```
1
      LTBRARTES------
  2
  3
      from tkinter import *
      import tkinter.font as font
  4
      from PIL import Image, ImageTk
  5
  6
      import sqlite3
  7
      from tkinter import messagebox
  8
      import serial
  9
      import bluetooth
      import sys
 10
 11
      import os
 12
 13
      #for converting to windows software
 14
      def resource_path(relative_path):
          """ Get absolute path to resource, works for dev and for PyInstaller """
 15
 16
          try:
              # PyInstaller creates a temp folder and stores path in _MEIPASS
 17
 18
              base_path = sys._MEIPASS2
 19
          except Exception:
 20
              base_path = os.path.abspath(".")
 21
 22
          return os.path.join(base_path, relative_path)
 23
 24
 25
      26
 27
      #saves the registered values into the database
 28
      def register_user():
 29
          name_info=name.get()
 30
          age_info=age.get()
 31
          conn_type_of_devices_info=conn_type_of_devices.get()
 32
          conn_type_of_devices_info_1=conn_type_of_devices_1.get()
          gender_info=gender.get()
 33
          username_info=username.get()
 34
 35
          password_info= password.get()
 36
 37
          #software.db contains registered info
 38
          conn= sqlite3.connect(resource_path("software.db"))
 39
          c=conn.cursor()
          c.execute("INSERT INTO person
 40
          VALUES('"+name_info+"',"+age_info+",'"+conn_type_of_devices_info+"','"+conn_type_o J
          f_devices_info_1+"','"+gender_info+"','"+username_info+"','"+password_info+"')")
 41
          messagebox.showinfo("Information", "Your record is saved!")
 42
          conn.commit()
 43
          conn.close()
 44
          conn= sqlite3.connect(resource_path("user_info.db"))
1 -
```

```
45
         c=conn.cursor()
46
         c.execute("INSERT INTO Curr_session(name) VALUES(("+name_info+"')")
47
         conn.commit()
48
         conn.close()
49
50
         name_entry.delete(0,END)
51
         age_entry.delete(0,END)
52
         conn_devices_entry.deselect()
53
         conn_devices_entry1.deselect()
         gen_entry_m.deselect()
54
         gen_entry_f.select()
55
56
         username_entry.delete(0,END)
57
         password_entry.delete(0,END)
58
59
         Label(screen1, text="Registeration is Successful", fg="green",
                                                                                                  \overline{\downarrow}
         font=("Calibri", 11)).pack()
60
61
62
     #these functions delete the screen
63
     def delete_s1():
64
         screen1.destroy()
65
66
     #for the registration page
67
     def register():
68
         global screen1
69
         screen1= Toplevel(screen)
         screen1.title("Register")
70
         screen1.geometry("400x650")
71
         screen1.configure(bg="#FFFEF2")
72
73
74
         global name
75
         global age
76
         global conn_type_of_devices
77
         global conn_type_of_devices_1
78
         global gender
79
         global username
80
         global password
81
82
         global name_entry
83
         global age_entry
84
         global conn_devices_entry
         global conn_devices_entry1
85
86
         global gen_entry_m
87
         global gen_entry_f
88
         global username_entry
89
         global password_entry
90
91
         name=StringVar()
```

```
92
           age=StringVar()
           conn_type_of_devices=StringVar()
 93
 94
           conn_type_of_devices_1=StringVar()
 95
           gender=StringVar()
 96
           username= StringVar()
 97
           password= StringVar()
 98
           Label(screen1, text="Please Enter details below
 99
                                                                                                   \Box
           ",width="400",height="3",bg="#caf0f8", font=("Times",16)).pack()
100
           Label(screen1, text="", bg="#FFFEF2").pack()
101
102
103
           Label(screen1, text="Enter Name", font=("Times", 13), bg="#FFFEF2").pack()
104
105
           name_entry= Entry(screen1, textvariable=name, width=30)
106
           name_entry.pack()
107
108
           Label(screen1, text="Enter Age", font=("Times", 13), bg="#FFFEF2").pack()
109
110
           age_entry= Entry(screen1,textvariable=age, width=30)
111
           age_entry.pack()
112
113
           Label(screen1, text="Select types of
