ONLINE MOVIE TICKET BOOKING SYSTEM

A MINI PROJECT REPORT

Submitted by

SANJAY.M(220701246)

B.RAGUNANDAN(220701211)

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE

RAJALAKSHMI ENGINEERING COLLEGE (AUTONOMOUS)

THANDALAM

CHENNAI-602105

2024

ABSTRACT

The online movie ticket booking system is a web-based application designed to facilitate the process of reserving tickets for movies in theaters. This system aims to enhance user convenience by allowing moviegoers to browse movie listings, select preferred showtimes, and purchase tickets from the comfort of their homes or while on the go. The system provides a user-friendly interface with features such as real-time seat selection, secure payment gateways, and instant confirmation of bookings. Additionally, it offers theaters an efficient way to manage ticket sales, track bookings, and reduce administrative overhead. By integrating features like promotional offers, user reviews, and movie trailers, the system also aims to enrich the user experience and drive engagement. This abstract outlines the key functionalities and benefits of the online movie ticket booking system, emphasizing its role in modernizing the ticket purchasing process and improving overall customer satisfaction.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 INTRODUCTION

1.2 OBJECTIVES

1.3 MODULES

2. TECHNOLOGIES USED:

2.1 USED LANGAUGES

2.2 USED TOOLS

3. REQUIREMENT AND ANALYSIS

3.1 ARCHITECTURE DIAGRAM

3.2 ER DIAGRAM

4. PROGRAM CODE

6. RESULT

7. CONCLUSION

1.INTRODUCTION

1.1 INTRODUCTION

The online movie ticket booking system is designed to make it easy for users to book movie tickets from their computers or mobile devices. It allows users to browse movie listings, view showtimes, select seats, and make secure payments. The system sends email and SMS notifications for booking confirmations and reminders. It also provides theater administrators with tools to manage movie schedules and seat availability. This project aims to enhance the convenience and efficiency of booking movie tickets.

1.2 OBJECTIVES

The objective of the online movie ticket booking system project is to develop a user-friendly platform that enables users to effortlessly browse movie listings, select seats, and purchase tickets online. The system aims to provide a seamless and secure booking experience, enhance user convenience by eliminating the need for physical queues, and offer robust management tools for theater administrators to efficiently handle movie schedules and seat availability. By achieving these goals, the project seeks to improve the overall movie-going experience for users and streamline operations for theaters.

1.3 MODULES

1. User Management

- Registration Module: Allows new users to create accounts using email, phone number, or social media.
- Login/Authentication Module: Manages user login, authentication, and password recovery.
- Profile Management Module: Enables users to view and update their personal information.

2. Movie Browsing

- Movie Listings Module: Displays currently showing and upcoming movies with detailed information (synopsis, cast, duration).
- Search and Filter Module: Allows users to search for movies by title, genre, date, and other criteria.

3. Seat Selection

- Seating Layout Module: Shows the seating chart for each theater.
- Seat Availability Module: Provides real-time updates on seat availability.
- Seat Reservation Module: Allows users to select and reserve seats.

4. Booking and Payment

- Ticket Booking Module: Manages the process of booking tickets and generates booking IDs.

- Payment Gateway Integration Module: Integrates with third-party payment systems (e.g., PayPal, Stripe) to handle transactions.
- Confirmation and Notification Module: Sends booking confirmations and reminders via email and SMS.

5. Theater Management (Admin Panel)

- Movie Management Module: Allows admins to add, update, and delete movie listings.
- Schedule Management Module: Enables admins to manage movie showtimes and schedules.
- Seat Management Module: Provides tools for managing seat availability and layout configurations.

6. Reporting and Analytics

- Booking Reports Module: Generates reports on ticket sales, revenue, and booking trends.
- User Activity Reports Module: Tracks user activity and engagement metrics.

7. Security and Compliance

- User Data Security Module: Ensures encryption and protection of user data.
- Payment Security Module: Complies with PCI-DSS standards for secure payment processing.

8. System Administration

- Backup and Recovery Module: Manages regular data backups and recovery processes.
- Performance Monitoring Module: Monitors system performance and uptime.
- Logging and Troubleshooting Module: Maintains detailed logs for troubleshooting and maintenance purposes.

3.TECHNOLOGIES USED:

3.1 LANGUAGES USED

1.PYTHON

Python is commonly used for web development, data analysis, artificial intelligence, and automation.

2.MY SQL

MySQL is frequently used as a relational database management system (RDBMS) for storing and managing structured data in various applications, including web development, enterprise software, and data-driven applications.

3.TINKER

Tkinter is utilized in Python for creating graphical user interfaces (GUIs) through its built-in library, facilitating the development of interactive desktop applications.

