bill(){
    //defining of the function Bill()
    item\_array[count] = item;
    quantity\_array[count] = quantity;
    count++;
    float price;
    //HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE ITEM CHOOSEN
    BY THE USER
    switch (item)
    {
    case 1:
        price = 220 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 2:
        price = 210 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 3:
        price = 215 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 4:
        price = 215 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 5:
        price = 220 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 6:
        price = 200 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 7:
        price = 200 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 8:
        price = 195 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 9:
        price = 195 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 10:
        price = 188 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 11:
        price = 88 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 12:
        price = 78 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 13:
        price = 98 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 14:
        price = 48 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 15:
        price = 230 * quantity;
        break; // break statement is used to terminate switch statement execution,

```

```

case 16:
    price = 195 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 17:
    price = 205 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 18:
    price = 180 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 19:
    price = 180 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 20:
    price = 195 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 21:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 22:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 23:
    price = 85 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 24:
    price = 100 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 25:
    price = 90 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 26:
    price = 140 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 27:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 28:
    price = 140 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 29:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 30:
    price = 170 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 31:
    price = 55 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 32:
    price = 55 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 33:
    price = 45 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 34:
    price = 45 * quantity;
    break; // break statement is used to terminate switch statement execution,
}

float gst = (price * gstRate) / 100;
float total = price + gst;
grandTotal += total;
int add;
printf("DO YOU WANT TO ADD MORE ITEMS TO YOUR CART YES = 1 / NO = 0 or press any other
key to return to main menu :");

```

```

scanf("%d", &add);
switch(add){
    case 0:{
        invoice();
        break; // break statement is used to terminate switch statement execution,
    }
    case 1: {
        menu();
        break; // break statement is used to terminate switch statement execution,
    }
}
}
//defining of the function invoice()
void invoice(){
    system("cls"); //this is the function which is defined in stdlib.h and it is used to
clear terminal screen
    printf("***** : ) YOUR INVOICE (:
*****\n\n\n");

    printf("Customer Name : %s\n\n", user1.User\_Name);
    printf("Customer Address : %s\n\n", user1.User\_address);
    printf("Contact Details : %lu\n\n",
user1.User\_contactdetails);
    printf("Email-id : %s\n\n", user1.User\_emial\_id);
    printf("Item name Quantity \n\n");
    for (int i = 0; i < count; i++)
    {
        printf(" %s -----> %d-only\n\n
", menu\_array[item\_array[i]], quantity\_array[i]);
    }
    printf("-----\n\n");
    printf("Total : %3f Rs/- Only\n\n\n",
grandTotal);

    printf("***** Thank you! Do visit again :)
*****");
    printf("\nPress any key to continue...");
    //getch(); //Here //getch will take any character input and then this function will be
terminated and the next line of the calling function will execute
    save\_details(); //calling of function ----> save\_details()
    exit(0); //exit(0) this function is used to successfully terminate and exit from the
program
}
void save\_details(){
    FILE * FILE\_Pointer; //creation of file pointer
    FILE\_Pointer = fopen("invoice.txt", "a+"); //opening the file invoice.txt in append plus
mode and assigning it to file pointer
    /* Write data to file */
    fprintf(FILE\_Pointer, "***** : ) YOUR INVOICE (:
*****\n\n\n");

    fprintf(FILE\_Pointer, "Customer Name : %s\n\n",
user1.User\_Name);
    fprintf(FILE\_Pointer, "Customer Address : %s\n\n",
user1.User\_address);
    fprintf(FILE\_Pointer, "Contact Details : %lu\n\n",
user1.User\_contactdetails);
    fprintf(FILE\_Pointer, "Email-id : %s\n\n",
user1.User\_emial\_id);
    fprintf(FILE\_Pointer, "Item name Quantity
\n\n");
    for (int i = 0; i < count; i++)
    {

```

```

        fprintf(FILE\_Pointer, "                                %s -----> %d-only\n\n",
        ",menu\_array[item\_array[i]],quantity\_array[i]);
    }
    fprintf(FILE\_Pointer, "-----\n\n");
    fprintf(FILE\_Pointer, "                                Total                :    %3f Rs/-
Only\n\n\n\n", grandTotal);
    /* Close file to save file data */
    fclose(FILE\_Pointer);
}
void Booking(){
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    printf("Here is the list of vacant and occupied tables\n\n");
    /*using the for loop to traverse through the array and check each value of element of
array and print booked if it is 1 and print vacant if it is 0 */
    for (int k = 0; k < 10; k++){
        if (Table\_array[k]==1){
            printf("Table no. %d is already Booked\n", k + 1);
        }
        else {
            printf("Table no. %d is vacant\n", k + 1);
        }
    }

