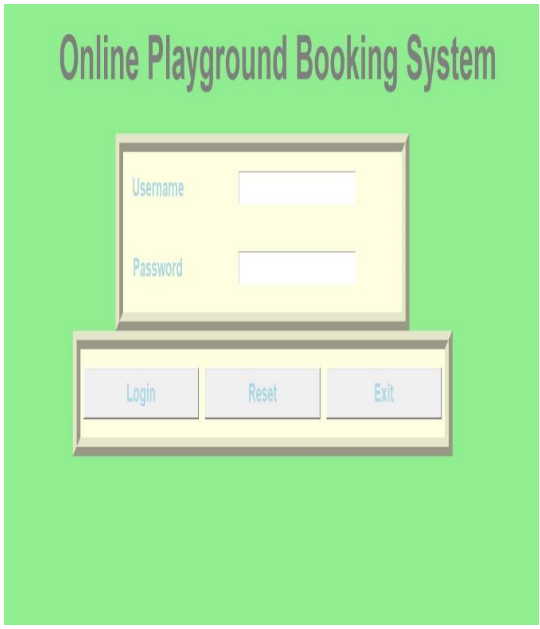
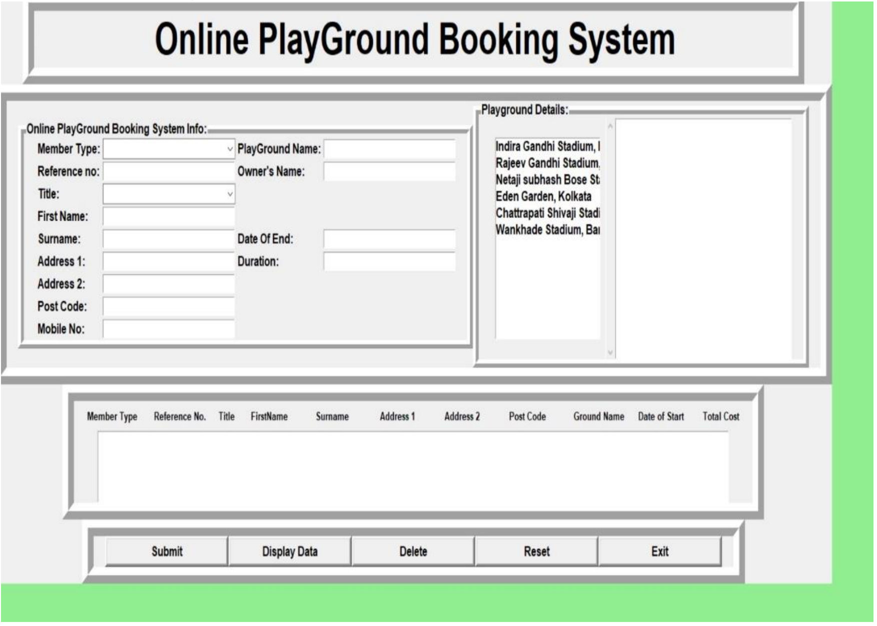
Lovely Professional University, Punjab





Indira Gandhi Stadium, I
Rajeev Gandhi Stadium,
Netaji subhash Bose St
Eden Garden, Kolkata
Chattrapati Shivaji Stadi
Wankhade Stadium, Ba

Project Topic:-

Online Sports Playground Booking System

Submitted To:-

Faculty Name :- Sagar Pande

Submitted By :-

Name :- Ashwani Kumar
Registration Number :- 11806645
Roll Number :- RK18PGB41
Section :- K18PG
Course Code :- INT213 (Python Programming)

Table of Contents :-

Abstract :-

This Sports Playground Booking Website is proposed for booking the Playground in an easy and efficient way. It has three modules namely, Admin, Manager and User. Admin can login and can add playground locations, assign manager by creating login credentials for manager, add price details for the particular Sport Playground, manages various Playground and view the details of sports venues booking for all locations. Managers assigned by the Admin are different for different playground locations. Managers will get login credentials from admin, he/she can login using credentials, he/she can check the rates, view the request for sports booking for the respective location, can accept booking, generate bill and can view the booking history. Users can check the availability of the Sports Playground, select timings, fill personal details, can pay by providing bank details or card details and he/she can also see view previous Sports Playground Booking history.