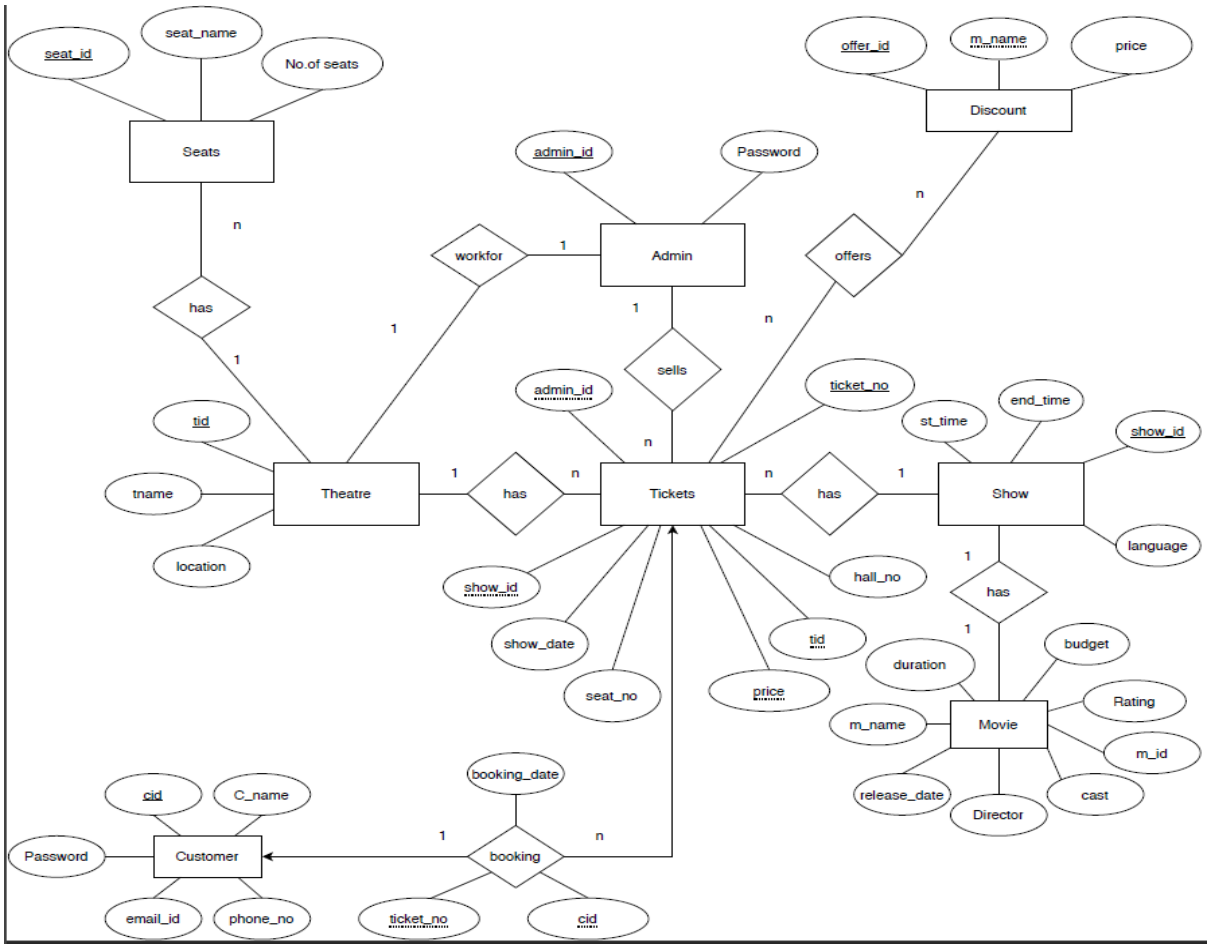
3.2 TOOLS USED

PYCHARM

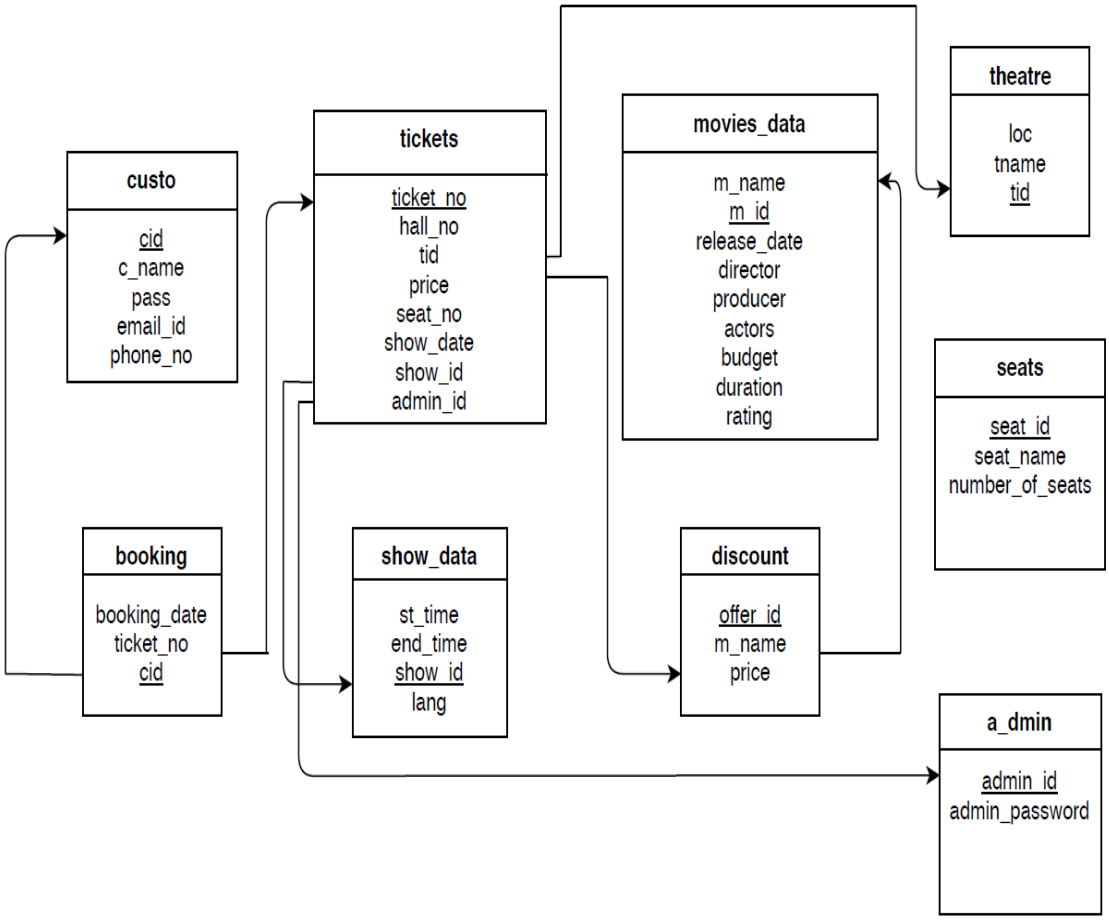
PyCharm is an integrated development environment (IDE) designed for Python programming, offering features such as code editing, debugging, and project management, thereby enhancing developer productivity and facilitating efficient Python application development.