                                                                                                   \overline{\downarrow}
           connection", font=("Times", 13), bg="#FFFEF2").pack()
114
115
           conn_devices_entry= Checkbutton(screen1,
                                                                                                   Į
           text="Wifi", variable=conn_type_of_devices, bg="#FFFEF2")
116
           conn_devices_entry.deselect()
117
           conn_devices_entry.pack()
118
119
           conn_devices_entry1= Checkbutton(screen1,
                                                                                                   \Box
           text="Bluetooth", variable=conn_type_of_devices_1, bq="#FFFEF2")
120
           conn_devices_entry1.deselect()
121
           conn_devices_entry1.pack()
122
123
           Label(screen1, text="Select gender", font=("Times", 13), bq="#FFFEF2").pack()
124
125
           gen_entry_m= Radiobutton(screen1, text="Male",variable=gender,
                                                                                                   \Box
           value="Male", bg="#FFFEF2")
126
           gen_entry_m.deselect()
127
           gen_entry_m.pack()
128
129
           gen_entry_f= Radiobutton(screen1, text="Female", variable=gender,
                                                                                                   7
           value="Female", bg="#FFFEF2")
130
           gen_entry_f.select()
131
           gen_entry_f.pack()
132
           Label(screen1, text="", bg="#FFFEF2").pack()
133
3 -
```

```
134
           Label(screen1, text="Enter Username", font=("Times", 13), bg="#FFFEF2").pack()
 135
136
           username_entry= Entry(screen1,textvariable=username, width=30)
137
           username_entry.pack()
138
139
           Label(screen1, text="Enter Password", font=("Times", 13), bg="#FFFEF2").pack()
140
141
           password_entry= Entry(screen1, textvariable=password, width=30)
142
           password_entry.pack()
143
144
145
           Label(screen1, text="", bg="#FFFEF2").pack()
           Button(screen1, text="Register", bg="#d1ffea", height=2, width= 20, command=
146
                                                                                                \Box
           register_user).pack()
147
148
           Label(screen1, text="", bg="#FFFEF2").pack()
           Button(screen1, text="Direct to Login Page", bg="#d1ffea", height=2, width= 20,
149
                                                                                                7
           command= login).pack()
150
           Label(screen1, text="", bg="#FFFEF2").pack()
151
           Button(screen1, text="Close window", bg="#ffcccb", height=2, width= 20,
152
                                                                                                7
           command=delete_s1).pack()
153
154
155
                         156
157
       #these functions delete the screens
158
       def delete_s4():
159
           screen4.destroy()
160
       def delete_s5():
           screen5.destroy()
161
162
       def delete_s8():
163
           screen8.destroy()
164
       def delete_s9():
165
           screen9.destroy()
166
       \mathbf{1}\cdot\mathbf{1}\cdot\mathbf{1}
167
168
       #not fully complete
169
       #purpose- saving the bluetooth info to new database
170
171
       def save_info():
172
           conn= sqlite3.connect(resource_path("user_info.db"))
173
           c=conn.cursor()
174
           # c.execute("INSERT INTO Curr_session(bluetooth_name, bluetooth_address)
                                                                                                \Box
           VALUES('"+addr+"', '"+name+"')")
175
176
           c.execute("UPDATE Curr_session SET bluetooth_name='"+addr+"',
                                                                                                \overline{\downarrow}
4 -
```

```
bluetooth_address='"+name+"' WHERE name='"+username1+"'")
            # messagebox.showinfo("Information", "Your record is saved!")
 177
 178
            conn.commit()
 179
            conn.close()
 180
            conn= sqlite3.connect(resource_path("user_info.db"))
 181
            c=conn.cursor()
 182
            c.execute("SELECT * FROM Curr_session")
 183
            r=c.fetchall()
 184
            for i in r:
 185
                print(i)
