

```
1 ###
2 ##
3 ##
4 ###
5
6 from tkinter import *
7 import sys
8 import tkinter.messagebox as Msgbox
9 import os, sqlite3
10
11 try:
12     import copy_b
13     print("you didnt run the first program")
14 except:
15     print("database already initialized - thank you for
        following rules.")
16
17 com = sqlite3.connect("muta.db")
18 cur = com.cursor()
19
20 ##
21 ##
22 #fn for search,register,commit,close,reset,remove,in,out,
    exit
23 def search_db(): #search is based on id, or name.
24     id_s = int(idno_t.get())
25     name_s = str(name_t.get())
26     if (id_s>10000):
27         cur.execute("select id,name,room_id,comments,grade
            from customer where id = {0}".format(id_s))
28     else:
29         cur.execute('select id,name,room_id,comments,grade
            from customer where name = "{0}"'.format(name_s))
30
31     data_set = cur.fetchone()
32     idno_t.set(data_set[0])
33     name_t.set(data_set[1])
34     roomno_t.set(data_set[2])
35     comment_t.set(data_set[3])
36     grade_t.set(data_set[4])
37
38
39
40 def add_db():
41     #try:
42         id_s = int(idno_t.get())
```

```

43         name_s = name_t.get()
44         roomno_s = int(roomno_t.get())
45         comment_s = comment_t.get()
46         grade_s = grade_t.get()
47         cur.execute('insert into customer (id, name,
room_id, comments, grade) values ({0},"{1}", {2}, "{3
}", "{4}")'.format(id_s,name_s,roomno_s,comment_s,grade_s
))
48         com.commit()
49
50         MsgBox.showinfo("info","Customer Data Added!")
51
52         idno_t.set(0)
53         name_t.set("")
54         roomno_t.set("")
55         comment_t.set("")
56         grade_t.set("")
57     # except:
58     #     MsgBox.showinfo("Error","Something Wrong, Please
check data or call admin")
59
60
61 def commit_db():
62     com.commit()
63     MsgBox.showinfo("info","Committed to Database")
64
65 def reset():
66     idno_t.set(0)
67     name_t.set("")
68     roomno_t.set("")
69     comment_t.set("")
70     grade_t.set("")
71
72 def remove_db():
73     # for now, only remove by id_no
74     print()
75
76 def quit_B():
77     MsgBox.showinfo("Thankyou","Thankyou for using the
software, hit, ok to quit.")
78     master.destroy()
79     sys.exit()
80
81
82 # master window
83 master = Tk()

```

```
84 master.title("Grand Renoir Hotel (GUI) v0.1")
85 master.minsize(870,540)
86
87 # global variable declare and assign
88 global idno_t
89 global name_t
90 global roomno_t
91 global comment_t
92 global grade_t
93
94 idno_t = IntVar()
95 name_t = StringVar()
96 roomno_t = StringVar()
97 comment_t = StringVar()
98 grade_t = StringVar()
99
100
101 # ALL FRAMES HERE
102 topframe = Frame(master)
103 topframe.grid(row=0,column=0)
104 leftframe = Frame(master)
105 leftframe.grid(row=4,column=0)
106 rightframe = Frame(master)
107 rightframe.grid(row=4,column=20)
108 bottomframe = Frame(master)
109 bottomframe.grid()
110
111
112 # Topframe widgets
113 maroon = "#800000"
114 renoir = PhotoImage(file="logo.png")
115 renoir_logo = Label(topframe,image=renoir)
116 renoir_logo.grid(row=1,column=0)
117 main_text = Label(topframe,text="GRAND RENOIR HOTEL EST.
    1978",fg=maroon)
118 main_text.grid(row=0,column=4) #need additional ipad
119 reservation_text = Label(topframe,text="CUSTOMER
    RESERVATION SOFTWARE",fg=maroon)
120 reservation_text.grid(row=1,column=4,padx=50)
121 dev_text = Label(topframe,text="by. Sanjana Dubey")
122 dev_text.grid(row=1,column=20,padx=100)
123
124 # leftframe widgets button
125 search = Button (leftframe,text=" SEARCH ", command =
    search_db)
126 search.grid(row=4,column=1,padx=20)
```

```
127 register = Button (leftframe,text=" REGISTER ", command
    = add_db)
128 register.grid(row=6,column=1)
129 commit = Button (leftframe,text=" COMMIT TO DB ", command
    = commit_db)
130 commit.grid(row=8,column=1)
131 # close = Button (leftframe,text=" CLOSE ", command =
    leave)
132 # close.grid(row=10,column=1)
133 reset = Button (leftframe,text=" RESET ", command = reset
    )
134 reset.grid(row=12,column=1)
135 remove = Button (leftframe,text=" REMOVE ", command =
    remove_db)
136 remove.grid(row=14,column=1)
137 in_b = Button (leftframe,text=" IN ") # cannot be
    implemented yet !
138 in_b.grid(row=16,column=1)
139 out_b = Button (leftframe,text=" OUT ") # cannot be
    implemented yet !
140 out_b.grid(row=16,column=2)
141 exit_b = Button (leftframe,text=" EXIT ",command=quit_B)
142 exit_b.grid(row=18,column=1)
143
144 # rightframe widgets
145 idno = Label(rightframe,text="ID NO")
146 idno_e = Entry(rightframe, textvariable = idno_t)
147 name = Label(rightframe,text="NAME")
148 name_e = Entry(rightframe, textvariable = name_t)
149 roomno = Label(rightframe,text="ROOM NO.")
150 roomno_e = Entry(rightframe, textvariable = roomno_t)
151 comment = Label(rightframe,text="COMMENTS & \n
    RESERVATION INFO")
152 comment_e = Entry(rightframe, textvariable = comment_t)
153 grade = Label(rightframe,text="CUSTOMER GRADE")
154 grade_e = Entry(rightframe, textvariable = grade_t)
155
156 idno.grid(row=4,column=20)
157 idno_e.grid(row=4,column=25)
158 name.grid(row=6,column=20)
159 name_e.grid(row=6,column=25)
160 roomno.grid(row=8,column=20)
161 roomno_e.grid(row=8,column=25)
162 comment.grid(row=10,column=20)
163 comment_e.grid(row=10,column=25)
164 grade.grid(row=12,column=25)
```

File - G:\PG\sanjana\grand_renoir\sanjana_GrandRenoirHotel.py

```
165 grade_e.grid(row=12,column=30)
166
167
168
169
170
171 # event trigger
172 mainloop()
173
```