3.REQUIREMENT ANALYSIS

3.1 ER DIAGRAM



3.2 SCHEMA DIAGRAM



4.PROGRAM CODE

```
from tkinter import *

import mysql.connector

import tkinter.messagebox as tm

from tkinter import ttk


# First Create the Database after run this program#

'''Create Database name as logindetails'''

'''Create Table name as userdatas'''

'''Create 6 Columns'''

'''

1. username

2. password

3. city

4. Moviename

5. seatname

6. Amount

'''



##Screen Size##

def scrsize(x):

    width = x.winfo_screenwidth()
```

```
    height = x.winfo_screenheight()

    x_pos = (width - 600) // 2
    y_pos = (height - 400) // 2

    x.geometry("550x350+{}+{}".format(x_pos, y_pos))



##Button Styles##
def button_style():
    return {'font': ('Arial', 12, 'bold'), 'bd': 3, 'width': 9, 'height': 1,
            'activebackground': 'yellow', 'activeforeground': 'black',
            'fg': 'white', 'bg': 'orange', 'relief': 'groove', }


def button_style1():
    return {'font': ('Arial', 12, 'bold'), 'bd': 1, 'width': 15, 'height': 10,
            'activebackground': 'yellow', 'activeforeground': 'black',
            'fg': 'white', 'bg': 'orange', 'relief': 'groove', }


def button_style2():
    return {'font': ('Arial', 10, 'bold'), 'bd': 2, 'width': 14, 'height': 1,
```

```
        'activebackground': 'yellow', 'activeforeground': 'black',
        'fg': 'black', 'bg': 'darkviolet', 'relief': 'groove', }


##Label Styles##
def labeltext():
    label_font = ('Gill Sans', 20, 'bold')
    label_width = main_screen.winfo_screenwidth()

    return label_font, label_width


# Hide Error Label in SignUp##
def hide_error_label():
    error_label.pack_forget()


# Hide Error Label in Login##
def hide_error_label1():
    error_label1.pack_forget()


##Registration or SignUp Setup##
```

```
def reg():
    global screen
    screen = Toplevel(main_screen)
    if 'screen' in globals():
        main_screen.withdraw()

    screen.title("REGISTER")

    scrsize(screen)
    screen.configure(bg="black")

    global usn_verify
    global password
    global show_password
    global entry_password

    usn_verify = StringVar()
    password = StringVar()
    show_password = BooleanVar(value=False)

    label_font, label_width = labeltext()

    l1 = Label(screen, text="Enter Your SignUp Details", bg="royalblue",
fg="black", height='2',
```

```
            font=label_font, width=label_width)
    l1.pack(pady=10)

    Label(screen, text='Enter Your Username', font=('Gill Sans', 14, 'bold'),
          bg="black", fg="white", height='2', width='30').pack()

    Entry(screen, textvariable=usn_verify, width='20', font=('Gill Sans',
14)).pack()

    Label(screen, text='Enter Your Password', font=('Gill Sans', 14, 'bold'),
          bg="black", fg="white", height='2', width='30').pack()

    entry_password = Entry(screen, textvariable=password, show='*',
width='20', font=('Gill Sans', 14))
    entry_password.pack(pady=2)

    toggle_button = Checkbutton(screen, text='Show Password',
variable=show_password,
                                command=toggle_password)
    toggle_button.pack(pady=10)

    Button(screen, text='Sign Up', **button_style(), command=signed).pack()

    reopen_button = Button(screen, text="Back", font=('Arial', 9, 'bold'),
width=8, height=1, bd=1, bg='black',
```

```
                    fg='white',
                    activebackground='yellow', activeforeground='black',
command=back1)

    reopen_button.place(relx=1.0, anchor='ne', x=-10, y=30)



def back1():
    main_screen.deiconify()
    if 'screen' in globals():
        screen.withdraw()

    elif 'login_screen' in globals():
        login_screen.withdraw()



def toggle_password():
    if show_password.get():
        entry_password.config(show='')
    else:
        entry_password.config(show='*')



##SignUp Submit##
```

```
def signed():
    # usn_info=username.get()
    pwd_info = password.get()

    if len(usn_verify.get()) < 6 or len(pwd_info) < 6:
        global error_label
        error_label = Label(screen, text="Username and Password must be at least
6 characters long",
                            bg='black', fg='red')
        error_label.pack(pady=2)
        screen.after(2500, hide_error_label)

    else:
        z = mysql.connector.connect(
            host='localhost',
            user='root',
            passwd='MoranCo3027M',
            database='logindetails'
        )

        mycursor = z.cursor()

        sql = 'INSERT INTO userdatas(username,password)VALUES(%s,%s)'
        val = (usn_verify.get(), pwd_info)
```

```
        try:
            mycursor.execute(sql, val)
            z.commit()
            Label(screen, text='Signed Up Successfully', bg='black',
fg='Yellow').pack()
            print('\nSigned Up Successfully')
        except:
            z.rollback()
        finally:
            print("\nWelcome to Film Buff\n")
            selectcity()
            mycursor.close()
            z.close()
            if 'screen' in globals():
                screen.withdraw()



##Login or SignIn Setup##
def login():
    global login_screen
    login_screen = Toplevel(main_screen)
    if 'login_screen' in globals():
        main_screen.withdraw()
```

```
    login_screen.title("Login")


    scrsize(login_screen)
    login_screen.configure(bg="black")


    global usn_verify
    global pwd_verify
    global show_password1
    global entry_password1


    usn_verify = StringVar()
    pwd_verify = StringVar()
    show_password1 = BooleanVar(value=False)


    label_font, label_width = labeltext()


    l2 = Label(login_screen, text="Enter Your Login Details", bg="darkviolet",
fg="black", height='2',
               font=label_font, width=label_width)
    l2.pack(pady=10)


    Label(login_screen, text='Username', font=('Gill Sans', 14, 'bold'),
          bg="black", fg="white", height='2', width='30').pack()
```

```
   Entry(login_screen, textvariable=usn_verify, width='20', font=('Gill Sans',
14)).pack()


   Label(login_screen, text='Password', font=('Gill Sans', 14, 'bold'),

       bg="black", fg="white", height='2', width='30').pack()


   entry_password1 = Entry(login_screen, textvariable=pwd_verify, show='*',

                           width='20', font=('Gill Sans', 14))

   entry_password1.pack(pady=2)


   toggle_button1 = Checkbutton(login_screen, text='Show Password',
variable=show_password1,

                                command=toggle_password1, bg="aquamarine")

   toggle_button1.pack(pady=10)


   Button(login_screen, text='Sign In', **button_style(),
command=submit).pack()


   reopen_button1 = Button(login_screen, text="Back", font=('Arial', 9, 'bold'),
width=8, height=1, bd=1, bg='black',

                           fg='white',

                           activebackground='yellow', activeforeground='black',
command=back2)
```

```
    reopen_button1.place(relx=1.0, anchor='ne', x=-10, y=30)



def back2():
    main_screen.deiconify()
    login_screen.withdraw()



def toggle_password1():
    if show_password1.get():
        entry_password1.config(show='')
    else:
        entry_password1.config(show='*')



##Login Submit##
def submit():
    if len(usn_verify.get()) < 6 or len(pwd_verify.get()) < 6:
        global error_label1
        error_label1 = Label(login_screen, text="Username and Password must be
at least 6 characters long",
                             bg='black', fg='red')
        error_label1.pack(pady=2)
        login_screen.after(2500, hide_error_label1)
```

```
    else:
        z1 = mysql.connector.connect(
            host='localhost',
            user='root',
            passwd='MoranCo3027M',
            database='logindetails'
        )

        mycursor = z1.cursor()
        mycursor.execute('SELECT * From userdatas WHERE username=%s AND
password=%s',
                         (usn_verify.get(), pwd_verify.get()))
        if mycursor.fetchone():
            Label(login_screen, text='Login Successfully', bg='black',
fg='Yellow').pack()
            selectcity()
            print("\nLogin Successfully")
            print("\nWelcome to Film Buff\n")
            mycursor.close()
            z1.close()
            if 'login_screen' in globals():
                login_screen.withdraw()
```

```
        else:
            tm.showerror("Error", "Invalid Username or Password")



##select city after login##
def selectcity():
    global city, lbl_text
    city = Toplevel(main_screen)

    if 'screen' in globals():
        screen.withdraw()

    elif 'login_screen' in globals():
        login_screen.withdraw()

    city.title("Select City & Movie")
    scrsize(city)
    city.configure(bg="black")
    label_font, label_width = labeltext()

    lbl_text = StringVar()
    lbl_text.set("Select Your City")
```

```
    label = Label(city, textvariable=lbl_text, height=1, width=label_width,
                  font=label_font, bg='skyblue', fg='black')
    label.pack()

    cities = ["Madurai", "Chennai", "Thirunelveli", "Coimbatore", "Salem",
"Tiruchirappalli",
              "Nagercoil", "Thanjavur", "Pallavaram", "Erode", "Dindigul", "Ooty",
"Kodaikanal",
              "Kanchipuram", "Puducherry", "Rameswaram", "Vellore",