            # messagebox.showinfo("Information", "Your record is saved!")
 186
 187
            conn.commit()
 188
            conn.close()
        1.1.1
 189
 190
 191
        #Scanning nearby bluetooth devices
 192
 193
        def button_clicked1():
 194
            # print("success")
 195
            # Label(screen9,text="hi").pack()
 196
 197
            #print("Scanning")
 198
            Label(screen9, text="Scanning", width="400", height="3",
                                                                                                     \overline{\downarrow}
            font=("Times", 13, "bold")).pack()
 199
  200
            nearby_devices = bluetooth.discover_devices(lookup_names=True)
 201
  202
            #print("Found {} devices.".format(len(nearby_devices)))
  203
            Label(screen9, text="Found {}
            devices.".format(len(nearby_devices)), width="400", height="3",
                                                                                                     Į
            font=("Times", 11, "bold")).pack()
  204
            for addr, name in nearby_devices:
  205
 206
                #print(" {} - {}".format(addr, name))
  207
                Label(screen9, text=" {} - {}".format(addr, name), width="400", height="3",
                                                                                                     \Box
                font=("Times", 12, "bold")).pack()
  208
  209
 210
        #Scan devices page
 211
 212
        def scan_devices():
 213
            global screen9
 214
            screen9=Toplevel(screen)
 215
            screen9.title("Scanning for nearby Bluetooth devices")
  216
            screen9.geometry("400x500")
 217
            Label(screen9,text="Nearby Bluetooth devices",font=("Times",16)).pack()
 218
            Label(screen9, text="").pack()
  219
            Button(screen9, text="Scan Devices", height=1, width= 30,
                                                                                                     \overline{\downarrow}
- 5 -
```

```
bg="#0077b6", font=("Times", 13, "bold"), command=button_clicked1).pack()
          # Button(screen9,text="Save", height=1, width= 30,
220
                                                                                                   \Box
          bg="#CD5C5C", font=("Times", 11, "bold"), command=save_info()).pack()
221
          #Label(screen9, text="hi").pack()
222
          #Label(screen9, text="").pack()
223
          Button(screen9, text="Exit", height=1, width= 30,
                                                                                                   \overline{\downarrow}
          bg="#CD5C5C", font=("Times", 13, "bold"), command=delete_s9).pack(side=BOTTOM)
224
225
226
227
      #Page for Dashboard
228
229
      def session():
230
          # delete_s()
231
          delete_s2()
232
          global screen8
233
          screen8= Toplevel(screen)
234
235
          screen8.title("Dashboard")
236
          screen8.geometry("400x400")
237
          Label(screen8, text="Welcome to the Dashboard", font=("Times", 16)).pack()
238
239
          Button(screen8, text="Sign_out", bg="#ffcccb", command=delete_s8).place(x=340, y=10)
240
241
          conn= sqlite3.connect(resource_path("software.db"))
242
          c1=conn.cursor()
243
          c1.execute("SELECT name, gender FROM person WHERE username='"+username1+""")
244
          r1=c1.fetchall()
245
246
          for i in r1:
247
               print(i[0])
               print(i[1])
248
249
               if(i[1]=="Female"):
250
                   Label(screen8, text="Hello! Ms. {}".format(i[0]), font=("Times", 14)).pack()
251
                   # print("Ms. {}".format(i[0]))
252
              else:
253
                   Label(screen8, text="Hello! Mr. {}".format(i[0]), font=("Times", 14)).pack()
                   # print("Mr. {}".format(i[0]))
254
255
256
          conn.commit()
257
          conn.close()
258
259
          Label(screen8, text="").pack()
260
          Button(screen8, text="Your Saved Devices", width=20, height=2, bg="#d1ffea").pack()
261
262
          Label(screen8, text="").pack()
263
          Button(screen8, text="Scan Nearby
                                                                                                   \Box
          Devices", width=20, height=2, bg="#d1ffea", command=scan_devices).pack()
```

```
264
265
266
                                                                                                 \Box
267
268
      #if login is successful
      def login_success():
269
270
          session()
271
272
      #if passwoord is not recognised
      def password_not_recognized():
273
274
          global screen4
275
          screen4 = Toplevel(screen)
          screen4.title("Failed")
276
277
          screen4.geometry("150x100")
          screen4.configure(bg="#ffcccb")
278
          Label(screen4, text="", bg="#ffcccb").pack()
279
          Label(screen4, text="Incorrect password", bg="#ffcccb", font=("Times", 13)).pack()
280
281
          Button(screen4, text="Try again", bg="#ff6863", command=delete_s4).pack()
282
283
      #to allow to go to registration page
284
      def direct_register():
285
          register()
286
          delete_s5()
287
288
      #if user is not found
289
      def user_not_found():
290
          global screen5
          screen5 = Toplevel(screen)
291
292
          screen5.title("Failed")
          screen5.geometry("150x150")
293
          screen5.configure(bg="#ffdcd1")
294
          Label(screen5, text="User not found!", bg="#ffdcd1", font=("Times", 13)).pack()
295
296
          Button(screen5, text="Try again", bg="#ff6863", command=delete_s5).pack()
          Label(screen5, text="", bg="#ffdcd1").pack()
297
          Button(screen5, text="Register new
298
                                                                                                 7
          user", bg="#d1ffea", command=direct_register).pack()
299
300
      #to check whether login credentials are correct or not
301
302
      def login_verify():
303
          global username1
304
          global password1
305
          username1=username_verify.get()
306
          password1=password_verify.get()
307
          usernamee_entry1.delete(0,END)
308
          password_entry1.delete(0,END)
