



School of Computer Science and Engineering

Project Title:

Booking Cabs within LPU using Python



FLASH-CAB

Ready To Go

Submitted By :

Registration No.	Name	RollNo.	Section
12102106	Pradip Dev	A 31	K21PG
12102180	Sunil Pal	A 33	K21PG
12101757	Shivam	A 02	K21PG

Submitted To: Dr. Ramandeep Sandhu Ma'am

Index

1. Declaration
2. Introduction
3. Modules
4. Objectives
5. Technologies Used
6. DFD Design
7. Login Page
8. Sign Up Page
9. Booking Page
10. Admin Page
11. Booking Form
12. Coding Screenshot
13. References

Declaration

I declare that this CA is our group work. we have not copied it from any other students' work or from any other source except where due acknowledgement is made explicitly in the text, nor has any part been written for us by any other person.

Introduction

Cab booking Management System allows passenger to get cab easily from nearest location and allows driver to manage all the passengers. The proposed cab booking management system ensures that the users can book the cab as per their requirement by logging on to the website. It allows user to book their cabs online, manage their booking and cancel their booking.

This project aims to provide software to students studying at lovely professional university. Cab Booking Management System (Flash-Cab) is the software which will provide booking of the car from one destination to another destination all over campus. Cab booking system consists of different model's such as Registration (admin, user, driver), login (admin, user, driver), Available routs, Booking Request. This project will provide the services to the students as well as faculties in the campus. Currently there is no booking system of car in the campus. So, our project aims to help all the student and staffs who want to book their car using their phones and get the car at their destination. Due to cab booking system student can easily get cabs on time within limited time.

Team

Pradeep Dev: 12102106

Sunil Pal: 12102180

Shivam: 12101757

Modules

In this Cab Booking Management system, there are Different modules which are listed below:

- ✓ Home page
- ✓ Login
- ✓ Register
- ✓ Booking Request

Home Page:

Home page or front page consists of three buttons Login, New User, Available routs. Where login has three types Admin, User and driver. User will be able to create account, Login, Book Cabs and View the price for the Ride, Similarly, Drivers are able to accept the request for the cab booking, Show the routs, Price of the ride. Admin is able to manage all the rides.

Login:

Login has input Username and password and login button. It has Create new account option if user is not already registered.

Register:

Register consists of Name, Gender (M or F), Mobile no, Email ID, LPU Reg No.

Booking Request:

Booking Request consists of Mobile. No, From, to (Destination), Price, Day, Time And booking Option.

Objective :

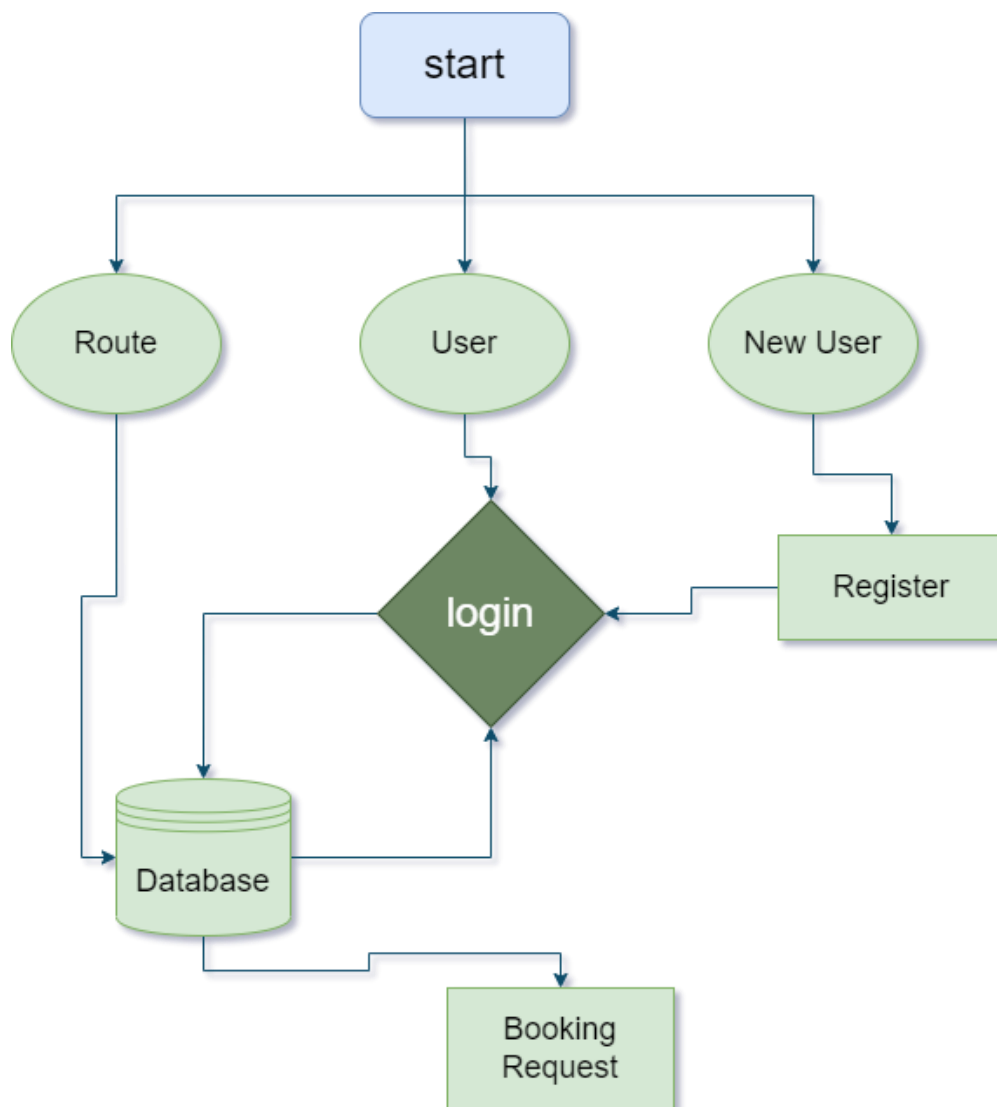
1. Easy To Go Any place in LPU at For limited rides.
2. The system lets the driver know the location so that pickups can be easily arranged.
3. System brings simplicity to both drivers and customers as customers can choose to book.
4. The goal of an online cab booking system is to bring riders and drivers together on one platform.