"Pudukkottai", "Tenkasi"]

    global sc
    sc = StringVar()
    sc.set("Cities")

    font_config = ('bold', 14)
    width_config = 15
    height_config = 1

    menubutton = Menubutton(city, textvariable=sc, font=font_config,
width=width_config,
                            height=height_config, bd=0, bg="azure", fg="black",
                            activebackground="yellow", activeforeground='black')
```

```
    menubutton.menu = Menu(menubutton, tearoff=0, bd=0, bg="white",
fg="black",

                           activebackground="yellow", activeforeground='black',
font=font_config)

    menubutton["menu"] = menubutton.menu


    for city_name in cities:

        menubutton.menu.add_radiobutton(label=city_name, variable=sc,
value=city_name,

                                        command=lambda: print_ans(lbl_text, menubutton,
label, city))

    menubutton.pack()

    label.pack()


    movies_list(city)


    reopen_button2 = Button(city, text="Sign Out", font=('Arial', 9, 'bold'),
width=8, height=1, bd=1, bg='dimgrey',

                            fg='white',

                            activebackground='yellow', activeforeground='black',
command=lambda: signout(city))


    reopen_button2.place(relx=1.0, anchor='ne', x=-10, y=8)

    reopen_button2.lift()
```

```
def signout(city):
    main_screen.deiconify()
    if 'screen' in globals():
        screen.withdraw()

    elif 'login_screen' in globals():
        login_screen.destroy()
    city.withdraw()



def movies_list(city):
    btn1 = Button(city, text='Movie 1', **button_style1(), command=lambda:
seat_selection(city, "Movie 1"))
    btn1.pack(side=LEFT, padx=15, pady=10, anchor=CENTER)

    btn2 = Button(city, text='Movie 2', **button_style1(), command=lambda:
seat_selection(city, "Movie 2"))
    btn2.pack(side=LEFT, padx=15, pady=10, anchor=CENTER)

    btn3 = Button(city, text='Movie 3', **button_style1(), command=lambda:
seat_selection(city, "Movie 3"))
    btn3.pack(side=LEFT, padx=15, pady=10, anchor=CENTER)
```

```
def seat_selection(city, movie_name_fetch):
    global seat, movie_name
    seat = Toplevel(city)
    if 'city' in globals():
        city.withdraw()
    movie_name = movie_name_fetch

    global s1, s2, s3, s4, s5, s6, s7, s8, s9, s10, s11, s12, s13, s14, s15, s16, s17,
s18, s19, s20, s21, s22
    global s23, s24, s25, s26, s27, s28, s29, s30, s31, s32, s33, s34, s35, s36, s37,
s38, s39, s40, s41, s42
    global s43, s44, s45, s46, s47, s48, s49, s50, s51, s52, s53, s54, s55, s56, s57,
s58, s59, s60, s61, s62
    global s63, s64, s65, s66, s67, s68, s69, s70, s71, s72, s73, s74, s75, s76, s77,
s78, s79, s80, s81, s82
    global s83, s84, s85, s86, s87, s88, s89, s90, s91, s92, s93, s94, s95, s96

    s1 = IntVar()
    s2 = IntVar()
    s3 = IntVar()
    s4 = IntVar()
    s5 = IntVar()
    s6 = IntVar()
    s7 = IntVar()
```

```
s8 = IntVar()
s9 = IntVar()
s10 = IntVar()
s11 = IntVar()
s12 = IntVar()
s13 = IntVar()
s14 = IntVar()
s15 = IntVar()
s16 = IntVar()
s17 = IntVar()
s18 = IntVar()
s19 = IntVar()
s20 = IntVar()
s21 = IntVar()
s22 = IntVar()
s23 = IntVar()
s24 = IntVar()
s25 = IntVar()
s26 = IntVar()
s27 = IntVar()
s28 = IntVar()
s29 = IntVar()
s30 = IntVar()
```

```
s31 = IntVar()
s32 = IntVar()
s33 = IntVar()
s34 = IntVar()
s35 = IntVar()
s36 = IntVar()
s37 = IntVar()
s38 = IntVar()
s39 = IntVar()
s40 = IntVar()
s41 = IntVar()
s42 = IntVar()
s43 = IntVar()
s44 = IntVar()
s45 = IntVar()
s46 = IntVar()
s47 = IntVar()
s48 = IntVar()
s49 = IntVar()
s50 = IntVar()
s51 = IntVar()
s52 = IntVar()
s53 = IntVar()
```

```
s54 = IntVar()
s55 = IntVar()
s56 = IntVar()
s57 = IntVar()
s58 = IntVar()
s59 = IntVar()
s60 = IntVar()
s61 = IntVar()
s62 = IntVar()
s63 = IntVar()
s64 = IntVar()
s65 = IntVar()
s66 = IntVar()
s67 = IntVar()
s68 = IntVar()
s69 = IntVar()
s70 = IntVar()
s71 = IntVar()
s72 = IntVar()
s73 = IntVar()
s74 = IntVar()
s75 = IntVar()
s76 = IntVar()
```

```
s77 = IntVar()
s78 = IntVar()
s79 = IntVar()
s80 = IntVar()
s81 = IntVar()
s82 = IntVar()
s83 = IntVar()
s84 = IntVar()
s85 = IntVar()
s86 = IntVar()
s87 = IntVar()
s88 = IntVar()
s89 = IntVar()
s90 = IntVar()
s91 = IntVar()
s92 = IntVar()
s93 = IntVar()
s94 = IntVar()
s95 = IntVar()
s96 = IntVar()

scrsize(seat)
```

```
    seat.title("Seat Selection")

    seat.configure(bg="black")


    label_font, label_width = labeltext()


    label_0 = Label(seat, text="SELECT YOUR SEATS", width=label_width,
font=label_font).pack()


    reopen_btn1 = Button(seat, text="Back", font=('Arial', 9, 'bold'), width=7,
height=1, bd=1, bg='black', fg='white',

                         activebackground='yellow', activeforeground='black',
command=back_1)


    reopen_btn1.place(relx=1.0, anchor='ne', x=-10, y=8)


    reopen_btn2 = Button(seat, text="LogOut", font=('Arial', 9, 'bold'), width=8,
height=1, bd=1, bg='black',

                         fg='white',

                         activebackground='yellow', activeforeground='black',
command=signout_1)


    reopen_btn2.place(relx=0.0, anchor='nw', x=10, y=8)


    # Row A#

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s1,
```

```
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=50, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s2,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=80, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s3,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=110, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s4,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=140, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s5,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=170, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s6,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=200, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s7,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=230, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s8,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=260, y=65)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s9,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=290, y=65)
```

```
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s10,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=320, y=65)

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s11,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=350, y=65)

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s12,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=380, y=65)

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s13,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=410, y=65)

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s14,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=440, y=65)

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s15,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=470, y=65)

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s16,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=500, y=65)


    # Row B#

    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s17,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=50, y=95)
```

```
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s18,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=80, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s19,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=110, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s20,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=140, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s21,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=170, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s22,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=200, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s23,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=230, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s24,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=260, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s25,
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=290, y=95)
Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s26,
```

```
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=320, y=95)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s27,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=350, y=95)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s28,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=380, y=95)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s29,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=410, y=95)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s30,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=440, y=95)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s31,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=470, y=95)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s32,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=500, y=95)


   # Row C#

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s33,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=50, y=125)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s34,
```