Introduction :-

Playground are used to play various sports like Football, Rugby, Tennis, Cricket, etc. People enjoy playing on the Spots Playground, It has vibrant environment and huge area to play with safety. Many School Teams and Clubs prefer Sports playground for practice and Training Purpose. Sometime, It becomes difficult to book Sports Playground because of timing issue or the slot getting booked previously.

Objective :-

The objective is too making it easier for people to play sport. Online Sports Playground Booking System is on a mission to remove barriers to getting people active. Whether that's finding facilities, booking activities or connecting people in their local area, we want to simplify it all.

Modules :-

The system comprises of 3 major modules with their sub-modules as follows:

❖ Admin :-

- **Add Manager:** Admin can add Playground location and manager of the respective Playground location.
- **Add Price List:** Admin can add price for the respective playground.

- **Manage Playground:** Admin can manage request by allocating respective Playground.
- **View Booking:** Admin can view booking done and the user details.

❖ Manager :-

- **Login:** Manager can login with the credentials provided by user.
- **Check Rates:** Manager can check rates for the respective location Playground.
- **View Request:** Manager can view request for playground bookings.
- **Confirm Booking:** Manager can confirm the booking of the Playground.
- **Bill Generation:** Manager can generate bills as per the rates.
- **Bookings History:** Manager can check previous booking history

❖ Users Login :-

- **Check playground:** User can check for playground of nearby location and prices.
- **Check Availability:** User can see the availability of the respective Playground which is selected by him.
- **Book Playground:** User can provide date, time and other personal details and he can also do payment.
- **Booking History:** User can see his previous booking history.

❖ Update :-

It is used for updating the client information as per the requirement of the user.

❖Delete :-

It is used for deleting the complete record of the client effectively.

❖ Exit :-

It is used for going back or logging out from the Application.

❖ Reset :-

It is used for resetting the record of a client as per requirement.

❖ Display Data :-

It is used for showing or checking the record of the various clients.

❖ Submit :-

It is used for submitting the record of the client for the confirmation for booking the playground.