Technologie

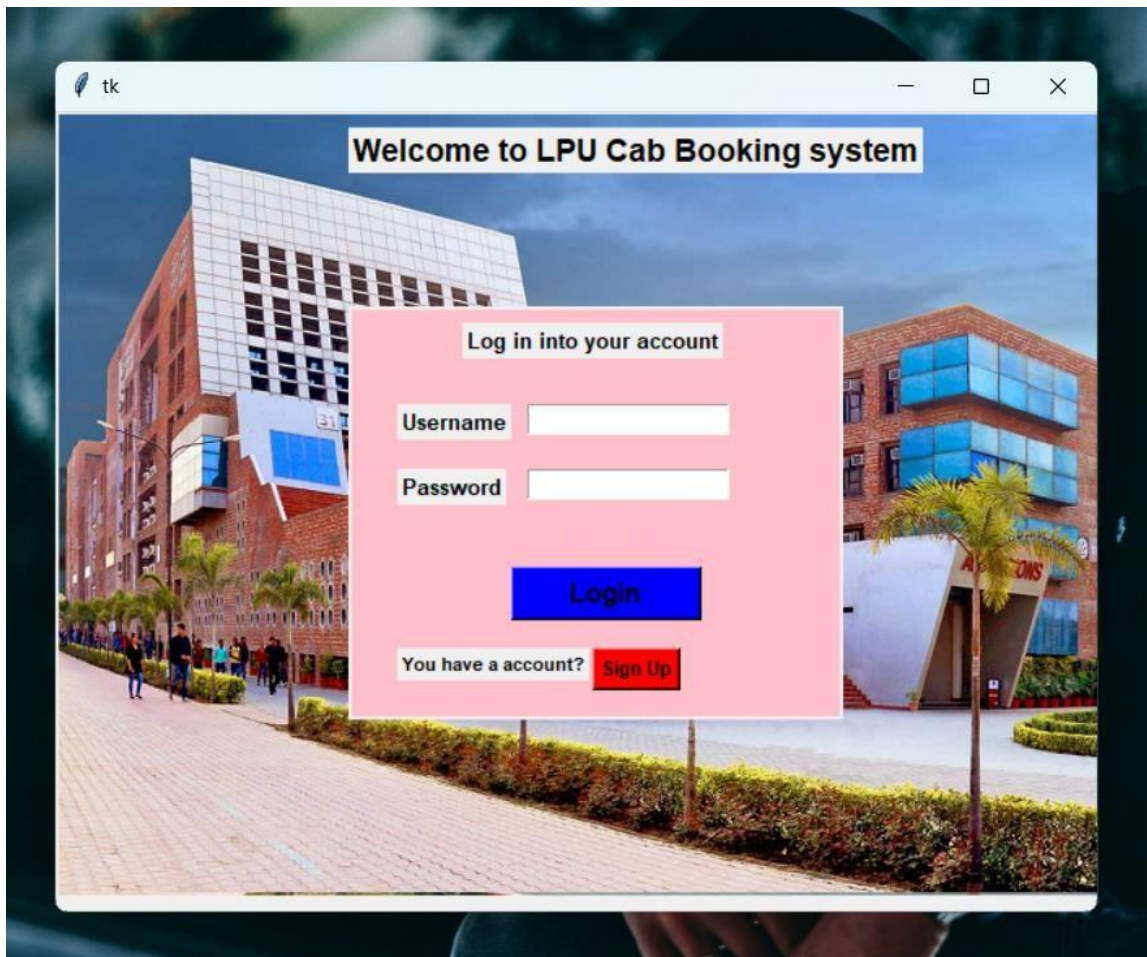
1. Only Python is used in that Project
2. Python GUI Tkinter Used .
3. Personal Database can be used.

DFD Design

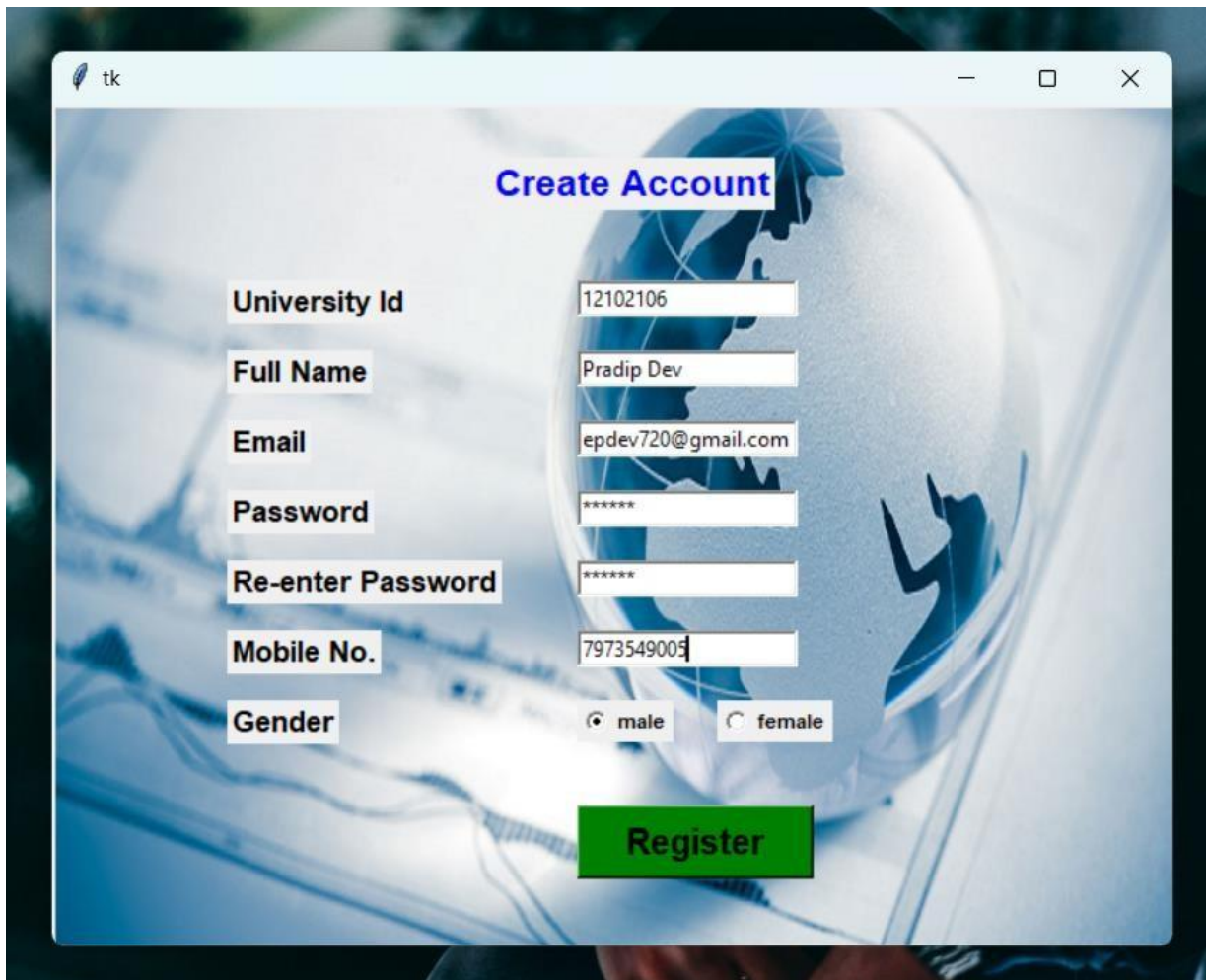
Data Flow Diagram show how data going to flow in the software.