    printf("Enter the table number of the table you want to book\n");
    scanf("%d", &Table\_no);
    Table\_array[Table\_no+1] == 1; /*assigning 1 to [Table\_no+1] of the array to mark it as
booked */
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    printf("
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n");
    printf("                                Customer Name      : %s\n\n", user1.User\_Name);
    printf("                                Contact  Details : %lu\n\n",
user1.User\_contactdetails);
    printf("                                Email-id          : %s\n\n", user1.User\_emial\_id);
    printf("                                Congratulations table number %d successfully
booked\n", Table\_no);
    printf("
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n");
    save\_booking\_details(); //calling of function ----> save\_booking\_details()
    exit(0); //exit(0) this function is used to successfully terminate and exit from the
program
}
void save\_booking\_details(){
    FILE * FILE\_Pointer; //creation of file pointer
    FILE\_Pointer = fopen("booking.txt", "a+"); //opening the file booking.txt in append plus
mode and assigning it to file pointer
    /* Write data to file */
    fprintf(FILE\_Pointer, "
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n");
    fprintf(FILE\_Pointer, "                                Customer Name      : %s\n\n",
user1.User\_Name);
    fprintf(FILE\_Pointer, "                                Contact  Details : %lu\n\n",
user1.User\_contactdetails);
    fprintf(FILE\_Pointer, "                                Email-id          : %s\n\n",
user1.User\_emial\_id);
    fprintf(FILE\_Pointer, "                                Congratulations table number %d
successfully booked\n", Table\_no);
    fprintf(FILE\_Pointer, "
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n");
    /* Close file to save file data */
    fclose(FILE\_Pointer);
}

```

```

void Cancel\_Booking(){
    int cancel;
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    printf("Enter table number to cancel your booking \n\n");
    scanf("%d", &cancel);
    Table\_array[cancel+1] == 0; /*assigning 0 to [cancel+1] of the array to mark it as
vacant */
    system("cls"); //this is the function which is defined in stdlib.h and it is used to clear
terminal screen
    printf("
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n\n");
    printf("                                Customer Name      : %s\n\n", user1.User\_Name);
    printf("                                Contact   Details : %lu\n\n",
user1.User\_contactdetails);
    printf("                                Email-id          : %s\n\n", user1.User\_emial\_id);
    printf("                                Your booking for table number %d successfully
canceled\n\n", cancel);
    printf("
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n");
    exit(0); //exit(0) this function is used to successfully terminate and exit from the
program
}

void print\_invoice(){
    FILE *file\_pointer; //creation of file pointer
    file\_pointer = fopen("invoice.txt", "r+"); //opening the file invoice.txt in read plus
mode and assigning it to file pointer
    char c; //declaration of char c to read data from the file
    c = fgetc(file\_pointer); //fgetc is used to fetch data from the file
/*while loop is used it will print data until the eof occurs*/
    while (c != EOF)
    {
        printf("%c", c);
        c = fgetc(file\_pointer);
    }
    /* Close file to save file data */
    fclose(file\_pointer);
    exit(0); //exit(0) this function is used to successfully terminate and exit from the
program
}

void view\_bookings(){
    FILE *file1\_pointer; //creation of file pointer
    file1\_pointer = fopen("booking.txt", "r+"); //opening the file booking.txt in read plus
mode and assigning it to file pointer
    char c; //declaration of char c to read data from the file
    c = fgetc(file1\_pointer); //fgetc is used to fetch data from the file
/*while loop is used it will print data until the eof occurs*/
    while (c != EOF)
    {
        printf("%c", c);
        c = fgetc(file1\_pointer);
    }
    fclose(file1\_pointer); /* Close file to save file data */
    exit(0); //exit(0) this function is used to successfully terminate and exit from the
program
}

\end{lstlisting}
\newpage
\section{\Huge{Code with Comment in C++ Language}}
\begin{lstlisting}

/*adding all the necessary header files*/
#include <iostream>

```



```

#include <cstring>

/*Declaration of class*/
class User
{
private:
    char User Name[20];
    long long User contactdetails;
    char User address[50];
    char User emial id[30];

public:
    /*Declaration of the other variables */
    int Table no;
    int gstRate;
    int item;
    int quantity;
    float grandTotal;
    int restaurant name;
    int option;
    int count = 0;

    /*Declaration of all the necessary functions */
    void save_booking_details();
    void Booking();
    void Cancel_Booking();
    void menu();
    void BHOPAL info(int restaurant name);
    void MUMBAI info(int restaurant name);
    void INDORE info(int restaurant name);
    void City choice(int);
    void city display();
    void place_order();
    int Admin authentication();
    void Admin Page();
    void User Registration Page();
    void INDORE();
    void BHOPAL();
    void MUMBAI();
    void options();
    void save_details();
    void invoice();
    void Bill();
    void print_invoice();
    void Welcome_Page();
    void view_bookings();
} user1; /*Declaration of variable of class type*/

/*Declaration of array to store data*/
char food name[50];
int item array[31] = {0};
int quantity array[35] = {0};
int Table array[10] = {0};
char menu array[][35] = {"Nothing Selected", "Paneer Angara", "Paneer Pasaenda", "Paneer Lababdar", "Paneer Tawa Masala", "Paneer Toofani", "Paneer Handi", "Paneer Kadai", "Paneer Tikka Masala", "Paneer Butter Masala plus", "Paneer Palak", "Cheese Nan", "Hariyali Nan", "Kashmiri Nan", "Plain Nan", "Handi Dum Parda Biryani", "Special Veg. Biryani", "Veg. Hyderabad Biryani", "Veg. Pulao (Plain or Masala)", "Peas Pulao", "Mushroom Onion Pulao", "Cheese Garlic Bread", "Chesse Corn Chilly Toast", "Veg. Cutlet", "Aloo Tikki", "Veg. Cheese Sandwich", "Mysore Masala Dosa", "Mysore Plain Dosa", "Hyderabad Masala Dosa", "Hyderabad Plain Dosa", "Sunday Special Ice-cream", "Almond Carnival Ice-cream", "Kesar Pista Ice-cream", "Kaju Draksh Ice-cream", "Butter Scotch Ice-cream"};

int main()
{

```

```

    std ::cout << "Welcome to restaurant booking and food odering system developed by Aman
Bagadiya";
    user1.gstRate = 18;    // default gst rates
    user1.Welcome Page(); // calling of function ----> Welcome Page();
}