Code :-

```
from tkinter import*
from tkinter import ttk
import random
import sqlite3
import tkinter.messagebox
def main():
    root=Tk()
    app=Window1(root)
class Window1:
    def __init__(self,master):
        self.master=master
        self.master.title("Online Playground Booking System")
        self.master.geometry('1350x750+0+0')
        self.master.configure(bg='Light Green')
        self.frame=Frame(self.master, bg='Light Green')
        self.frame.pack()
        self.Username=StringVar()
        self.Password=StringVar()
        self.lblTitle=Label(self.frame, text='Online Playground Booking System', font=('arial',50,'bold'),bg='Light Green',fg='grey')
        self.lblTitle.grid(row=0,column=0,columnspan=2,pady=40)

#==========================================================================================================
=======================
        self.LoginFrame1=LabelFrame(self.frame, width=1350, height=600, font=('arial',20,'bold'),relief='ridge', bg='Light Yellow',bd=20)
        self.LoginFrame1.grid(row=1, column=0)
        self.LoginFrame2=LabelFrame(self.frame, width=1000, height=600, font=('arial',20,'bold'),relief='ridge', bg='Light Yellow',bd=20)
        self.LoginFrame2.grid(row=2, column=0)
  #============================================= Label &
Entry======================================================================
        self.lblUsername=Label(self.LoginFrame1, text= 'Username', font=('arial',20,'bold'), bd=22,bg='Light Yellow' , fg='light blue')
        self.lblUsername.grid( row=0, column=0)
        self.txtUsername=Entry(self.LoginFrame1, font=('arial',20,'bold'),textvariable=self.Username)
        self.txtUsername.grid( row=0, column=1, padx=119)
        self.lblPassword=Label(self.LoginFrame1, text= 'Password', font=('arial',20,'bold'),bd=22,bg='Light Yellow' , fg='light blue')
        self.lblPassword.grid( row=1, column=0)
        self.txtPassword=Entry(self.LoginFrame1, font=('arial',20,'bold'),show = "*", textvariable=self.Password)
        self.txtPassword.grid( row=1, column=1, columnspan=2, pady=30)
  #================================================
Button==========================================================================
        self.btnLogin=Button(self.LoginFrame2, text='Login',width=17,font=('arial',20,'bold'),fg='light blue',command=self.Login_System)
        self.btnLogin.grid(row=3,column=0,pady=20,padx=8)
        self.btnReset=Button(self.LoginFrame2, text='Reset',width=17,font=('arial',20,'bold'),fg='light blue',command=self.Reset)
        self.btnReset.grid(row=3,column=1,pady=20,padx=8)
        self.btnExit=Button(self.LoginFrame2, text='Exit',width=17,font=('arial',20,'bold'),fg='light blue', command=self.iExit)
        self.btnExit.grid(row=3,column=2,pady=20,padx=8)

#==========================================================================================================
========================
    def Login_System(self):
        u=(self.Username.get())
        p=(self.Password.get())
        if(u==str(1234) and p==str(1234)):
            self.newWindow=Toplevel(self.master)
            self.app=Library(self.newWindow)
        else:
            tkinter.messagebox.askyesno("Login Systems", "Invalid login detail")
            self.Username.set("")
            self.Password.set("")
            self.txtUsername.focus()
    def Reset(self):
        self.Username.set("")
        self.Password.set("")
        self.txtUsername.focus()
    def iExit(self):
        self.iExit=tkinter.messagebox.askyesno("Login Systems", "Confirm if you want to exit")
        if self.iExit > 0:
            self.master.destroy()
        else:
            command=self.new_window
            return
    def new_window(self):
        self.newWindow=Toplevel(self.master)
        self.app=PlayGround(self.newWindow)
class Library:
    def __init__(self,master):
        self.master=master
        self.master.title("Online PlayGround Booking System")
        self.master.geometry('1350x750+0+0')
        self.master.configure(bg='Light Green')
        MType=StringVar()
        Ref=StringVar()
        Title=StringVar()
        Firstname=StringVar()
        Surname=StringVar()
        Address1=StringVar()
        Address2=StringVar()
        PostCode=StringVar()
        MobileNo=StringVar()
        GroundName=StringVar()
        GroundArea=StringVar()
        GroundType=StringVar()
        OwnerName=StringVar()
        DateOfStart=StringVar()
        Period=StringVar()
        DateOfEnd=StringVar()
        TotalCost=StringVar()
        Prescription=StringVar()
        def iReset2():
            MType.set("")
            Ref.set("")
            Title.set("")
            Firstname.set("")
            Surname.set("")
            Address1.set("")
            Address2.set("")
            PostCode.set("")
            MobileNo.set("")
            GroundName.set("")
            GroundArea.set("")
            GroundType.set("")
            OwnerName.set("")
            DateOfStart.set("")
            Period.set("")
            DateOfEnd.set("")
            TotalCost.set("")
            self.txtFrameDetail.delete("1.0",END)
            self.txtDisplayR.delete("1.0",END)
        def iDelete():
            iReset2()
            self.txtDisplayR.delete("1.0",END)
        def iEliminate():
            msg=tkinter.messagebox.askyesno("Online Playground Booking System", "Confirm if you want to exit")