Login Page



Signup Page

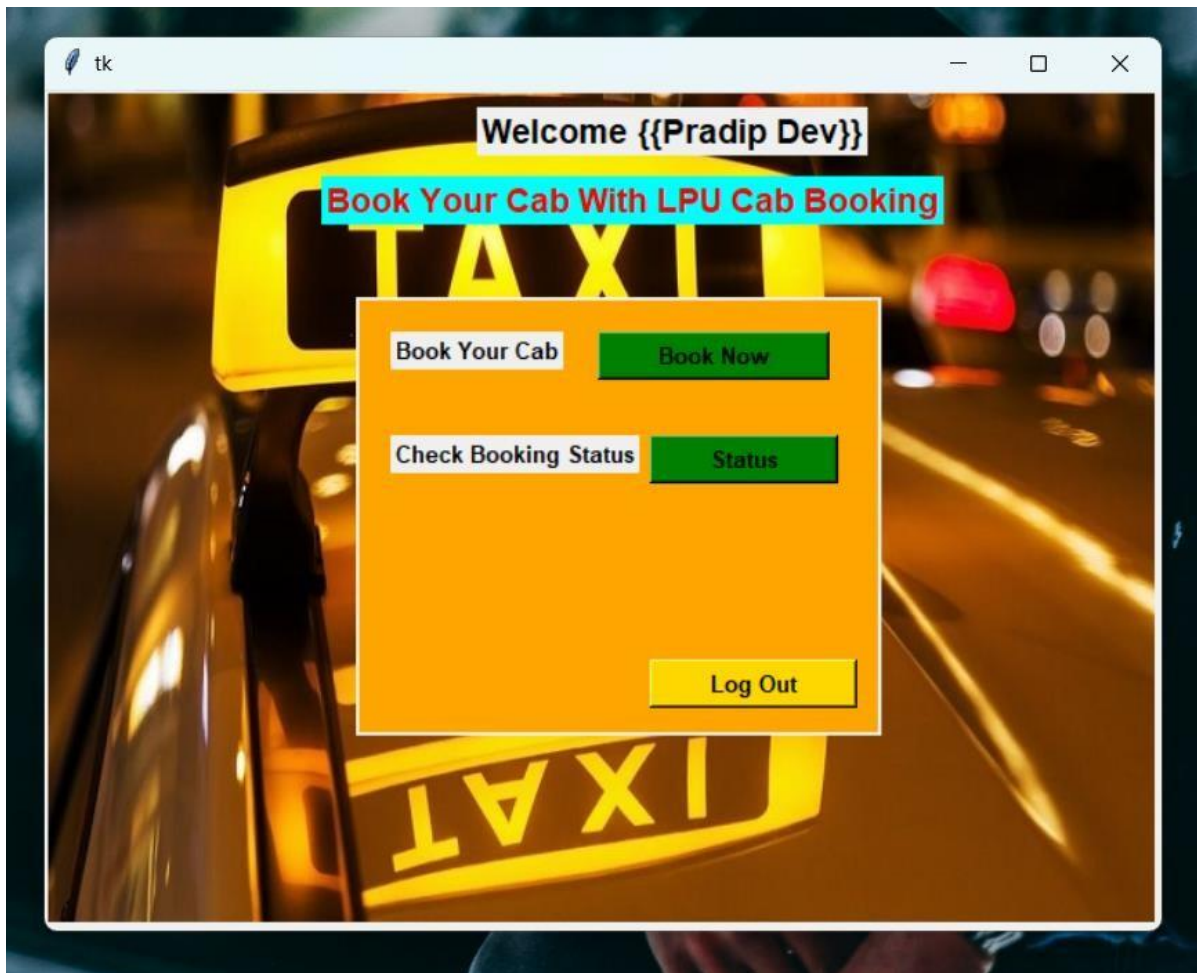


The image shows a screenshot of a Tkinter window titled 'tk' with standard window controls. The window contains a 'Create Account' form. The form has a background image of a globe. The form fields and their values are as follows:

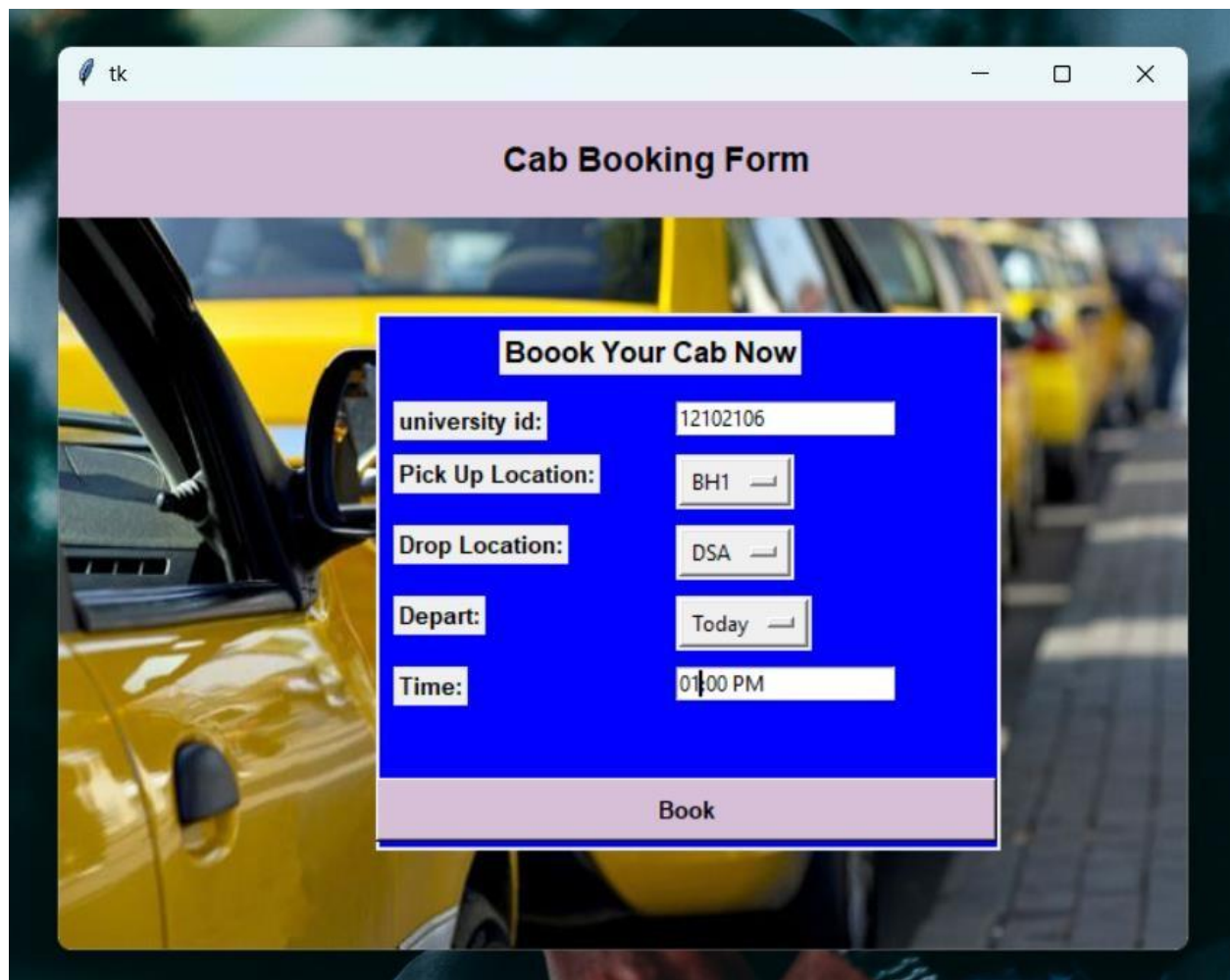
Field Label	Value
University Id	12102106
Full Name	Pradip Dev
Email	epdev720@gmail.com
Password	*****
Re-enter Password	*****
Mobile No.	7973549005
Gender	<input checked="" type="radio"/> male <input type="radio"/> female

At the bottom of the form is a green 'Register' button.

Booking page



Booking form



The image shows a screenshot of a Tkinter window titled "tk" with standard window controls. The window has a light purple header bar with the text "Cab Booking Form". The background of the window is a blurred image of yellow taxis. Overlaid on this is a blue rectangular form titled "Boook Your Cab Now" (note the typo). The form contains several input fields and dropdown menus:

- university id:** A text input field containing "12102106".
- Pick Up Location:** A dropdown menu showing "BH1".
- Drop Location:** A dropdown menu showing "DSA".
- Depart:** A dropdown menu showing "Today".
- Time:** A text input field containing "01:00 PM".

At the bottom of the blue form is a light purple bar with a "Book" button.

Admin pannel

The screenshot displays the Admin Panel of the LPU Cab Booking system, featuring three main windows:

- Customer List Window:** A table listing customer details.
- Booking Details Window:** A table showing individual booking records.
- Welcome Window:** A screen with a taxi background and navigation buttons.

Customer List Window Data:

Customer Id	Costumer Name	Email	Mobile
123	Neeraj	neeraj11@gmail.com	8755946301
11712314	neeraj	neeraj@gmail.com	8755946301
1	Resham	reshambharti145@gmail.com	1234567809
12115852	ASHISH KUMAR	ashishkumar267031@gmail.com	6201257288
125632	Kumar	kumar@gmail.com	4563217890
147852	Suraj	suraj@gmail.com	6201232651
12102180	sunil pat	abc@gmail.com	7901897670
12102106	Pradip Dev	pradeepdev720@gmail.com	7973549005

Booking Details Window Data:

Customer Id	pick up	Drop	depart	Time	status	Confirm
123	(CAMPUS CAFE)	BH7	Tomorrow	(1:00 AM)	Pending	Confirm
123	(UNI MALL)	BH4	Today	(12:00 PM)	Pending	Confirm
11712314	(CAMPUS CAFE)	APARTMENTS	Today	(12:00 PM)	Pending	Confirm
1	DSA	GH6	Today	(12:00 PM)	Pending	Confirm
1	DSA	GH6	Today	(12:00 PM)	Pending	Confirm
1	DSA	GH6	Today	(12:00 PM)	Pending	Confirm
1	DSA	GH6	Today	(12:00 PM)	Pending	Confirm
1	DSA	GH6	Today	(12:00 PM)	Pending	Confirm
1	DSA	GH6	Today	(12:00 PM)	Pending	Confirm
12115852	BH2	(CAMPUS CAFE)	Today	(12:00 PM)	Pending	Confirm
12115852	BH2	(CAMPUS CAFE)	Today	(12:00 PM)	Pending	Confirm
12115863	BH4	(UNI MALL)	Today	(12:00 PM)	Pending	Confirm
12115852	BH2	(CAMPUS CAFE)	Today	(12:00 PM)	Pending	Confirm
12102180	BH2	BH6	Today	2:00pm	Pending	Confirm
12102180	BH2	DSA	Today	(12:00 PM)	Pending	Confirm
12102106	BH1	(CAMPUS CAFE)	Today	(12:00 PM)	Pending	Confirm
12102106	BH1	DSA	Today	(01:00 PM)	Pending	Confirm
12102106	BH1	(CAMPUS CAFE)	Today	(12:00 PM)	Pending	Confirm

Welcome Window: Displays "Welcome to LPU Cab Booking system" with a "TAXI" sign. It includes buttons for "Customer Detail", "Customer Booking", and "Log Out".