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=80, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s35,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=110, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s36,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=140, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s37,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=170, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s38,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=200, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s39,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=230, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s40,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=260, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s41,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=290, y=125)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s42,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=320, y=125)

```
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s43,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=350, y=125)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s44,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=380, y=125)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s45,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=410, y=125)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s46,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=440, y=125)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s47,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=470, y=125)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s48,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=500, y=125)


    # Row D#
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s49,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=50, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s50,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=80, y=155)
```

```
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s51,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=110, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s52,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=140, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s53,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=170, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s54,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=200, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s55,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=230, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s56,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=260, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s57,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=290, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s58,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=320, y=155)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s59,
```

```
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=350, y=155)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s60,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=380, y=155)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s61,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=410, y=155)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s62,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=440, y=155)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s63,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=470, y=155)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s64,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=500, y=155)


   # Row E#

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s65,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=50, y=185)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s66,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=80, y=185)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s67,
```

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=110, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s68,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=140, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s69,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=170, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s70,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=200, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s71,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=230, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s72,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=260, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s73,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=290, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s74,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=320, y=185)

Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s75,

bg="grey", fg="black", activebackground="darkviolet", activeforeground="black").place(x=350, y=185)

```
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s76,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=380, y=185)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s77,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=410, y=185)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s78,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=440, y=185)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s79,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=470, y=185)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s80,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=500, y=185)


    # Row F#
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s81,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=50, y=215)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s82,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=80, y=215)
    Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s83,
                bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=110, y=215)
```

```
   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s84,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=140, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s85,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=170, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s86,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=200, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s87,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=230, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s88,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=260, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s89,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=290, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s90,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=320, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s91,

            bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=350, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s92,
```

```
        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=380, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s93,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=410, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s94,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=440, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s95,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=470, y=215)

   Checkbutton(seat, text="", onvalue=1, offvalue=0, height=2, variable=s96,

        bg="grey", fg="black", activebackground="darkviolet",
activeforeground="black").place(x=500, y=215)


   Label(seat, text="Screen", bg="black", fg="darkviolet", font=('Gill Sans', 9,
'bold')).place(x=255, y=265)

   canvas = Canvas(seat, width=200, height=10, bg="lightcyan")

   canvas.place(x=180, y=285)

   rect_width = 350

   rect_height = 100

   # seat selection #

   Button(seat, text="Confirm Booking", **button_style2(),
command=seat_confirmation).pack(side=BOTTOM)


   font_new = ('Gill Sans', 12, 'bold')
```

```
    Label(seat, text="A", bg="black", fg="darkviolet",
font=font_new).place(x=10, y=70)

    Label(seat, text="B", bg="black", fg="darkviolet",
font=font_new).place(x=10, y=100)

    Label(seat, text="C", bg="black", fg="darkviolet",
font=font_new).place(x=10, y=130)

    Label(seat, text="D", bg="black", fg="darkviolet",
font=font_new).place(x=10, y=160)

    Label(seat, text="E", bg="black", fg="darkviolet",
font=font_new).place(x=10, y=190)

    Label(seat, text="F", bg="black", fg="darkviolet", font=font_new).place(x=10,
y=220)


    Label(seat, text="1", bg="black", fg="darkviolet",
font=font_new).place(x=52, y=40)

    Label(seat, text="2", bg="black", fg="darkviolet",
font=font_new).place(x=82, y=40)

    Label(seat, text="3", bg="black", fg="darkviolet",
font=font_new).place(x=112, y=40)

    Label(seat, text="4", bg="black", fg="darkviolet",
font=font_new).place(x=142, y=40)

    Label(seat, text="5", bg="black", fg="darkviolet",
font=font_new).place(x=172, y=40)

    Label(seat, text="6", bg="black", fg="darkviolet",
font=font_new).place(x=202, y=40)

    Label(seat, text="7", bg="black", fg="darkviolet",
font=font_new).place(x=232, y=40)
```

```
    Label(seat, text="8", bg="black", fg="darkviolet",
font=font_new).place(x=262, y=40)

    Label(seat, text="9", bg="black", fg="darkviolet",
font=font_new).place(x=292, y=40)

    Label(seat, text="10", bg="black", fg="darkviolet",
font=font_new).place(x=320, y=40)

    Label(seat, text="11", bg="black", fg="darkviolet",
font=font_new).place(x=350, y=40)

    Label(seat, text="12", bg="black", fg="darkviolet",
font=font_new).place(x=380, y=40)

    Label(seat, text="13", bg="black", fg="darkviolet",
font=font_new).place(x=410, y=40)

    Label(seat, text="14", bg="black", fg="darkviolet",
font=font_new).place(x=440, y=40)

    Label(seat, text="15", bg="black", fg="darkviolet",
font=font_new).place(x=470, y=40)

    Label(seat, text="16", bg="black", fg="darkviolet",
font=font_new).place(x=500, y=40)


    seat.mainloop()



def seat_confirmation():

    global seat_name, Amount

    seat_name = ""

    Amount = 0
```

```
if s1.get() == 1:
    seat_name += "A1,"
    Amount += 190

if s2.get() == 1:
    seat_name += "A2,"
    Amount += 190

if s3.get() == 1:
    seat_name += "A3,"
    Amount += 190

if s4.get() == 1:
    seat_name += "A4,"
    Amount += 190

if s5.get() == 1:
    seat_name += "A5,"
    Amount += 190

if s6.get() == 1:
    seat_name += "A6,"
```

```
    Amount += 190

if s7.get() == 1:
    seat_name += "A7,"
    Amount += 190

if s8.get() == 1:
    seat_name += "A8,"
    Amount += 190

if s9.get() == 1:
    seat_name += "A9,"
    Amount += 190

if s10.get() == 1:
    seat_name += "A10,"
    Amount += 190

if s11.get() == 1:
    seat_name += "A11,"
    Amount += 190

if s12.get() == 1:
```

```
    seat_name += "A12,"
    Amount += 190

if s13.get() == 1:
    seat_name += "A13,"
    Amount += 190

if s14.get() == 1:
    seat_name += "A14,"
    Amount += 190

if s15.get() == 1:
    seat_name += "A15,"
    Amount += 190

if s16.get() == 1:
    seat_name += "A16,"
    Amount += 190
####################################
if s17.get() == 1:
    seat_name += "B1,"
    Amount += 190
```

```
if s18.get() == 1:
    seat_name += "B2,"
    Amount += 190

if s19.get() == 1:
    seat_name += "B3,"
    Amount += 190

if s20.get() == 1:
    seat_name += "B4,"
    Amount += 190

if s21.get() == 1:
    seat_name += "B5,"
    Amount += 190

if s22.get() == 1:
    seat_name += "B6,"
    Amount += 190

if s23.get() == 1:
    seat_name += "B7,"
    Amount += 190
```

```
if s24.get() == 1:
    seat_name += "B8,"
    Amount += 190

if s25.get() == 1:
    seat_name += "B9,"
    Amount += 190

if s26.get() == 1:
    seat_name += "B10,"
    Amount += 190

if s27.get() == 1:
    seat_name += "B11,"
    Amount += 190

if s28.get() == 1:
    seat_name += "B12,"
    Amount += 190

if s29.get() == 1:
    seat_name += "B13,"
```

```
    Amount += 190

if s30.get() == 1:
    seat_name += "B14,"
    Amount += 190

if s31.get() == 1:
    seat_name += "B15,"
    Amount += 190

if s32.get() == 1:
    seat_name += "B16,"
    Amount += 190
#####################################
if s33.get() == 1:
    seat_name += "C1,"
    Amount += 190

if s34.get() == 1:
    seat_name += "C2,"
    Amount += 190

if s35.get() == 1:
```

```
    seat_name += "C3,"
    Amount += 190

if s36.get() == 1:
    seat_name += "C4,"
    Amount += 190

if s37.get() == 1:
    seat_name += "C5,"
    Amount += 190

if s38.get() == 1:
    seat_name += "C6,"
    Amount += 190

if s39.get() == 1:
    seat_name += "C7,"
    Amount += 190

if s40.get() == 1:
    seat_name += "C8,"
    Amount += 190
```

```
if s41.get() == 1:
    seat_name += "C9,"
    Amount += 190

if s42.get() == 1:
    seat_name += "C10,"
    Amount += 190

if s43.get() == 1:
    seat_name += "C11,"
    Amount += 190

if s44.get() == 1:
    seat_name += "C12,"
    Amount += 190

if s45.get() == 1:
    seat_name += "C13,"
    Amount += 190

if s46.get() == 1:
    seat_name += "C14,"
    Amount += 190
```

```
if s47.get() == 1:
    seat_name += "C15,"
    Amount += 190

if s48.get() == 1:
    seat_name += "C16,"
    Amount += 190
##########################################
if s49.get() == 1:
    seat_name += "D1,"
    Amount += 190

if s50.get() == 1:
    seat_name += "D2,"
    Amount += 190

if s51.get() == 1:
    seat_name += "D3,"
    Amount += 190

if s52.get() == 1:
    seat_name += "D4,"
```

```
    Amount += 190

if s53.get() == 1:
    seat_name += "D5,"
    Amount += 190

if s54.get() == 1:
    seat_name += "D6,"
    Amount += 190

if s55.get() == 1:
    seat_name += "D7,"
    Amount += 190

if s56.get() == 1:
    seat_name += "D8,"
    Amount += 190

if s57.get() == 1:
    seat_name += "D9,"
    Amount += 190

if s58.get() == 1:
```

```
    seat_name += "D10,"
    Amount += 190

if s59.get() == 1:
    seat_name += "D11,"
    Amount += 190

if s60.get() == 1:
    seat_name += "D12,"
    Amount += 190

if s61.get() == 1:
    seat_name += "D13,"
    Amount += 190

if s62.get() == 1:
    seat_name += "D14,"
    Amount += 190

if s63.get() == 1:
    seat_name += "D15,"
    Amount += 190
```

```
if s64.get() == 1:
    seat_name += "D16,"
    Amount += 190
#####################################
if s65.get() == 1:
    seat_name += "E1,"
    Amount += 190

if s66.get() == 1:
    seat_name += "E2,"
    Amount += 190

if s67.get() == 1:
    seat_name += "E3,"
    Amount += 190

if s68.get() == 1:
    seat_name += "E4,"
    Amount += 190

if s69.get() == 1:
    seat_name += "E5,"
    Amount += 190
```

```
if s70.get() == 1:
    seat_name += "E6,"
    Amount += 190

if s71.get() == 1:
    seat_name += "E7,"
    Amount += 190

if s72.get() == 1:
    seat_name += "E8,"
    Amount += 190

if s73.get() == 1:
    seat_name += "E9,"
    Amount += 190

if s74.get() == 1:
    seat_name += "E10,"
    Amount += 190

if s75.get() == 1:
    seat_name += "E11,"
```

```
    Amount += 190

if s76.get() == 1:
    seat_name += "E12,"
    Amount += 190

if s77.get() == 1:
    seat_name += "E13,"
    Amount += 190

if s78.get() == 1:
    seat_name += "E14,"
    Amount += 190

if s79.get() == 1:
    seat_name += "E15,"
    Amount += 190

if s80.get() == 1:
    seat_name += "E16,"
    Amount += 190
#####################################
if s81.get() == 1:
```

```
    seat_name += "F1,"
    Amount += 190

if s82.get() == 1:
    seat_name += "F2,"
    Amount += 190

if s83.get() == 1:
    seat_name += "F3,"
    Amount += 190

if s84.get() == 1:
    seat_name += "F4,"
    Amount += 190

if s85.get() == 1:
    seat_name += "F5,"
    Amount += 190

if s86.get() == 1:
    seat_name += "F6,"
    Amount += 190
```

```
if s87.get() == 1:
    seat_name += "F7,"
    Amount += 190

if s88.get() == 1:
    seat_name += "F8,"
    Amount += 190

if s89.get() == 1:
    seat_name += "F9,"
    Amount += 190

if s90.get() == 1:
    seat_name += "F10,"
    Amount += 190

if s91.get() == 1:
    seat_name += "F11,"
    Amount += 190

if s92.get() == 1:
    seat_name += "F12,"
    Amount += 190
```

```
    if s93.get() == 1:
        seat_name += "F13,"
        Amount += 190

    if s94.get() == 1:
        seat_name += "F14,"
        Amount += 190

    if s95.get() == 1:
        seat_name += "F15,"
        Amount += 190

    if s96.get() == 1:
        seat_name += "F16,"
        Amount += 190
    ##############################################

    updating_values(seat)


def back_1():
    city.deiconify()
```

```
    seat.withdraw()



def signout_1():
    main_screen.deiconify()
    if 'screen' in globals():
        screen.withdraw()

    elif 'login_screen' in globals():
        login_screen.withdraw()
    city.withdraw()
    seat.withdraw()



def updating_values(seat):
    z = mysql.connector.connect(
        host='localhost',
        user='root',
        passwd='MoranCo3027M',
        database='logindetails'
    )

    mycursor = z.cursor()
```

```
    sql1 = "UPDATE userdatas SET
city='{}',Moviename='{}',seatname='{}',Amount='{}' where username='{}'"
    val1 = sql1.format(lbl_text.get(), movie_name, seat_name, Amount,
usn_verify.get())
    print(val1)
    try:
        print(lbl_text.get())
        print("Your Seats: " + seat_name)
        print("Total Cost: " + str(Amount))
        print(movie_name)
        mycursor.execute(val1)
        z.commit()
        booking_successfully(seat)
    except:
        z.rollback()
    finally:
        mycursor.close()
        z.close()



def booking_successfully(seat):
    global booked, seat_name, Amount
```

```
    booked = Toplevel(seat)
    if 'seat' in globals():
        seat.withdraw()

    booked.title('Tickets')
    scrsize(booked)
    booked.configure(bg="black")

    label_font, label_width = labeltext()

    Label(booked, text="Tickets Booked Successfully", bg="pink", fg="black",
height='2',
          font=label_font, width=label_width).pack()

    reopen_btn3 = Button(booked, text="Back", font=('Arial', 9, 'bold'), width=7,
height=1, bd=1, bg='black',
                         fg='white',
                         activebackground='yellow', activeforeground='black',
command=back_2)

    reopen_btn3.place(relx=1.0, anchor='ne', x=-10, y=8)

    reopen_btn4 = Button(booked, text="LogOut", font=('Arial', 9, 'bold'),
width=8, height=1, bd=1, bg='black',
                         fg='white',
```

```
                    activebackground='yellow', activeforeground='black',
command=signout_2)


   reopen_btn4.place(relx=0.0, anchor='nw', x=10, y=8)


   Label(booked, text="Your Tickets\n~~~~~~", bg="black", fg="darkviolet",
height='3',
        font=label_font, width=label_width).pack()


   seat_label = Label(booked, text="Your Seats: " + seat_name, bg="yellow",
fg="black", height='1',
                    font=label_font)
   seat_label.pack()


   cost_label = Label(booked, text="Total Cost: " + str(Amount), bg="yellow",
fg="black", height='1',
                    font=label_font)
   cost_label.pack()



def back_2():
   city.deiconify()
   booked.withdraw()
```