            if msg=="True":
               master.quit()
            def iDisplayData():
            self.txtFrameDetail.insert(END,MType.get()+"\t\t"+Ref.get()+"\t"+Title.get()+"\t"+Firstname.get()+"\t"+ Surname.get() +"\t
    \t"+Address1.get()+"\t\t"+Address2.get()+"\t"+PostCode.get()+"\t"+GroundName.get()+"\t\t    "+DateOfStart.get()+"            "+TotalCost.get()+"\n")
        def iReceipt():
            self.txtDisplayR.delete("1.0",END)
            self.txtDisplayR.insert(END, "Member Type: \t\t" + MType.get() + "\n")
            self.txtDisplayR.insert(END, "Ref No : \t\t" + Ref.get() + "\n")
            self.txtDisplayR.insert(END, "Title: \t\t" + Title.get() + "\n")
            self.txtDisplayR.insert(END, "First Name: \t\t" + Firstname.get() + "\n")
            self.txtDisplayR.insert(END, "Surname: \t\t" + Surname.get() + "\n")
            self.txtDisplayR.insert(END, "Address 1: \t\t" + Address1.get() + "\n")
            self.txtDisplayR.insert(END, "Address 2 : \t\t" + Address2.get() + "\n")
            self.txtDisplayR.insert(END, "Post Code: \t\t" + PostCode.get() + "\n")
            self.txtDisplayR.insert(END, "Mobile No: \t\t" + MobileNo.get() + "\n")
            self.txtDisplayR.insert(END, "Ground Name: \t\t" + GroundName.get() + "\n")
            self.txtDisplayR.insert(END, "Date Of Start: \t\t" + DateOfStart.get() + "\n")
            self.txtDisplayR.insert(END, "Owner Name: \t\t" + OwnerName.get() + "\n")
            self.txtDisplayR.insert(END, "Date Of End: \t\t" + DateOfEnd.get() + "\n")
        MainFrame=Frame(self.master)
        MainFrame.grid()
        TitleFrame = Frame(MainFrame, width=1350, padx=20, bd=20, relief=RIDGE)
        TitleFrame.pack(side=TOP)
        self.lblTitle=Label(TitleFrame, width=39 , font=("arial", 40 , "bold"),text="\t Online PlayGround Booking System \t", padx=12)
        self.lblTitle.grid()
        ButtonFrame=Frame(MainFrame, bd=20, width=1350, height=50, padx=20, relief=RIDGE)
        ButtonFrame.pack(side=BOTTOM)
        FrameDetail=Frame(MainFrame, bd=20, width=1350, height=100, padx=20, relief=RIDGE)
        FrameDetail.pack(side=BOTTOM)
        DataFrame=Frame(MainFrame, bd=20, width=1300, height=400, padx=20, relief=RIDGE)
        DataFrame.pack(side=BOTTOM)
        DataFrameLEFT=LabelFrame(DataFrame , bd=10, width=800, height=300, padx=20, relief=RIDGE, font=("arial",12,"bold"), text="Online PlayGround
    Booking System Info:")
        DataFrameLEFT.pack(side=LEFT)
        DataFrameRIGHT=LabelFrame(DataFrame , bd=10, width=450, height=300, padx=20, relief=RIDGE, font=("arial",12,"bold"), text="Playground Details:",)
        DataFrameRIGHT.pack(side=RIGHT)
        #======================== Widgets=====================================#
        self.lblMemberType = Label(DataFrameLEFT, font=("arial", 12, "bold"), text ="Member Type:", padx=2, pady=2)
        self.lblMemberType.grid(row=0, column=0, sticky=W)
        self.cboMemberType = ttk.Combobox(DataFrameLEFT, state="readonly",textvariable=MType, font=("arial", 12, "bold"), width=23)
        self.cboMemberType['value']=('', 'Client', 'Member', 'Admin')
        self.cboMemberType.current(0)
        self.cboMemberType.grid(row=0, column=1)
        self.lblGroundName = Label(DataFrameLEFT, font=("arial", 12, "bold"), text ="PlayGround Name:", padx=2, pady=2)
        self.lblGroundName.grid(row=0, column=2, sticky=W)
        self.txtGroundName=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=GroundName)
        self.txtGroundName.grid(row=0,column=3)
        self.lblRef = Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Reference no:", padx=2,pady=2)
        self.lblRef.grid(row=1,column=0,sticky=W)
        self.txtRef=Entry(DataFrameLEFT, font=("arial", 12, "bold"),textvariable=Ref, width=25)
        self.txtRef.grid(row=1,column=1)
        self.lblOwnerName= Label(DataFrameLEFT, font=("arial", 12, "bold"), text="Owner's Name:", padx=2,pady=2)
        self.lblOwnerName.grid(row=1,column=2,sticky=W)
        self.txtOwnerName=Entry(DataFrameLEFT, font=("arial", 12, "bold"), textvariable=OwnerName,width=25)
        self.txtOwnerName.grid(row=1,column=3)
        self.lblTitle=Label(DataFrameLEFT, font= ("arial",12,"bold"), text="Title:",padx=2,pady=2)
        self.lblTitle.grid(row=2,column=0,sticky=W)
        self.cboTitle=ttk.Combobox(DataFrameLEFT,state="readonly",textvariable=Title,font=("arial",12,"bold"),width=23)
        self.cboTitle['value']=('', 'Mr.', 'Miss.', 'Mrs.', 'Dr.', 'Capt.', 'Ms.')
        self.cboTitle.current(0)
        self.cboTitle.grid(row=2,column=1)
        self.lblFirstName = Label(DataFrameLEFT, font=("arial", 12, "bold"),text="First Name:", padx=2,pady=2)
        self.lblFirstName.grid(row=3,column=0,sticky=W)
        self.txtFirstName=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=Firstname)
        self.txtFirstName.grid(row=3,column=1)
        self.lblDateOfStart= Label(DataFrameLEFT, font=("arial", 12, "bold"), text="Date of Start:", padx=2,pady=2)
        self.lblDateOfStart.grid(row=3,column=2,sticky=W)
        self.txtDateOfStart=Entry(DataFrameLEFT, font=("arial", 12, "bold"), textvariable=DateOfStart,width=25)
        self.txtDateOfStart.grid(row=3,column=3)
        self.lblSurname = Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Surname:", padx=2,pady=2)
        self.lblSurname.grid(row=4,column=0,sticky=W)