// defining of the function Welcome Page
void User ::Welcome Page()
{
    system("cls");    // this is the function which is defined in stlib.h and it is used to
clear terminal screen
    int Login option; // defining of the variable
    std :: cout<<"\t\t\t\tWelcome to Restaurant Booking and Food odering system developed
by Aman Bagadiya under the guidance of Prof.Surendra Gupta Sir.\n";
    std :: cout<<"\t\t\t\t\t-----\n\n";
    std :: cout<<"\t\t\t\t\t
::::::::::::::::::::::::::::::::::::::::::::::::::\n";
    std :: cout<<"\t\t\t\t\t:: THIS MINI PROJECT AIMS TO
PERFORM VARIOUS FUNCTIONS ::\n";
    std :: cout<<"\t\t\t\t\t
::::::::::::::::::::::::::::::::::::::::::::::::::\n\n";
    std :: cout<<"\t\t\t\t\tA.USER FUNCTIONS :\n";
    std :: cout<<"\t\t\t\t\t-> ORDER FOOD :\n";
    std :: cout<<"\t\t\t\t\t-> BOOK TABLE AT
RESTAURANT :\n";
    std :: cout<<"\t\t\t\t\t-> CANCELL BOOKING
:\n";
    std :: cout<<"\t\t\t\t\t-> GET INFORMATION
ABOUT RESTAURANT :\n\n";
    std :: cout<<"\t\t\t\t\tB.ADMIN FUNCTIONS
:\n";
    std :: cout<<"\t\t\t\t\t-> ADMIN
AUTHENTICATION :\n";
    std :: cout<<"\t\t\t\t\t-> SET/UPDATE GST
RATES :\n";
    std :: cout<<"\t\t\t\t\t-> GENERATE INVOICE
:\n";
    std :: cout<<"\t\t\t\t\t-> VIEW ALL
INVOICES :\n";
    std :: cout<<"\t\t\t\t\t-> VIEW ALL
BOOKINGS :\n";
    std ::cout << "Login: \n";
    std ::cout << "For User login press 1: \n";
    std ::cout << "For Admin login press 2: \n";
    std ::cout << "To exit press 0: \n";
    std ::cin >> Login option;
    // HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE USER CHOICE
switch (Login option)
{
case 1:
    User Registration Page(); // calling of function ----> User Registration Page()
    break;    // break statement is used to terminate switch statement
execution,

case 2:
    Admin Page(); // calling of function ----> Admin Page();
    break;    // break statement is used to terminate switch statement execution,

case 0:
    std ::cout << "You have successfully exited from program \n\n ";
    std ::cout << "This project is made by Aman Bagadiya under the guidence of Mr
Surendra Gupta Sir \n\n";
    break; // break statement is used to terminate switch statement execution,

```

```

        default:
            std ::cout << "You have entered an invalid choice...Exiting...\n \n";
            exit(0); // exit(0) this function is used to successfully terminate and exit from the
program
    }
}

// defining of the function Admin Page()
void User ::Admin Page()
{
    if (Admin authentication() == 1)
    {
        // calling of the Admin authentication() function to verify the admin
        system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
        std ::cout << "Current GST rate " << gstRate << " percent : \n";
        int select; // defining of the variable
        std ::cout << "Press 1 to update GST rate:\n";
        std ::cout << "Press 2 to view all Invoices : \n";
        std ::cout << "Press 3 to view all Bookings : \n";
        std ::cout << "To exit press 0: \n";
        std ::cin >> select;
        // HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE ADMIN
CHOICE

        switch (select)
        {
            case 1:
                std ::cout << "Enter New GST rates (in percentage): \n";
                std ::cin >> gstRate;
                Welcome Page();

            case 2:
                print invoice(); // calling of function ----> print invoice()

            case 3:
                view bookings(); // calling of function ----> view bookings()

        }
    }
}

// defining of the function Admin authentication()
int User ::Admin authentication()
{
    system("cls"); // This is the function which is defined in stlib.h and it is used
to clear terminal screen
    int admin check = 1; // defining of the variable
    char loginID[] = "Aman";
    char password[] = "Aman";
    char user entered password[11]; // Declaration of the array user entered password to
store login id
    char user entered login[10]; // Declaration of the array user entered login to store
password

    std ::cout << "*****\t Welcome to Admin Login Page \t*****\n\n";
    std ::cout << "Kindly enter your credentials below : \n\n";
    std ::cout << "Enter Admin login ID: ";
    std ::cin >> user entered login;
    std ::cout << "Enter Admin password: ";
    std ::cin >> user entered password;
    if (strcmp(loginID, user entered login) != 0)
    { // Using strcmp function to compare 2 strings loginID and user entered login
        admin check = 0;
    }
    if (strcmp(password, user entered password) != 0)
    { // Using strcmp function to compare 2 strings password and user entered password
        admin check = 0;
    }
}

```