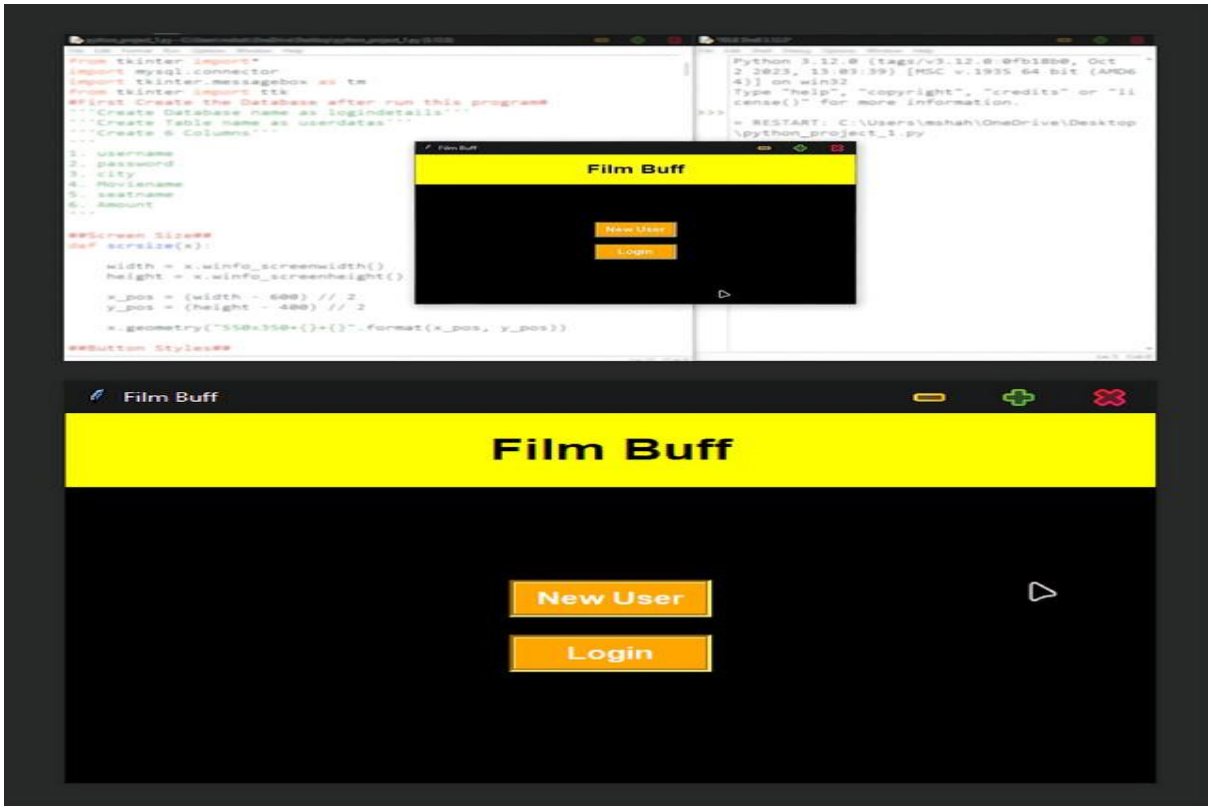
```
def signout_2():
    main_screen.deiconify()
    if 'screen' in globals():
        screen.withdraw()

    elif 'login_screen' in globals():
        login_screen.withdraw()
    city.withdraw()
    seat.withdraw()
    booked.withdraw()



def print_ans(lbl_text, menubutton, label, city):
    global scity
    scity = sc.get()
    print("Selected City: " + scity + "\n")
    lbl_text.set("Your City: " + scity)
    label.config(bg='orangered')

    global changecity
    changecity = StringVar()
    changecity.set("Change\nYou're City")
```

```
    menubutton.config(textvariable=changecity, font=('Arial', 9, 'bold'), width=8,
height=1, bd=4,
                      bg='pink', fg='black', activebackground='yellow',
activeforeground='black')

    x = label.winfo_x()
    y = label.winfo_y()

    menubutton.place(x=x + 5, y=y + 3)
    menubutton.lift()



##Main Screen for All windows##
def main_account_screen():
    global main_screen
    main_screen = Tk()

    scrsize(main_screen)
    main_screen.configure(bg="black")

    label_font, label_width = labeltext()

    main_screen.title('Film Buff')
    Label(main_screen, text="Film Buff", bg="yellow", fg="black", height='2',
```

```
        font=label_font, width=label_width).pack()
    Label(main_screen, text='', bg="black").pack()

    btn = Button(text='New User', command=reg, **button_style())
    btn.place(relx=0.5, rely=0.5, anchor=CENTER)

    btn1 = Button(text='Login', command=login, **button_style())
    btn1.place(relx=0.5, rely=0.6, anchor=N)

    main_screen.mainloop()


main_account_screen()
```