309
```

7 -

```
310
            conn= sqlite3.connect(resource_path("software.db"))
  311
            c=conn.cursor()
  312
  313
            c.execute("SELECT username, password FROM person")
  314
            r=c.fetchall()
 315
  316
            user_list=[]
 317
            pass_list=[]
 318
            for i in r:
                user_list.append(i[0])
  319
                pass_list.append(i[1])
 320
 321
            print(user_list)
            print(pass_list)
  322
            for i in range(len(user_list)):
 323
 324
                if(username1 in user_list):
  325
                    if(password1 in pass_list):
                         idx=user_list.index(username1)
 326
                         if(pass_list[idx]==password1):
 327
  328
                             # print("YES")
                             login_success()
 329
                             break
 330
 331
                         else:
 332
                             # print("Incorrect Password")
                             messagebox.askretrycancel("Try Again", "Password Incorrect")
 333
  334
                             break
  335
                    else:
                         # print("incorrect password")
 336
                         messagebox.askretrycancel("Try Again", "Password Incorrect")
 337
                         break
  338
 339
                else:
                    # print("User Not Found")
  340
                    user_not_found()
 341
                    break
  342
 343
  344
            conn.commit()
            conn.close()
  345
 346
  347
 348
        def delete_s2():
 349
            screen2.destroy()
  350
 351
        #The login page
        def login():
 352
            print("Login session started")
 353
  354
            global screen2
            screen2=Toplevel(screen)
 355
            # screen2=Tk()
  356
  357
            screen2.title("Login page")
- 8 -
```

```
358
          screen2.geometry("400x350")
359
          screen2.configure(bg="#E5F6DF")
360
          Label(screen2, text="Please enter details below to
                                                                                                   \Box
          login", width="400", height="3", bg="#b6e9f2", font=("Times", 16)).pack()
361
          Label(screen2, text="", bg="#E5F6DF").pack()
362
363
          global username_verify
          global password_verify
364
365
          username_verify= StringVar()
366
          password_verify=StringVar()
367
          Label(screen2, text="Enter your registered
                                                                                                   \Box
          Username", font=("Times", 13), bg="#E5F6DF").pack()
368
          global usernamee_entry1
369
          global password_entry1
370
          usernamee_entry1= Entry(screen2, textvariable=username_verify, width=30)
371
          usernamee_entry1.pack()
          Label(screen2, text="Enter your Password", font=("Times", 13), bq="#E5F6DF").pack()
372
          password_entry1=Entry(screen2,textvariable=password_verify, show="*",width=30)
373
374
          password_entry1.pack()
          Label(screen2, text="", bg="#E5F6DF").pack()
375
          Button(screen2, text="Login", width=20, height=2,
376
                                                                                                   \Box
          command=login_verify, bg="#d1ffea").pack()
377
          Label(screen2, text="", bg="#E5F6DF").pack()
378
          Button(screen2, text="Close window", bg="#ffcccb", height=2, width= 20,
                                                                                                   \overline{a}
          command=delete_s2).pack()
379
380
381
                                                                                                   7
382
383
      def main_screen():
384
385
          global screen
386
          screen=Tk()
387
          screen.geometry("400x400")
388
          screen.minsize(400,400)
389
          screen.maxsize(400,400)
390
391
          #QUERIES to create databases(implemented for only once)
          # conn= sqlite3.connect("software.db")
392
393
          # c=conn.cursor()
394
          # c.execute("CREATE TABLE person(name TEXT, age INT, conn_type TEXT,
                                                                                                   7
          conn_type_other TEXT, gender TEXT, username TEXT, password TEXT)")
395
          # conn.commit()
396
          # conn.close()
          # conn= sqlite3.connect("user_info.db")
397
398
          # c=conn.cursor()
399
          # c.execute("CREATE TABLE Curr_session(name TEXT, bluetooth_name TEXT,
                                                                                                   \overline{\downarrow}
```

```
bluetooth_address TEXT)")
400
          # conn.commit()
          # conn.close()
401
402
          screen.title("Sign up/ Login page")
403
          screen.configure(bg="#caf0f8")
404
          Label(text="Sign up/ Log in", width="400", height="3", fg="#caf0f8", bg="#04035e",
405
                                                                                                 Į
          font=("Times", 16, "bold")).pack()
          photo = PhotoImage(file=resource_path("image.png"), width="400", height="150")
406
407
          new=Label(image=photo)
408
          new.pack()
409
          Label(text="", bg="#caf0f8").pack()
          Button(text="Login", height=1, width= 30, bg="#0077b6", font=("Times", 13, "bold"),
410
                                                                                                 \Box
          command=login).pack()
411
          Label(text="", bg="#caf0f8").pack()
          Button(text="Register", height=1, width=
412
                                                                                                 7
          30, bg="#0077b6", font=("Times", 13, "bold"), command=register).pack()
413
          Label(text="", bg="#caf0f8").pack()
          Button(text="Quit", height=1, width= 30,bg="#0077b6",font=("Times",13,"bold"),
414
                                                                                                 7
          command=lambda: screen.quit()).pack()
415
          screen.mainloop()
416
417
      main_screen()
418
```