```

if (admin_check == 0)
{
    std ::cout << "Sorry You Have Entered Wrong Credentials.\n";
    std ::cout << "Try Again\n";
    return 0;
}
else
{
    return 1; // if admin is successfully verified it will return 1
}
}

void User ::User Registration Page()
{
    // defining of the function User Registration Page()
    system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
    std ::cout << "Enter Your Name\n";
    fflush(stdin); // it is used to clear the buffer
    gets(user1.User Name);
    std ::cout << "Enter Your Phone Number\n";
    std ::cin >> user1.User contactdetails;
    std ::cout << "Enter Your Email-id\n";
    fflush(stdin); // it is used to clear the buffer
    gets(user1.User emial id);
    std ::cout << "Enter Address\n";
    fflush(stdin); // it is used to clear the buffer
    gets(user1.User address);
    std ::cout << "Your details registered successfully\n";
    options(); // calling of function ----> options
}

void User ::options()
{
    do
    {
        system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
        std ::cout << "\n
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::";
        std ::cout << "\n                                     ::             !!!!!!!!!!!!!!!welcome!!!!!!!!!!!!!!
::";
        std ::cout << "\n                                     ::
::";
        std ::cout << "\n                                     ::      ~> Please select from below options:
::";
        std ::cout << "\n                                     ::
::";
        std ::cout << "\n                                     ::          1.Get Information About Restaurant
:>          ::";
        std ::cout << "\n                                     ::          2.Order Food:>
::";
        std ::cout << "\n                                     ::          3.Book table:>
::";
        std ::cout << "\n                                     ::          4.Cancel your Booking:>
::";
        std ::cout << "\n                                     ::          5.Exit:>
::";
        std ::cout << "\n
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n\n";
        std ::cout << "PLEASE SELECT FROM THE ABOVE OPTIONS:\n";
        std ::cin >> option;
        // HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE ADMIN
CHOICE
        switch (option)

```

```

{
case 1:
    city display();
    break; // break statement is used to terminate switch statement execution,
case 2:
    place order();
    break; // break statement is used to terminate switch statement execution,
case 3:
{
    city display();
    Booking();
}
case 4:
{
    city display();
    Cancel Booking();
}
case 5:
{
    std ::cout << "\nExiting...\n";
    exit(0); // exit(0) this function is used to successfully terminate and exit from
the program
}
default:
{
    std ::cout << "Please Enter A Valid Choice";
}
}
} while (1);
}

void User ::place order()
{
    city display();
    menu();
}

void User ::city display()
{
    system("cls"); // this is the function which is defined in stdlib.h and it is used to
clear terminal screen
    std ::cout << "
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n";
    std ::cout << "                                ::
::\n";
    std ::cout << "                                :: ~ Please select your city from below list:
::\n";
    std ::cout << "                                ::
::\n";
    std ::cout << "                                :: > 1.INDORE
::\n";
    std ::cout << "                                :: >> 2.BHOPAL
::\n";
    std ::cout << "                                :: >>> 3.MUMBAI
::\n";
    std ::cout << "                                ::
::\n";
    std ::cout << "                                ::
::\n";
    std ::cout << "                                ::
::\n";
    std ::cout << "
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n\n";
    getchar();
    std ::cout << "Enter your choice by entering the serial number of city : ";

```

```

    int city choice;
    std ::cin >> city choice;
    City choice(city choice); // calling of function ----> City choice(city choice)
}

void User ::City choice(int city choice)
{
    switch (city choice)
    {
        // HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE ADMIN
        CHOICE
        case 1:
        {
            INDORE();
            break; // break statement is used to terminate switch statement execution,
        }
        case 2:
        {
            BHOPAL();
            break; // break statement is used to terminate switch statement execution,
        }
        case 3:
        {
            MUMBAI();
            break; // break statement is used to terminate switch statement execution,
        }
    }
}

void User ::INDORE()
{
    system("cls"); // this is the function which is defined in stlib.h and it is used to
    clear terminal screen
    std ::cout << "
    ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n";
    std ::cout << "                ::      ~> Please select restaurant below list:
    ::\n";
    std ::cout << "                ::
    ::\n";
    std ::cout << "                ::      Name of Restaurant                Ratings
    ::\n";
    std ::cout << "                ::
    ::\n";
    std ::cout << "                ::      1.Kebabsville                **
    ::\n";
    std ::cout << "                ::      2.Ginger Ganesha                ***
    ::\n";
    std ::cout << "                ::      3.Shree Chotiwal                *
    ::\n";
    std ::cout << "                ::      4.Village                **
    ::\n";
    std ::cout << "                ::      5.Vidorra                *
    ::\n";
    std ::cout << "                ::
    ::\n";
    std ::cout << "
    ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\n\n";
    getchar();
    std ::cout << "Enter your choice by entering the serial number of restaurant : \n";
    int restaurant name;
    std ::cin >> restaurant name;
    if (option == 1)
        INDORE info(restaurant name); // calling of function ----> INDORE info
    std ::cout << "\n\n";
}

```

```

void User ::BHOPAL()
{
    system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
    std ::cout << "
::::::::::::::::::::::::::::::::::::::::::::::::::::\n";
    std ::cout << "                ::      ~> Please select restaurant below list:
::\n";
    std ::cout << "                ::
::\n";
    std ::cout << "                ::      Name of Restaurant                Ratings
::\n";
    std ::cout << "                ::
::\n";
    std ::cout << "                ::      1.Manohar                **
::\n";
    std ::cout << "                ::      2.Cafe Chokolade            **
::\n";
    std ::cout << "                ::      3.Greek Food & Beyond        *
::\n";
    std ::cout << "                ::      4.Bapu Ki Kutia                **
::\n";
    std ::cout << "                ::      5.Al-Beik                ***
::\n";
    std ::cout << "                ::
::\n";
    std ::cout << "
::::::::::::::::::::::::::::::::::::::::::::::::::::\n\n";
    getchar();
    std ::cout << "Enter your choice by entering the serial number of restaurant : \n";
    int restaurant name;
    std ::cin >> restaurant name;
    if (option == 1)
        BHOPAL info(restaurant name); // calling of function ----> BHOPAL info
    std ::cout << "\n\n";
}

```