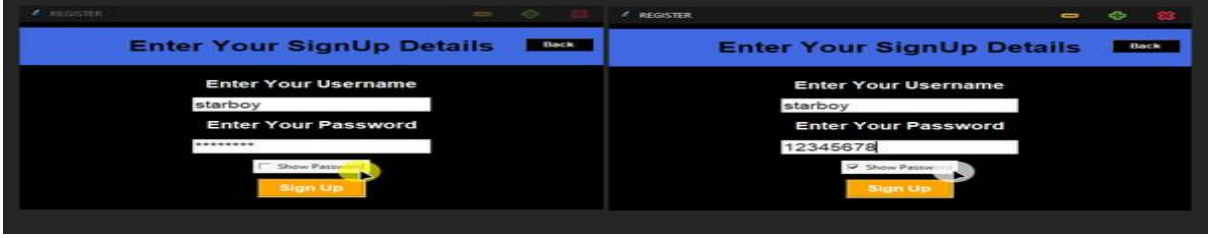
5.RESULT

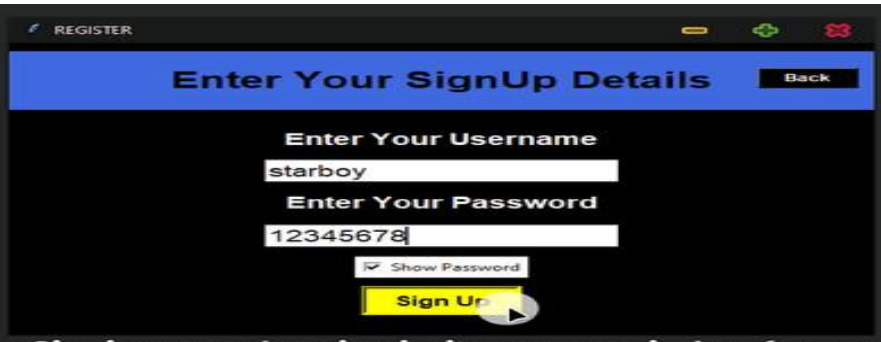
"Create 6 Columns"
'username' , 'password' , 'city' ,
'Moviename' , 'seatname' , 'Amount'.
phpMyAdmin
Recent
Favourites
New
information_schema
logindetails
New
userdatas
mysql
performance_schema
phpmyadmin
Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
Operations
Tracking
Triggers
Table structure
Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 username varchar(255) utf8mb4_general_ci No None Change Drop More
2 password varchar(255) utf8mb4_general_ci No None Change Drop More
3 city varchar(255) utf8mb4_general_ci No None Change Drop More
4 Moviename varchar(255) utf8mb4_general_ci No None Change Drop More
5 seatname varchar(255) utf8mb4_general_ci No None Change Drop More
6 Amount varchar(255) utf8mb4_general_ci No None Change Drop More
Check all
With selected:
Browse
Change
Drop
Primary
Unique
Index
Spatial
Fulltext
Add to central columns
Remove from central columns
Print
Propose table structure
Track table
Move columns
Normalise
Add
1
column(s)
after Amount
Go


Film Buff
New User
Login
Film Buff
Film Buff
New User
Login

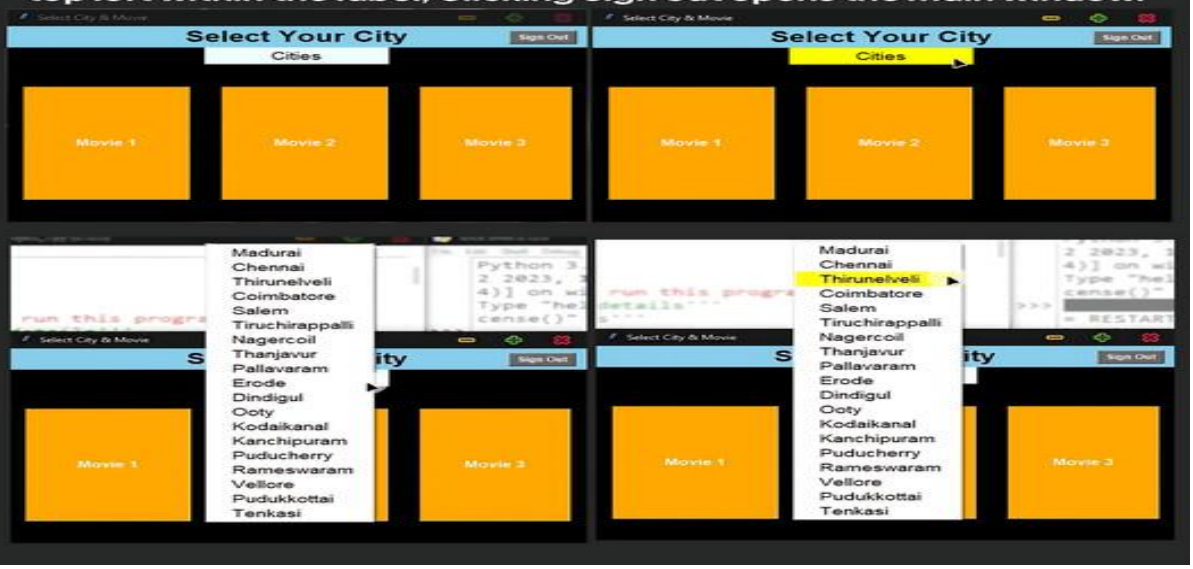
In the Sign Up and Login window, ensure usernames and password are at least 6 characters long. If less than 6 characters are entered, display an error message, which disappears after 3 seconds.

I added a "Show Password" option in the Sign Up and Login window for toggling password visibility.



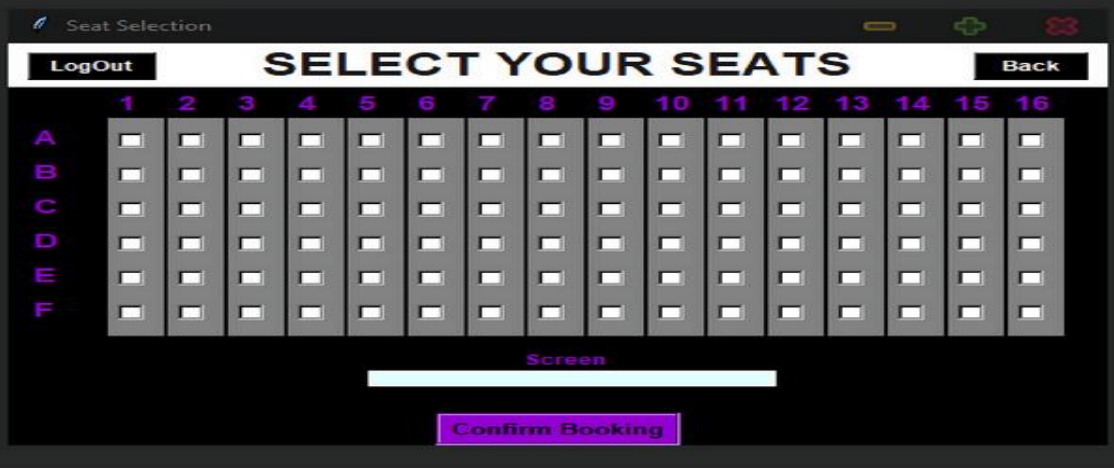


After Signing up or logging in, it opens a window for users to choose their city and movies, And I added a Sign Out button at the top left within the label, Clicking Sign out opens the main window.