References

<https://www.geeksforgeeks.org/python-projects-beginner-to-advanced/>

Coding parts

```
main.py x
D:\Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...
1 from tkinter import *
2 import sqlite3
3 from tkinter import messagebox
4 import re
5 class Login:
6     def __init__(self,root):
7         #Variable declaration
8         self.lname=StringVar()
9         self.lpass=StringVar()
10        self.root=root
11
12        self.root.geometry("640x480")
13        self.photo=PhotoImage(file="lpu.png")
14        self.label2=Label(self.root, image=self.photo).place(x=0,y=0)
15        self.label = Label(self.root, text="Welcome to LPU Cab Booking system", font="Helvetica 14 bold ",).place(x=180,y=10)
16        self.canvas=Canvas(self.root,bg="pink",height=250, width=300).place(x=180,y=120)
17        self.label=Label(self.root,text="Log in into your account",font="Helvetica 10 bold").place(x=250,y=130)
18        self.login=Label(self.root,text="Username",font="Helvetica 10 bold").place(x=210,y=180)
19        self.entry1=Entry(self.root,textvar=self.lname).place(x=290,y=180)
20        self.login1 = Label(self.root, text="Password", font="Helvetica 10 bold").place(x=210, y=220)
21        self.entry2 = Entry(self.root, textvar=self.lpass,show="").place(x=290, y=220)
22        self.button1 = Button(self.root, text="login", font="Helvetica 12 bold", bg="blue",padx=30,command=self.login_access1).place(x=280,y=280)
23        self.asking=Label(self.root,text="You have a account?",font="Helvetica 8 bold").place(x=210,y=330)
24        self.signup_button=Button(self.root,text="sign up",font="Helvetica 9 bold",bg="red",command=self.rg_page).place(x=330,y=330)
25
26        #calling function after clicking on sign up button
27        def rg_page(self):
28            top=Toplevel(self.root)
29            obj=registration(top)
30        def login_access1(self):
31            #fetching data
32            self.login_name = self.lname.get()
33            self.login_pass = self.lpass.get()
34
35            if (self.login_name == "pradeep" and self.login_pass == "password"):
36                top1=Toplevel(self.root)
37                obj1=admin_page(top1)
```

```
main.py x
D:\Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...
35 if (self.login_name == "pradeep" and self.login_pass == "password"):
36     top1=Toplevel(self.root)
37     obj1=admin_page(top1)
38 elif (self.login_name==' ' and self.login_pass==' '):
39     messagebox.showinfo("warning", "enter username password")
40 else:
41     self.Costumer_login()
42 def Costumer_login(self):
43     self.id = self.lname.get()
44     self.id_pass = self.lpass.get()
45     con = sqlite3.connect("Cab booking.db")
46     cursor=con.cursor()
47     self.find_user = ("select * from costumer_detail where uid=? and password=? ")
48     cursor.execute(self.find_user,[(self.lname.get()),(self.lpass.get())])
49     self.results=cursor.fetchall()
50     if self.results:
51         self.top3=Toplevel(self.root)
52         self.top3.geometry("640x480")
53         self.photo_c = PhotoImage(file="Taxi-Booking.png")
54         self.label_c = Label(self.top3, image=self.photo_c).place(x=0, y=0)
55         con = sqlite3.connect("Cab booking.db")
56         self.find_user1 = ("select full_name from costumer_detail where uid=?")
57         cursor = con.cursor()
58         cursor.execute(self.find_user1, [(self.lname.get())])
59         self.results = cursor.fetchall()
60         self.label_c1=Label(self.top3,text="Welcome", font="Helvetica 14 bold ").place(x=250,y=10)
61         self.label_c2 = Label(self.top3, text=self.results, font="Helvetica 14 bold ").place(x=340, y=10)
62         self.label_c3 = Label(self.top3, text="Book Your Cab with LPU Cab Booking", font="Helvetica 14 bold ",fg="red",bg="cyan").place(x=160, y=50)
63         self.canvas_c = Canvas(self.top3, bg="orange", height=250, width=300).place(x=180, y=120)
64         self.label_c4 = Label(self.top3, text="Book Your Cab", font="Helvetica 10 bold ").place(x=200, y=140)
65         self.button_c = Button(self.top3, text="Book Now", font="Helvetica 10 bold", bg="green", padx=30,command=self.Book_now).place(x=320, y=140)
66         self.label_c5 = Label(self.top3, text="Check Booking Status", font="Helvetica 10 bold ").place(x=200, y=200)
67         self.button_c1= Button(self.top3, text="Status", font="Helvetica 10 bold", bg="green", padx=30,command=self.check_status).place(x=350, y=200)
68         self.button_logout=Button(self.top3,text="log Out",font="Helvetica 10 bold", bg="gold", padx=30,command=self.gotoback).place(x=350, y=330)
69     else:
70         messagebox.showinfo("warning", "Check Username & Password")
71     con.commit()
```

```
File Edit Selection View Go Run Terminal Help
main.py - Visual Studio Code

main.py x
D:\> Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...

69         else:
70             messagebox.showinfo("Warning", "Check Username & Password")
71         con.commit()
72         con.close()
73     def check_status(self):
74         self.uid=int(self.lname.get())
75         con = sqlite3.connect("Cab booking.db")
76         self.find_user1 = ("select status from Booking1 where uid=?")
77         cursur = con.cursor()
78         cursur.execute(self.find_user1, [(self.uid)])
79         self.results = cursur.fetchall()
80         if(self.results):
81             messagebox.showinfo("Booking Status",self.results[0])
82         con.commit()
83         con.close()
84     def gotoback(self):
85         self.master=toplevel(self.top3)
86         login(self.master)
87     def Book_now(self):
88         self.top3=toplevel(self.root)
89         Book(self.top3)
90
91     class registration:
92     def __init__(self,top):
93         self.uid=IntVar()
94         self.fname=StringVar()
95         self.Email=StringVar()
96         self.pswd=StringVar()
97         self.rpswd=StringVar()
98         self.mob=IntVar()
99         self.gender=StringVar()
100         self.master=top
101         self.master.geometry("640x480")
102         self.photo1 = PhotoImage(file="ibsm-hero-1200.png")
103         self.label_1 = Label(self.master, image=self.photo1).place(x=0, y=0)
104         self.create=Label(self.master,text="Create Account",font="Helvetica 16 bold",fg="blue").place(x=250,y=30)
105         self.usrn=Label(self.master,text="University Id",font="Helvetica 13 bold").place(x=100,y=100)
```

```
File Edit Selection View Go Run Terminal Help
main.py - Visual Studio Code

main.py x
D:\> Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...

103         self.label_1 = Label(self.master, image=self.photo1).place(x=0, y=0)
104         self.create=Label(self.master,text="Create Account",font="Helvetica 16 bold",fg="blue").place(x=250,y=30)
105         self.usrn=Label(self.master,text="University Id",font="Helvetica 13 bold").place(x=100,y=100)
106         self.entry3=Entry(self.master,bd=2,textvar=self.uid).place(x=300,y=100)
107         self.usrn1 = Label(self.master, text="Full Name", font="Helvetica 13 bold").place(x=100, y=140)
108         self.entry4 = Entry(self.master, bd=2,textvar=self.fname).place(x=300, y=140)
109         self.usrn5 = Label(self.master, text="Email", font="Helvetica 13 bold").place(x=100, y=180)
110         self.entry5 = Entry(self.master, bd=2,textvar=self.Email).place(x=300, y=180)
111         self.usrn2 = Label(self.master, text="Password", font="Helvetica 13 bold").place(x=100, y=220)
112         self.entry6 = Entry(self.master, bd=2,textvar=self.pswd,show="*").place(x=300, y=220)
113         self.usrn3 = Label(self.master, text="Re-enter Password", font="Helvetica 13 bold").place(x=100, y=260)
114         self.entry7 = Entry(self.master, bd=2,textvar=self.rpswd,show="*").place(x=300, y=260)
115         self.usrn4 = Label(self.master, text="Mobile No.", font="Helvetica 13 bold").place(x=100, y=300)
116         self.entry8 = Entry(self.master, bd=2,textvar=self.Mob).place(x=300, y=300)
117         self.usrn = Label(self.master, text="Gender", font="Helvetica 13 bold").place(x=100, y=340)
118         self.radio=Radiobutton(self.master,text="male",value="male",variable=self.gender,font="Helvetica 8 bold").place(x=300,y=340)
119         self.radio=Radiobutton(self.master,text="female",value="female",variable=self.gender,font="Helvetica 8 bold").place(x=380,y=340)
120         self.button_reg=Button(self.master,text="Register",font="Helvetica 15 bold",padx=20,bg="green",command=self.database_registration).place(x=300,y=400)
121         self.master.mainloop()
122     def database_registration(self):
123         #variable declaration
124         self.Uid=self.uid.get()
125         self.fname=self.fname.get()
126         self.email=self.Email.get()
127         self.pasw=self.pswd.get()
128         self.rpasw=self.rpswd.get()
129         self.mob=self.Mob.get()
130         self.gnd=self.gender.get()
131
132         if(self.Uid==" " or self.fname==" " or self.email==" " or self.pasw==" " or self.rpasw==" " or self.mob==" "):
133             messagebox.showinfo("Empty", "Section cannot be empty!")
134         elif(self.pasw!=self.rpasw):
135             messagebox.showerror("error","make sure that password nd re-enter password must be same!")
136         elif(len(self.pasw)<6):
137             messagebox.showerror("password", "password should have atleast 6 character!")
138         elif(len(str(self.mob))<10):
139             messagebox.showerror("Mobile number", "Mobile number should be atleast ten digit!")
```

```
File Edit Selection View Go Run Terminal Help main.py - Visual Studio Code
D: > Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...
138 elif(len(str((self.mob)))!=10):
139     messagebox.showerror("Mobile number", "Mobile number should be atleast ten digit")
140 else:
141     if(re.match("(A(?P<name>[a-zA-Z_]+)@(?P<domain>[a-zA-Z_]+)(?P<toplevel>[a-zA-Z]+)",self.email,re.IGNORECASE)):
142         con=sqlite3.connect("Cab booking.db")
143         #con.execute("create table costumer_detail(uid int primary key, email varchar(30) unique ,full_name varchar(15) not null,"
144         #            #password varchar(30),mobile int not null,gender varchar(10) not null)")
145         con.execute("insert into costumer_detail(uid , email,full_name,password,mobile,gender)values(?,?,?,?,?,?)",( self.uid,self.email,self.fname,self.pasw,self
146         con.commit()
147         con.close()
148         answer=messagebox.askokcancel("Welcome","You have succesfully created your account")
149         if(answer):
150             ob_1=login(root)
151     else:
152         messagebox.showerror("Email", "Please enter valid Email Address")
153
154 class Book():
155     def __init__(self,top4):
156         #variable declaration
157         self.time = StringVar()
158         self.pickup = StringVar()
159         self.drop1 = StringVar()
160         self.dept = StringVar()
161         self.unid=StringVar()
162         self.top4 = top4
163         self.top4.geometry("640x480")
164         self.photo5 = PhotoImage(file="texi.png")
165         self.Book_photo = Label(self.top4, image=self.photo5).pack()
166         self.label_booking=Label(self.top4,text="Cab Booking form", font="Helvetica 14 bold",padx=250,pady=20 ,bg="Thistle").place(x=0,y=0)
167         self.canvas_book = Canvas(self.top4, bg="blue", height=300, width=350).place(x=180, y=120)
168         self.header = Label(self.top4, text="Book Your Cab Now", font="Helvetica 12 bold").place(x=250, y=130)
169         self.label_user=Label(self.top4,text="university id:",font="Helvetica 10 bold").place(x=190,y=170)
170         self.entry_user=Entry(self.top4,textvar=self.unid).place(x=350,y=170)
171         self.pickup = Label(self.top4, text="Pick up location:", font="Helvetica 10 bold").place(x=190, y=200)
172         #list for pick up and drop
173         list1 = ['BH1','BH2','BH3','BH4','BH5','BH6','BH7','GH1','GH2','GH3','GH4','GH5','GH6','CAMPUS CAFE','UNI HOSPITAL','UNI MALL','DSA','LAW BLOCK','ADMISSION BLOCK']
174         droplist = OptionMenu(self.top4, self.pickup, *list1)
175         self.pickup.set("Current location") # hydeFault text
```

```
File Edit Selection View Go Run Terminal Help main.py - Visual Studio Code
D: > Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...
173 droplist = OptionMenu(self.top4, self.pickup, *list1)
174 self.pickup.set("Current location") # hydeFault text
175 droplist.place(x=350, y=200)
176
177 self.drop =Label(self.top4, text="Drop Location:", font="Helvetica 10 bold").place(x=190, y=240)
178
179 #droplist for drop
180 droplist = OptionMenu(self.top4, self.drop1, *list1)
181 self.drop1.set("Drop location") # hydeFault text
182 droplist.place(x=350, y=240)
183 self.deprt = Label(self.top4, text="Depart:", font="Helvetica 10 bold").place(x=190, y=280)
184
185 list2 = ['Today', 'Tomorrow', 'Day After Tomorrow'] # list data
186 droplist = OptionMenu(self.top4, self.deprt, *list2)
187 self.deprt.set('Today') # hydeFault text
188 droplist.place(x=350, y=280)
189
190 self.time1 = Label(self.top4, text="Time:", font="Helvetica 10 bold").place(x=190, y=320)
191 self.time.set("12:00 PM")
192 self.entry_time=Entry(self.top4,textvar=self.time).place(x=350,y=320)
193
194 self.book1 = Button(self.top4, text="Book", font="Helvetica 10 bold",bg="Thistle",padx=155,pady=5,command=self.database_book).place(x=180, y=383)
195 self.top4.mainloop()
196 def database_book(self):
197     self.unid=self.unid.get()
198     self.pick=self.pick.get()
199     self.drop2=self.drop1.get()
200     self.time1=self.time.get()
201     self.depart=self.deprt.get()
202     self.status="Pending"
203     if (self.unid=='' or self.time1=='' or self.pick == 'Current location' or self.drop2 == 'Drop location'):
204         messagebox.showinfo("Empty", "Section cannot be empty!")
205     else:
206         con=sqlite3.connect("Cab booking.db")
207         con.execute("create table Booking1(unid varchar(20) not null,pickup varchar(20) not null ,drop_p varchar(25) not null,time_p varchar(10) not null,depart varc
208         con.execute("insert into Booking1(unid,pickup,drop_p,time_p,depart,status)values(?,?,?,?,?)",(self.unid,self.pick,self.drop2,self.time1,self.depart,self.s
209         messagebox.showinfo("Successful", "Booking Processing!")
```