```

void User ::MUMBAI()
{
    system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
    std ::cout << "
::::::::::::::::::::::::::::::::::::::::::::::::::::\n";
    std ::cout << "                ::      ~> Please select restaurant below list:
::\n";
    std ::cout << "                ::
::\n";
    std ::cout << "                ::      Name of Restaurant                Ratings
::\n";
    std ::cout << "                ::
::\n";
    std ::cout << "                ::      1.Bayroute Restaurant            **
::\n";
    std ::cout << "                ::      2. Hakkasan                ***
::\n";
    std ::cout << "                ::      3.Dome Intercontinental            *
::\n";
    std ::cout << "                ::      4.Yauatcha Restaurant            **
::\n";
    std ::cout << "                ::      5.Pali Village cafe                *
::\n";
    std ::cout << "                ::
::\n";

```

```

    std ::cout << "
    ::::::::::::::::::::::::::::::::::::::\n\n";
    getchar();
    std ::cout << "Enter your choice by entering the serial number of restaurant : \n";

    std ::cin >> restaurant_name;
    if (option == 1)
        MUMBAI_info(restaurant_name); // calling of function ----> MUMBAI_info
    std ::cout << "\n\n";
}

void User ::INDORE_info(int restaurant_name)
{
    system("cls"); // this is the function which is defined in stlib.h and it is used to
    clear terminal screen
    switch (restaurant_name)
    {
    case 1:
    {
        std ::cout << "LOCATION : Sayaji Indore, H/1 , Scheme No.54, Vijay Nagar, Indore,
        Madhya Pradesh - 452010\n\nPHONE :0731 4006666, 73899 48912\n\nMORE INFORMATION: PRICE RANGE
        =₹805 - ₹1,208\n\nCUISINES =Indian, Barbecue, Asian\n\nSPECIAL DIETS =Vegetarian Friendly, Vegan
        Options, Halal, Gluten Free Options Dinners can now become even more special with a laid back
        and relaxed time relishing the buffet at a Kebabsville! Sayaji Indore's buffet style
        restaurant- Kebabsville is a class apart! Set alongside the pool with lavish seating
        arrangement, you can have a relaxing buffet full of mouth-watering kebabs, grilled veggies
        and North Indian food dishes. Variety of kebabs in vegetarian and non-vegetarian options,
        made by our expert chefs are sure to win your hearts. You can eat the kebabs to your heart's
        content and also enjoy the North India dishes alongside. Relish in our lavishly spread buffet
        service and make your evenings worth remembering.\n\n";
        exit(0); // break statement is used to terminate switch statement execution,
    }
    case 2:
    {
        std ::cout << "LOCATION LG-12,13, DM Tower, 21/1, Lala Banarasilal Dawar Marg, Race
        Course Road, Near Zanjeerwala Chowraha, New Palasia, Indore, Madhya Pradesh 452001,
        India\n\nPhone=+91-731-3301441\n\nMore information-A good and lively ambience with amazing
        South Indian and Chinese dishes, Ginger Ganesha might just steal your heart with the sheer
        simplicity of its. If you want to have excellent dosas and super delicious vegetarian biryani
        then this is the place to be. Do try the Bisebele Bath and the Mexican Roll Dosa. Not only
        the food, but the restaurant is also known for its quick service which is an added star. This
        is also a popular place to dine at with the family because of the reasonable price at which
        you can eat with your heart and the cozy environment that gives you the comfort of being in
        the company of your loved ones\n\n";
        exit(0); // break statement is used to terminate switch statement execution,
    }
    case 3:
    {
        std ::cout << "LOCATION: 8 B, Raunak Plaza, Opposite Nath Mandir, South Tukoganj,
        Indore, Madhya Pradesh 452001, India\n\nPHONE: +91-731-2528586\n\nMore Information: One of
        the most popular eating joints in the city, Shree Chotiwalla serves you with delicious veg
        Indian food. The wide variety of dishes in the menu might confuse you a bit. That's a good
        thing, right? There is a booth seating and a family table seating arrangement that are
        available for different purposes and guests. Not to miss is the veg thali that has a huge
        range of dishes that are served on a single platter. It's quite a treat in every sense of the
        word. There are also dishes for the stricity display followers of Jainism available at this
        restaurant which essentially means vegetarian dishes that do not even have onion or garlic in
        it\n\n";
        exit(0); // break statement is used to terminate switch statement execution,
    }
    case 4:
    {
        std ::cout << "LOCATION: 5th Floor, Central Mall, RNT Marg, Indore, Madhya Pradesh
        452001, India\n\nPHONE: +91-731-4006617|+91-78699-10313\n\nMore Information: Sit on a chair,
        on a khatiya or with a cozy takiya, Village is a sure shot dream to dine at. Serving

```


Rajasthani and Gujarati cuisines as their specialty, the place ensures a fulfilling experience for every foodie who enters the place. The thatched stalls serving street food from various parts of the country add another world charm to the place. The truck with a buffet arrangement and the puppet show are amazing. If you are the kind of person who does not get satisfied with just the food then the overall experience here is going to stay with you for a long time.\n";