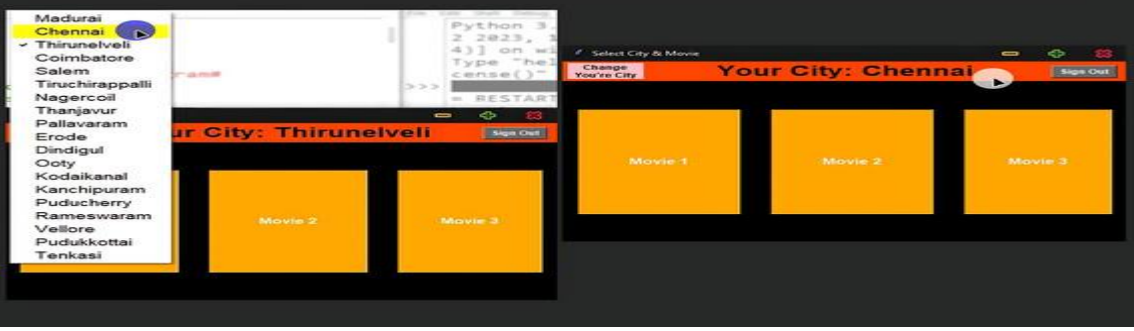
Now, the user selects their preferred movie.

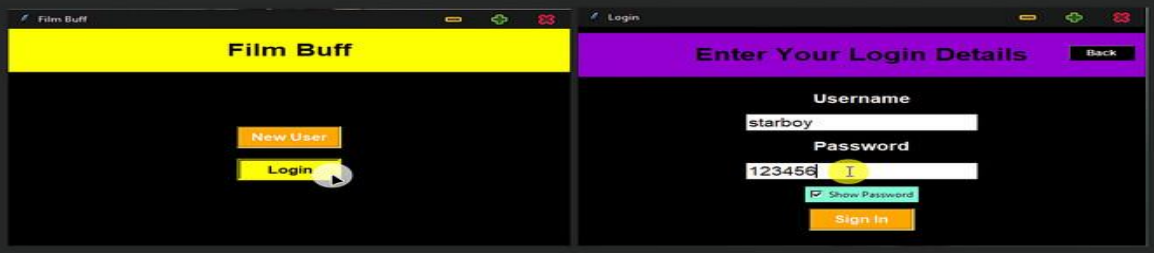
After selecting a movie, the seat selection window opens, and I added logout and back buttons in label.



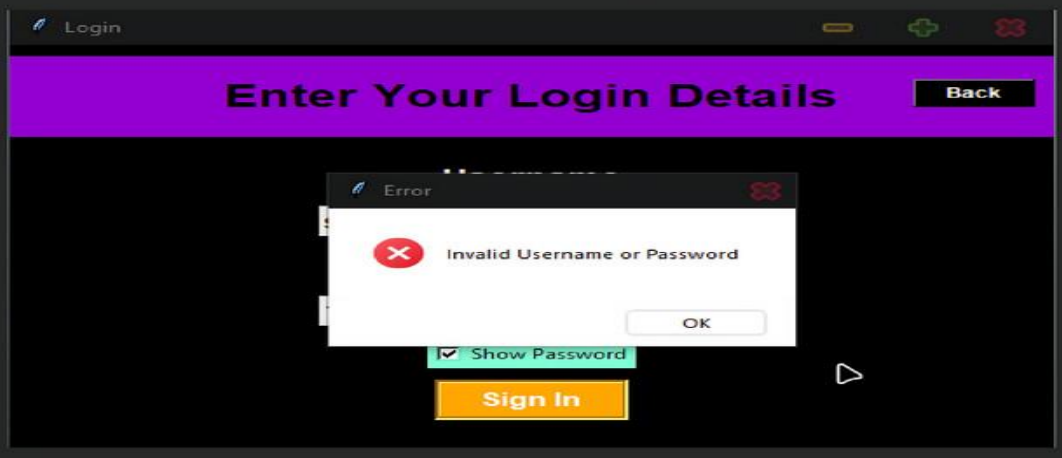
After city selection, the label switches to "Your City", and the menu button relocates to the top-left side of the label.

For city changes after selection, I display the preview as shown.

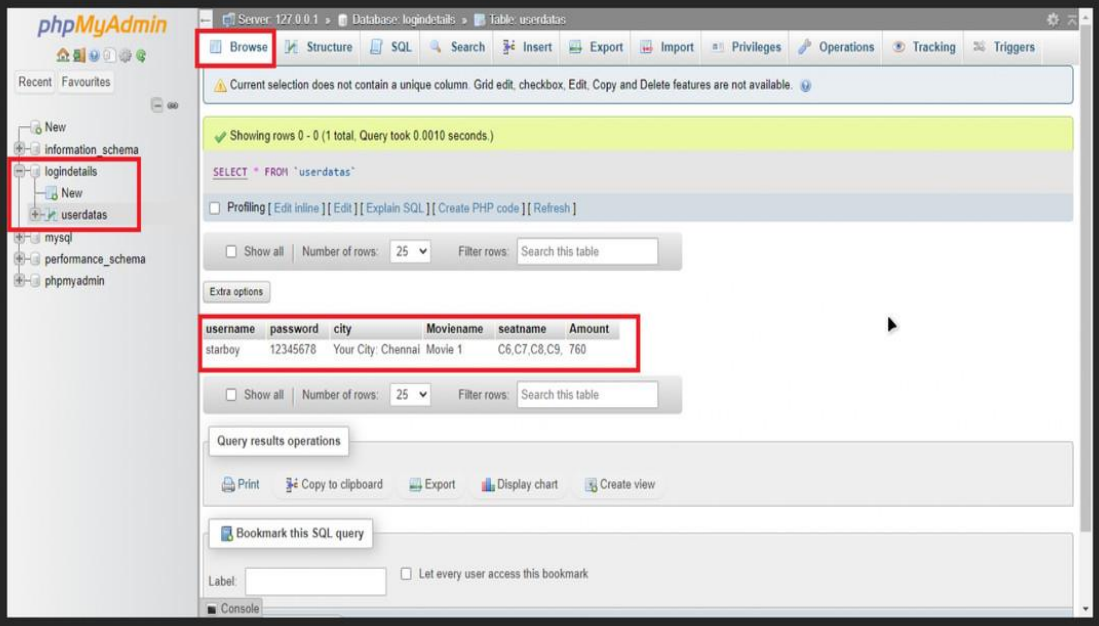




In the login window, entering incorrect username and password displays an error.



All data, Including username, password, city, Movie name, seat name and total amount is stored in the SQL database.



7.CONCLUSION

In conclusion, the development of an online movie ticket booking system offers significant benefits to both users and theater administrators. By providing a user-friendly platform for booking movie tickets from any location, the system enhances convenience and eliminates the need for physical queues. Additionally, robust management tools empower theater administrators to efficiently manage movie schedules and seat availability, while features such as real-time updates, secure payment processing, and notification systems enhance the overall user experience. With its focus on usability, security, and reliability, the online movie ticket booking system project aims to streamline operations, improve customer satisfaction, and contribute to the modernization of the movie-going experience.

REFERENCES

REFERENCES Tutorials and Guides 1. Tkinter Basics: Official Tkinter Documentation: Tkinter – Python Interface to Tcl/Tk o Tkinter Tutorial: Python Tkinter Tutorial 2. MySQL with Python: Official MySQL Connector/Python Developer Guide: MySQL Connector/Python Developer Guide o MySQL and Python Tutorial: MySQL Database Connection with Python 3. Combining Tkinter and MySQL: Blog Post Example: Python Tkinter and MySQL Database o YouTube Video Tutorial: Python Tkinter with MySQL - CRUD Operation