The screenshot displays a Windows 11 desktop environment. The primary application is Visual Studio Code, which is open to a Python file named `main.py`. The code is a Tkinter-based application for a cab booking system, written in Hindi. It features a main window titled "main.py" with a grid layout. The interface includes buttons for "Login", "Admin", "Customer Detail", and "Logout". The "Admin" section has buttons for "Click Here", "Click Here", "Click Here", and "Click Here". The "Customer Detail" section has buttons for "Click Here", "Click Here", "Click Here", and "Click Here". The "Logout" button is labeled "Log out". The code uses the `sqlite3` module to connect to a database named "cab_booking.db". The application is running, and the window title bar shows "Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.10.6 64-bit". The taskbar at the bottom shows the Windows Start button, a search bar, and several pinned applications including File Explorer, Edge, and VS Code. The system tray shows the date and time as 11:56 AM on 11/18/2022.

```
File Edit View Go Run Terminal Help
main.py x
D:\> Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...
200 messagebox.showInfo("Successful", "Booking Processing!")
201
202 c=con.execute("select * from Booking1")
203
204 for i in c:
205     print(i)
206     con1.commit()
207     con1.close()
208
209 class admin_page:
210
211     def __init__(self,top1):
212         self.top1=top1
213         self.top1.geometry("640x480")
214         self.photo2 = PhotoImage(file="Taxi-Booking.png")
215         self.admin_photo = Label(self.top1, image=self.photo2).place(x=0, y=0)
216         self.admin1 = Label(self.top1, text="Welcome to LPU Cab Booking system", font="Helvetica 14 bold ", ).place(x=180, y=10)
217         self.canvas_admin = Canvas(self.top1, bg="purple", height=250, width=300).place(x=180, y=120)
218         self.admin2 = Label(self.top1, text="Customer Detail", font="Helvetica 10 bold").place(x=200, y=150)
219         self.admin_button=Button(self.top1,text="Click Here",bg='blue',font="Helvetica 9 bold",command=self.customer_detail).place(x=400,y=150)
220         self.admin2 = Label(self.top1, text="Customer Booking", font="Helvetica 10 bold").place(x=200, y=200)
221         self.admin_button1 = Button(self.top1, text="Click Here", bg='blue', font="Helvetica 9 bold",command=self.customer_booking).place(x=400, y=200)
222         self.logout_button = Button(self.top1, text="Log out", font="Helvetica 9 bold", bg="red",command=self.login).place(x=330, y=330)
223         self.top1.mainloop()
224
225     def login(self):
226         login(self.top1)
227     def customer_detail(self):
228         self.top2=Toplevel( self.top1)
229         self.top2.geometry("640x480")
230         self.photo_cd = PhotoImage(file="mario.png")
231         self.cd_photo = Label(self.top2, image=self.photo_cd).place(x=0, y=0)
232         #self.c_header=Label(self.top2,text="Customer details",font="Helvetica 14 bold").place(x=250,y=10)
233         self.c_id=Label(self.top2,text="Customer Id",font="Helvetica 10 bold").grid(row=1,column=1)
234         self.c_name=Label(self.top2,text="Customer Name",font="Helvetica 10 bold").grid(row=1,column=2)
235         self.c_email = Label(self.top2, text="Email", font="Helvetica 10 bold").grid(row=1, column=3)
236         self.c_Mobile = Label(self.top2, text="Mobile", font="Helvetica 10 bold").grid(row=1, column=4)
237         con = sqlite3.connect("Cab booking.db")
238         c=con.execute("select * from customer_detail")
239         self.j=0
240         for i in c:
```

```

File Edit Selection View Go Run Terminal Help
main.py - Visual Studio Code

main.py x
D:\> Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...

245     for i in c:
246         self.entry_d=Label(self.top2,text=i[0]).grid(row=self.j+2,column=1)
247         self.j=self.j+1
248     c1 = con.execute("select * from costumer_detail")
249     self.a = 0
250     for j in c1:
251         self.name = Label(self.top2, text=j[2]).grid(row=self.a + 2, column=2)
252         self.a = self.a + 1
253     c2 = con.execute("select * from costumer_detail")
254     self.d = 0
255     for k in c2:
256         self.name = Label(self.top2, text=k[1]).grid(row=self.d + 2, column=3)
257         self.d = self.d + 1
258     c3 = con.execute("select * from costumer_detail")
259     self.l = 0
260     for p in c3:
261         self.name = Label(self.top2, text=p[4]).grid(row=self.l + 2, column=4)
262         self.l = self.l + 1
263
264     con.commit()
265     con.close()
266 def costumer_booking(self):
267     self.bookin_top = Toplevel(self.top1)
268     self.bookin_top.geometry("640x480")
269     con = sqlite3.connect("cab_booking.db")
270     self.booking_id = Label(self.bookin_top, text="costumer id", font="Helvetica 10 bold").grid(row=1, column=1)
271     self.booking_pick = Label(self.bookin_top, text="pick up", font="Helvetica 10 bold").grid(row=1, column=2)
272     self.booking_drop = Label(self.bookin_top, text="Drop", font="Helvetica 10 bold").grid(row=1, column=3)
273     self.booking_depart = Label(self.bookin_top, text="depart", font="Helvetica 10 bold").grid(row=1, column=4)
274     self.booking_time = Label(self.bookin_top, text="time", font="Helvetica 10 bold").grid(row=1, column=5)
275     self.booking_status = Label(self.bookin_top, text="status", font="Helvetica 10 bold").grid(row=1, column=6)
276     self.booking_confirm = Label(self.bookin_top, text="confirm", font="Helvetica 10 bold").grid(row=1, column=7)
277
278     self.booking_info=con.execute("select unid from Booking1")
279     self.g=0
280     for i in self.booking_info:
281         self.booking_id = Label(self.bookin_top, text=i,fg='red').grid(row=self.g + 2, column=1)