        self.txtSurname=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=Surname)
        self.txtSurname.grid(row=4,column=1)
        self.lblDateOfEnd=Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Date Of End:", padx=2,pady=2)
        self.lblDateOfEnd.grid(row=4,column=2,sticky=W)
        self.txtDateOfEnd=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=DateOfEnd)
        self.txtDateOfEnd.grid(row=4,column=3)
        self.lblAddress1=Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Address 1:", padx=2,pady=2)
        self.lblAddress1.grid(row=5,column=0,sticky=W)
        self.txtAddress1=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=Address1)
        self.txtAddress1.grid(row=5,column=1)
        self.lblPeriod=Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Duration:", padx=2,pady=2)
        self.lblPeriod.grid(row=5,column=2,sticky=W)
        self.txtPeriod=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=Period)
        self.txtPeriod.grid(row=5,column=3)
        self.lblAddress2 = Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Address 2:", padx=2,pady=2)
        self.lblAddress2.grid(row=6,column=0,sticky=W)
        self.txtAddress2=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=Address2)
        self.txtAddress2.grid(row=6,column=1)
        self.lblTotalCost= Label(DataFrameLEFT, font=("arial", 12, "bold"), text="Total Cost:", padx=2,pady=2)
        self.lblTotalCost.grid(row=6,column=2,sticky=W)
        self.txtTotalCost=Entry(DataFrameLEFT, font=("arial", 12, "bold"), textvariable=TotalCost,width=25)
        self.txtTotalCost.grid(row=6,column=3)
        self.lblPostCode=Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Post Code:", padx=2,pady=2)
        self.lblPostCode.grid(row=7,column=0,sticky=W)
        self.txtPostCode=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=PostCode)
        self.txtPostCode.grid(row=7,column=1)
        self.lblMobileNo=Label(DataFrameLEFT, font=("arial", 12, "bold"),text="Mobile No:", padx=2,pady=2)
        self.lblMobileNo.grid(row=8,column=0,sticky=W)
        self.txtMobileNo=Entry(DataFrameLEFT, font=("arial", 12, "bold"),width=25,textvariable=MobileNo)
        self.txtMobileNo.grid(row=8,column=1)
        #========================Widgets==========================================================================#
        self.txtDisplayR=Text(DataFrameRIGHT, font=("arial", 12, "bold"),width=32, height=13, padx=8,pady=20)
        self.txtDisplayR.grid(row=0, column=2)
        scrollbar=Scrollbar(DataFrameRIGHT)
        scrollbar.grid(row=0,column=1,sticky='ns')
        ListOfPlayGround = ['Indira Gandhi Stadium, New Delhi','Rajeev Gandhi Stadium, Hyderabad','Netaji subhash Bose Stadium, Kolkata','Eden Garden,
    Kolkata', 'Chattrapati Shivaji Stadium, Mumbai',
                    'Wankhade Stadium, Bangalore']
        def SelectedPlayGround(evt):
            value=str(PlayGround.get(PlayGroundList.curselection()))
            w=value
            conn=sqlite3.connect('PlayGround.db')
            c=conn.cursor()
            c.execute('''SELECT * FROM PlayGroundDb WHERE Book_Title =?''' ,(w,))
            for row in c.fetchall():
                GroundName.set(row[0])
                OwnerName.set(row[1])
                DateOfStart.set(row[2])
                Period.set(row[3])
                DayOfEnd.set(4)
                TotalCost.set(row[5])
        PlayGround= Listbox(DataFrameRIGHT, width=20,height=12,font=('arial',12,'bold'),yscrollcommand=scrollbar.set)
        PlayGround.bind('<<ListboxSelect>>',SelectedPlayGround)
        PlayGround.grid(row=0, column=0,padx=8)
        scrollbar.config(command=PlayGround.yview)
        for items in ListOfPlayGround:
            PlayGround.insert(END,items)
        #==========================Labels===================================================================================#
        self.lblLabel=Label(FrameDetail, font=("arial",10,'bold'), pady=8,
        text="Member Type\t Reference No.\t Title\t FirstName\tSurname\tAddress 1\tAddress 2\tPost Code\tGround Name\tDate of Start" " Total Cost")
        self.lblLabel.grid(row=0, column=0)
        self.txtFrameDetail=Text(FrameDetail,font=('arial',12,'bold'),width=121,height=4,padx=2, pady=4)
        self.txtFrameDetail.grid(row=1,column=0)
        #==========================Buttons==================================================================================#
        self.btnDisplayData=Button(ButtonFrame, text='Display Data', font=('arial',12,'bold'),width=20, bd=4,command=iDisplayData)
        self.btnDisplayData.grid(row=0,column=1)
        self.btnDelete=Button(ButtonFrame, text='Delete', font=('arial',12,'bold'),width=20, bd=4,command=iDelete)
        self.btnDelete.grid(row=0,column=2)
        self.btnReset1=Button(ButtonFrame, text='Reset', font=('arial',12,'bold'),width=20, bd=4, command=iReset2)
        self.btnReset1.grid(row=0,column=3)
        self.btnExit1=Button(ButtonFrame, text='Exit', font=('arial',12,'bold'),width=20, bd=4, command=iEliminate)
        self.btnExit1.grid(row=0,column=4)
        self.btnSubmit=Button(ButtonFrame, text='Submit', font=('arial',12,'bold'),width=20, bd=4, command=iReceipt)
        self.btnSubmit.grid(row=0,column=0)
        #======================Frames===========================================#
    if __name__=="__main__":
      main()
```