```
        exit(0); // break statement is used to terminate switch statement execution,
    }
    case 5:
    {
        std ::cout << "LOCATION : New Palasia, Indore\n\nCONTACT DETAILS:12345678 \n\nMORE
INFORMATION :CUISINES Indian, Asian SPECIAL DIETS Vegan Options, Vegetarian Friendly MEALS
Dinner, Lunch.\n";
        exit(0); // break statement is used to terminate switch statement execution,
    }
    }
    std ::cout << "\nPress any key to continue...";
}

void User ::BHOPAL info(int restaurant_name)
{
    system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
        // HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON
THE USER CHOICE
    switch (restaurant_name)
    {
    case 1:
    {
        std ::cout << "132, Zone 1, Maharana Pratap Nagar, Bhopal\nStreet Food, South Indian,
Fast Food, Desserts, North Indian, Chinese, Bakery, Mithai \n8:30am to 10:30pm (Mon-Sun)
\n300\n";
    }
    case 2:
    {
        std ::cout << "\n206, Zone 2, Near Arya Bhavan, Maharana Pratap Nagar, Bhopal,\n
India Cafe, Bakery \n11 AM to 11 PM \nINR 450\n \n";
        break; // break statement is used to terminate switch statement execution,
    }
    case 3:
    {
        std ::cout << "Third Floor, DB City Mall, Maharana Pratap Nagar, Bhopal, \nIndia
Greek, Mediterranean \n12 Noon to 10:30 PM \nINR 1000\n";
        break; // break statement is used to terminate switch statement execution,
    }

    case 4:
    {
        std ::cout << " 123, Jyoti Shopping Centre, Behind Jyoti Talkies Zone-I, Maharana
Pratap Nagar,Bhopal, Madhya Pradesh 462011\n North-Indian, Chinese, South-Indian, Snacks \n7
AM to 11 PM \nINR 500 \n";
        break; // break statement is used to terminate switch statement execution,
    }

    case 5:
    {
        std ::cout << "Regiment Rd,Badabagh, Shahjahanabad,Bhopal, Madhya Pradesh 462001\n
Snacks \n1:30 PM to 10 PM\n INR 300\n";
        break; // break statement is used to terminate switch statement execution,
    }
    }
    std ::cout << "\nPress any key to continue...";
}
```

```

void User ::MUMBAI_info(int restaurant_name)
{
    system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
    // HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON
THE USER CHOICE
    switch (restaurant_name)
    {
        case 1:
        {
            std ::cout << "Location: Bayroute Juhu, 14, Silver Beach Estate Opposite Juhu Post
Office, AB Nair Road, Juhu, Mumbai.\n Foods to try: Harissa Chicken Kebab, Kebab Istanbul,
Baklava, Gold Souk are some of the must-try dishes.\n Price for two: INR 3000 \n";
            break; // break statement is used to terminate switch statement execution,
        }
        case 2:
        {
            std ::cout << "Location: Hakkasan, 2nd Floor, Krystal Building, Waterfield Road,
Bandra West, Mumbai. \nFoods to try: Everything in Cantonese cuisine menu is so yum and do
not forget to try the dumplings. \nPrice for two: INR 2500\n";
            break; // break statement is used to terminate switch statement execution,
        }
        case 3:
        {
            std ::cout << "Location: Dome Intercontinental 135, Marine Drive, Mumbai, \nFoods to
try: Afghani Chicken, Brownie and cocktails.\n Price for two: INR 3,500\n";
            break; // break statement is used to terminate switch statement execution,
        }
        case 4:
        {
            std ::cout << "Location: Yauatcha, Raheja Towers, Bandra Kurla Complex, Bandra East,
Mumbai\n Foods to try: Crispy Prawns Cheung Fun, Pork char sui bun are extraordinary. \nPrice
for two: INR 3,500\n";
            break; // break statement is used to terminate switch statement execution,
        }
        case 5:
        {
            std ::cout << "Location: Pali Village Cafe, 602, Ambedkar Road, Pali Naka, Pali Hill,
Bandra West, Mumbai\n Foods to try: Risotto, Salads, Panna Cotta, Pizzas are worth a try.
\nPrice for two: INR 3000\n";
            break; // break statement is used to terminate switch statement execution,
        }
        }
    std ::cout << "\nPress any key to continue...";
}

void User ::menu()
{
    // defining of the function menu()
    std ::cout << "\n ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::";
    std ::cout << "\n\n
ITEMS.....PRICE \n1 Paneer
Angara.....220 \n2 Paneer Pasasnda
.....210 \n3 Paneer
Lababdar.....215 \n4 Paneer Tawa
Masala.....215 \n5 Paneer Toofani
.....220 \n6 Paneer
Handi.....200 \n7 Paneer
Kadai..... 200 \n8 Paneer Tikka
Masala.....195 \n9 Paneer Butter
Masala.....195 \n10 Paneer
Palak.....188 \n*ROTIES - TANDURI \n11
Cheese Nan .....88 \n12 Hariyali
Nan.....78 \n13 Kashmiri Nan
.....98 \n14 Plain

```