```

Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.10.6 64-bit 11/08/2022

```
File Edit Selection View Go Run Terminal Help
main.py - Visual Studio Code

main.py x
D:\> Python Project > CAB-BOOKING-SYSTEM-main > CAB-BOOKING-SYSTEM-main > main.py > ...

284
285 self.booking_pickup = con.execute("select pickup from Booking1")
286 self.w=0
287 for j in self.booking_pickup:
288     self.booking_pickup = Label(self.bookin_top, text=j, fg='red').grid(row=self.w + 2, column=2)
289     self.w = self.w + 1
290
291 self.booking_drop = con.execute("select drop_p from Booking1")
292 self.e = 0
293 for j in self.booking_drop:
294     self.booking_drop = Label(self.bookin_top, text=j, fg='red').grid(row=self.e + 2, column=3)
295     self.e = self.e + 1
296
297 self.booking_depart = con.execute("select depart from Booking1")
298 self.r = 0
299 for j in self.booking_depart:
300     self.booking_depart = Label(self.bookin_top, text=j, fg='red').grid(row=self.r + 2, column=4)
301     self.r = self.r + 1
302
303 self.booking_time = con.execute("select Time_p from Booking1")
304 self.t = 0
305 for j in self.booking_time:
306     self.booking_time = Label(self.bookin_top, text=j, fg='red').grid(row=self.t + 2, column=5)
307     self.t = self.t + 1
308
309 self.booking_status = con.execute("select status from Booking1")
310 self.y = 0
311 for j in self.booking_status:
312     self.booking_status = Label(self.bookin_top, text=j, fg='red').grid(row=self.y + 2, column=6)
313     self.y = self.y + 1
314 con.commit()
315 con.close()
316
317
318 root=tk()
319 login(root)
320 root.mainloop()
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.10.6 64-bit 1:56 AM 11/18/2022