Software Required :-

- Any Python IDE i.e., Jupyter, or Spyder, or Pycharm

Libraries Used :-

- Any Python IDE
- An Application Of Classes and Object
- Mysql database
- GUI Application
- Embedding Tools of Classes and Objects in Python
- from tkinter import*
- from tkinter import ttk
- import random
- import sqlite3
- import time
- from datetime import datetime
- import tkinter.messagebox

Advantages of the Proposed Project :-

- Saves Time (It checks the availability of all playground at single platform)

- Easy to access the system anywhere and anytime.

Disadvantages :-

- Requires an active internet connection.

Application :-

- This application can be used by companies and people for sports events, and sports enthusiastic.

Reference :-

- ✓ https://shsu-ir.tdl.org/shsu-ir/bitstream/handle/20.500.11875/1164/0781.pdf?sequence=1
- ✓ https://ieeexplore.ieee.org/document/6208293/
- ✓ https://ieeexplore.ieee.org/document/4679917/
- ✓ https://www.w3schools.com/python/
- ✓ https://www.youtube.com/watch?v=G0rQ7AEl5LA
- ✓ https://www.youtube.com/watch?v=BHh654_7Cmw
- ✓ https://www.youtube.com/watch?v=1zfYZ1N4JEc&list=PLHuy11O2fJAWIqBEFuq8W-6YU53U_Qlac
- ✓ https://www.youtube.com/watch?v=_uQrJ0TkZlc&list=PLbMN7vqUdX-fxzc7ousbQOllh1VGnvrFq
- ✓ https://www.youtube.com/watch?v=hqC9tioGCi0
- ✓ https://www.youtube.com/watch?v=aqz8KMxCC9M
- ✓ https://www.youtube.com/watch?v=PPXhqYkKCDw
- ✓ https://www.javatpoint.com/python-tutorial
- ✓ https://www.tutorialspoint.com/python/index.html