```

Nan.....48 \nINDIAN RICE & BIRYANI
\n15 Handi Dum Parda Biryani .....230 \n16 Special
Veg. Biryani.....195 \n17 Veg. Hyderabadi
Biryani.....205 \n18 Veg. Pulao (Plain or
Masala).....180 \n19 Peas Pulao
.....180 \n20 Mushroom Onion
Pulao.....195 \nSNACKS \n21 Cheese Garlic
Bread.....130 \n22 Chesse Corn Chilly
Toast.....130 \n23 Veg. Cutlet
.....85 \n24 Aloo Tikki
.....100 \n25 Veg. Cheese Sandwich
.....90 \nSOUTH INDIAN \n26 Mysore Masala
Dosa.....140 \n27 Mysore Plain
Dosa.....130 \n28 Hyderabadi Masala Dosa
.....140 \n29 Hyderabadi Plain Dosa
.....130 \n*DESSERTS & SWEETS \n30 Sunday
Special Ice-cream.....170 \n31 Almond Carnival
Ice-cream .....55 \n32 Kesar Pista Ice-cream
.....55 \n33 Kaju Draksh Ice-
cream.....45 \n34 Butter Scotch Ice-
cream.....45\n";

```

```

    std ::cout << "\n::::::::::::::::::::::::::::::::::::\n\n";
    getchar();
    std ::cout << "ENTER THE S.NO. OF THE ITEM FROM THE ABOVE LIST :\n";
    std ::cin >> item;
    std ::cout << "ENTER THE QUANTITY: \n";
    std ::cin >> quantity;
    Bill();
}

void User ::Bill()
{
    // defining of the function Bill()
    item_array[count] = item;
    quantity_array[count] = quantity;
    count++;
    int price;
    // HERE SWITCH CASE IS USED TO CALL THE CORRESPONDING OPTION DEPENDING ON THE ITEM
    CHOOSEN BY THE USER
    switch (item)
    {
    case 1:
        price = 220 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 2:
        price = 210 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 3:
        price = 215 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 4:
        price = 215 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 5:
        price = 220 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 6:
        price = 200 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 7:
        price = 200 * quantity;
        break; // break statement is used to terminate switch statement execution,
    case 8:
        price = 195 * quantity;
        break; // break statement is used to terminate switch statement execution,

```

```
case 9:
    price = 195 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 10:
    price = 188 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 11:
    price = 88 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 12:
    price = 78 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 13:
    price = 98 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 14:
    price = 48 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 15:
    price = 230 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 16:
    price = 195 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 17:
    price = 205 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 18:
    price = 180 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 19:
    price = 180 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 20:
    price = 195 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 21:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 22:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 23:
    price = 85 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 24:
    price = 100 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 25:
    price = 90 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 26:
    price = 140 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 27:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 28:
    price = 140 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 29:
    price = 130 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 30:
    price = 170 * quantity;
```

```

        break; // break statement is used to terminate switch statement execution,
case 31:
    price = 55 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 32:
    price = 55 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 33:
    price = 45 * quantity;
    break; // break statement is used to terminate switch statement execution,
case 34:
    price = 45 * quantity;
    break; // break statement is used to terminate switch statement execution,
}

float gst = (price * gstRate) / 100;
float total = price + gst;
grandTotal += total;
int add;
std ::cout << "DO YOU WANT TO ADD MORE ITEMS TO YOUR CART YES = 1 / NO = 0 or press any
other key to return to main menu :";
std ::cin >> add;
switch (add)
{
case 0:
{
    invoice();
    break; // break statement is used to terminate switch statement execution,
}
case 1:
{
    menu();
    break; // break statement is used to terminate switch statement execution,
}
}
}

// defining of the function invoice()
void User ::invoice()
{
    system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
    std ::cout << "***** : ) YOUR INVOICE (:
*****\n\n\n";

    std ::cout << "                                Customer Name      : " << user1.User_Name << "
\n\n";
    std ::cout << "                                Customer Address : " << user1.User_address << "
\n\n";
    std ::cout << "                                Contact  Details : " << user1.User_contactdetails
<< " \n\n";
    std ::cout << "                                Email-id          : " << user1.User_email_id << "
\n\n";
    std ::cout << "                                Item name          Quantity          \n\n";
    for (int i = 0; i < count; i++)
    {
        std ::cout << "                                "<< menu_array[item_array[i]] <<" ----->
"<< quantity_array[i] <<"-only\n\n ";
    }
    std ::cout << "-----\n\n";

    std ::cout << "                                Total              : " << grandTotal << " Rs/-
Only\n\n\n\n";

```

```

    std ::cout << "***** Thank you! Do visit again :)
1*****";
    std ::cout << "\nPress any key to continue...";
    getch(); //Here getch will take any character input and then this function will be
terminated and the next line of the calling function will execute

    save_details(); // calling of function ----> save_details()
    exit(0);        // exit(0) this function is used to successfully terminate and exit from
the program
}

void User ::save_details()
{
    FILE *FILE Pointer;                // creation of file pointer
    FILE Pointer = fopen("invoice.txt", "a"); // opening the file invoice.txt in append plus
mode and assigning it to file pointer
    /* Write data to file */
    fprintf(FILE Pointer, "***** :) YOUR INVOICE (:
*****\n\n\n");

    fprintf(FILE Pointer, "                Customer Name      :  %s\n\n",
user1.User Name);
    fprintf(FILE Pointer, "                Customer Address :  %s\n\n",
user1.User address);
    fprintf(FILE Pointer, "                Contact  Details :  %lu\n\n",
user1.User contactdetails);
    fprintf(FILE Pointer, "                Email-id          :  %s\n\n",
user1.User emial id);
    fprintf(FILE Pointer, "                Item name              Quantity
\n\n");
    for (int i = 0; i < count; i++)
    {
        fprintf(FILE Pointer, "                %s -----> %d-only\n\n ",
menu array[item array[i]], quantity array[i]);
    }
    fprintf(FILE Pointer, "-----
-----\n\n");
    fprintf(FILE Pointer, "                Total                :  %3f Rs/-
Only\n\n\n", grandTotal);
    /* Close file to save file data */
    fclose(FILE Pointer);
}

void User ::Booking()
{
    system("cls"); // this is the function which is defined in stdlib.h and it is used to
clear terminal screen
    std ::cout << "Here is the list of vacant and occupied tables\n\n";
    /*using the for loop to traverse through the array and check each value of element of
array and print booked if it is 1 and print vacant if it is 0 */
    for (int k = 0; k < 10; k++)
    {
        if (Table array[k] == 1)
        {
            std ::cout << "Table no. " << k + 1 << " is already Booked\n";
        }
        else
        {
            std ::cout << "Table no. " << k + 1 << " is vacant\n";
        }
    }
}

std ::cout << "Enter the table number of the table you want to book\n";
std ::cin >> Table no;

```

```

    Table_array[Table_no + 1] == 1; /*assigning 1 to [Table_no+1] of the array to mark it as
booked */
    system("cls"); // this is the function which is defined in stlib.h and
it is used to clear terminal screen
    std::cout << "
::::::::::::::::::::::::::::::::::::::::::::\n";
    std::cout << "                                Customer Name      :  " << user1.User_Name <<
"\n\n";
    std::cout << "                                Contact  Details :  " << user1.User_contactdetails
<< "\n\n";
    std::cout << "                                Email-id           :  " << user1.User_email_id <<
"\n\n";
    std::cout << "                                Congratulations table number " << Table_no << "
successfully booked\n";
    std::cout << "
::::::::::::::::::::::::::::::::::::::::::::\n";
    save_booking_details(); // calling of function ----> save_booking_details()
    exit(0); // exit(0) this function is used to successfully terminate and
exit from the program
}
void User ::save_booking_details()
{
    FILE *FILE Pointer; // creation of file pointer
    FILE Pointer = fopen("booking.txt", "a"); // opening the file booking.txt in append plus
mode and assigning it to file pointer
    /* Write data to file */
    fprintf(FILE Pointer, "
::::::::::::::::::::::::::::::::::::::::::::\n");
    fprintf(FILE Pointer, "                                Customer Name      :   %s\n\n",
user1.User_Name);
    fprintf(FILE Pointer, "                                Contact  Details :   %lu\n\n",
user1.User_contactdetails);
    fprintf(FILE Pointer, "                                Email-id           :   %s\n\n",
user1.User_email_id);
    fprintf(FILE Pointer, "                                Congratulations table number %d
successfully booked\n", Table_no);
    fprintf(FILE Pointer, "
::::::::::::::::::::::::::::::::::::::::::::\n");
    /* Close file to save file data */
    fclose(FILE Pointer);
}

void User ::Cancel_Booking()
{
    int cancel;
    system("cls"); // this is the function which is defined in stlib.h and it is used to
clear terminal screen
    std::cout << "Enter table number to cancel your booking \n\n";
    std::cin >> cancel;
    Table_array[cancel + 1] == 0; /*assigning 0 to [cancel+1] of the array to mark it as
vacant */
    system("cls"); // this is the function which is defined in stlib.h and it
is used to clear terminal screen
    std::cout << "
::::::::::::::::::::::::::::::::::::::::::::\n\n";
    std::cout << "                                Customer Name      :  " << user1.User_Name <<
"\n\n";
    std::cout << "                                Contact  Details :  " << user1.User_contactdetails
<< "\n\n";
    std::cout << "                                Email-id           :  " << user1.User_email_id <<
"\n\n";
    std::cout << "                                Your booking for table number " << cancel << "
successfully canceled\n\n";

```

```

    std ::cout << "
    .....\\n";
    exit(0); // exit(0) this function is used to successfully terminate and exit from the
program
}
void User ::print invoice()
{
    FILE *file_pointer; // creation of file pointer
    file_pointer = fopen("invoice.txt", "r"); // opening the file invoice.txt in read plus
mode and assigning it to file pointer
    char c; // declaration of char c to read data from the
file
    c = fgetc(file_pointer); // fgetc is used to fetch data from the file
/*while loop is used it will print data until the eof occurs*/
while (c != EOF)
{
    std ::cout << c;
    c = fgetc(file_pointer);
}
/* Close file to save file data */
fclose(file_pointer);
exit(0); // exit(0) this function is used to successfully terminate and exit from the
program
}
void User ::view bookings()
{
    FILE *file1_pointer; // creation of file pointer
    file1_pointer = fopen("booking.txt", "r"); // opening the file booking.txt in read plus
mode and assigning it to file pointer
    char c; // declaration of char c to read data from
the file
    c = fgetc(file1_pointer); // fgetc is used to fetch data from the file
/*while loop is used it will print data until the eof occurs*/
while (c != EOF)
{
    std ::cout << c;
    c = fgetc(file1_pointer);
}
fclose(file1_pointer); /* Close file to save file data */
exit(0); // exit(0) this function is used to successfully terminate and
exit from the program
}

```

\end{lstlisting}

\newpage

\section{Code Output}

```

\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 102929.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103044.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103100.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103123.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103147.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103202.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103428.png}
\end{document}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103458.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103616.png}
\includegraphics[scale=1]{../../../../Pictures/Screenshots/Screenshot 20230130 103637.png}

```