IPL STATISTICS

A MINI PROJECT REPORT

Submitted by

Skandha TR 231501156

In partial fulfillment for the award of the degree of

BACHELOR OF

ENGINEERING IN

Artificial Intelligence And Machine Learning

RAJALAKSHMI ENGINEERING COLLEGE (AUTONOMOUS)

THANDALAM

CHENNAI-602105

2024 - 2025

BONAFIDE CERTIFICATE

Certified that this project report "**IPL STATISTICS**" is the bonafide work

of **"SIBHINANDHAN ER(231501155),SKANDHA TR**

(231501156),SHRIRAM N (231501154)"

who carried out the project work under my supervision.

Submitted for the Practical Examination held on __________________________

SIGNATURE **SIGNATURE**

INTERNAL EXAMINER **EXTERNAL EXAMINER**

INDEX

Project Synopsis

Background:-

IPL Statistics is an application which is used by the team management to view, sort statistics of players and teams to aid them in selecting a player for their team.

Sort and display modules are also included in this system which would keep track of a player's/team's performance in a particular field in a year of the league. With this computerized system there will be no wrong values entered which generally happens when a non-computerized system is used.

All these modules are able to help the team management to view player/team statistics with more convenience and in a more efficient way as compared to systems which are non-computerized.

SYSTEM REQUIREMENTS

SYSTEM:

OS- Window 10 Home Single Language

(19045.2364 OS Build)

Language: English

Processor: Intel(R) Core(TM) i3-4005U CPU @ 1.70GHz 1.70 GHz

Memory: 8.00 GB RAM

SOFTWARE:

MySQL Community server 8.0.29

Python IDLE version3.10.4

FUNCTIONS AND MODULES USED:

Functions:

play():

Used to play the music if the start button is clicked.

pause():

Used to stop the sound if the pause button is clicked.

AUC2019():

Used to view and sort statistics of players in the 2019 IPL auction.

AUC2020():

Used to view and sort statistics of players in the 2020 IPL auction

AUC2021():

Used to view and sort statistics of players in the 2021 IPL auction.

MVP1():

Used to view and sort statistics of Most Valuable players in the 2019 edition of the IPL.

MVP2():

Used to view and sort statistics of Most Valuable players in the 2020 edition of the IPL.

MVP3():

Used to view and sort statistics of Most Valuable players in the 2021 edition of the IPL.

BT2019():

Used to view and sort statistics of Batsmen in the 2019 edition of the IPL.

BT2020():

Used to view and sort statistics of Batsmen in the 2020 edition of the IPL.

BT2021():

Used to view and sort statistics of Batsmen in the 2021 edition of the IPL.

BO2019():

Used to view and sort statistics of Bowlers in the 2019 edition of the IPL.

BO2020():

Used to view and sort statistics of Bowlers in the 2020 edition of the IPL.

BO2021():

Used to view and sort statistics of Bowlers in the 2021 edition of the IPL.

PT2019():

Used to view the points table of the 2019 edition of the IPL.

PT2020():

Used to view the points table of the 2020 edition of the IPL.

PT2021():

Used to view the points table of the 2021 edition of the IPL.

Modules:-

Import mysql.connector:

By importing this package, we are establish a connection between python and MySql.

import sys:

By importing this package, The sys module in Python provides various functions and variables that are used to manipulate different parts of the Python runtime environment. It allows operating on the interpreter as it provides access to the variables and functions that interact strongly with the interpreter.

import os:

The functions OS module provides allows us to operate on underlying Operating System tasks, irrespective of it being a Windows Platform, Macintosh or Linux.

import PIL:

The Python Imaging Library adds image processing capabilities to your Python interpreter.

import PIL.Image:

The Image module provides a class with the same name which is used to represent a PIL image.

import pygame:

pygame is a Python wrapper for the SDL library, which stands for Simple DirectMedia Layer. SDL provides cross-platform access to your system's underlying multimedia hardware components, such as sound, video, mouse, keyboard, and joystick.

import tkinter:

Tkinter is the inbuilt python module that is used to create GUI applications.

Use of Technology

MySQL:

MySQL is a relational DBMS that can run virtually all platforms, including Linux, Unix and Windows. MySQL is a freely available open source RDBMS that uses Structured Query Language (SQL). It is fast, reliable, scalable alternative to many of the commercial RDBMs available today. It provides you with a rich set of features that support a secure environment for storing, maintaining, and accessing data. MySQL was created and supported by MySQL AB, a company based in Sweden. SQL provides many different types of commands used for different purposes. SQL commands can be divided into following categories:

i. Data Definition Language (DDL)

ii. Data Manipulation Language (DML)

iii. Transaction Control Language (TCL)

iv. Session Control Commands

v. System Control Commands Page

Python:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. It is high-level built in data structures, combined with dynamic typing and dynamic binding which make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. Often, programmers fall in love with Python because of the increased productivity it provides. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

Code:

Main prg code:

```
import tkinter as tk
import sys
import os
sys.path.append(os.path.abspath(r"C:\Users\vsais\IPL\AUCTION"))
sys.path.append(os.path.abspath(r"C:\Users\vsais\IPL\MVP"))
sys.path.append(os.path.abspath(r"C:\Users\vsais\IPL\LOGO"))
sys.path.append(os.path.abspath(r"C:\Users\vsais\IPL\PLAYER PERFORMANCE"))
sys.path.append(os.path.abspath(r"C:\Users\vsais\IPL\POINTS TABLE"))
from PPTW import *
from PTTW import *
from PTWW import *
from MVP19 import *
from MVP20 import *
from MVP21 import *
from PIL import Image, ImageTk
from BAT2020 import *
from BAT2019 import *
from BAT2021 import *
from BOWL2019 import *
from BOWL2020 import *
from BOWL2021 import *
from AUCTION19 import *
from AUCTION20 import *
from AUCTION21 import *
from AUCUNSOLD19 import *
from AUCUNSOLD20 import *
from AUCUNSOLD21 import *
```

```
import PIL.Image
import pygame
root=Tk()
l = Label(root)
l.pack()
root.geometry('750x750')
root.title("IPL STATISTICS")
root.attributes('-fullscreen', True)
pygame.mixer.init()
def play():
   pygame.mixer.music.load(r'C:\Users\vsais\IPL\Music\BGM.mp3')
   pygame.mixer.music.play(loops=-1)
def pause():
  pygame.mixer.music.pause()
def PP19():
  top= Toplevel(root)
  top.geometry("750x500")
  C=Label(top,text='DO YOU WANT TO CHECK:',font=('bold',16),anchor=CENTER)
  C.place(x=100,y=0)



  Button(top,text= "Batting Stats", command=BT2019).pack(pady= 5)
  button= Button(top, text="Bowling Stats", command=BO2019)
  button.pack(pady=5)

def PP20():
  top= Toplevel(root)
  top.geometry("750x500")
  C=Label(top,text='DO YOU WANT TO CHECK:',font=('bold',16),anchor=CENTER)
  C.place(x=100,y=0)
  Button(top,text= "Batting Stats", command=BT2020).pack(pady= 5)
  button= Button(top, text="Bowling Stats", command=BO2020)
  button.pack(pady=5)

def PP21():
  top= Toplevel(root)
  top.geometry("750x500")
  C=Label(top,text='DO YOU WANT TO CHECK:',font=('bold',16),anchor=CENTER)
  C.place(x=100,y=0)



  Button(top,text= "Batting Stats", command=BT2021).pack(pady= 5)
  button= Button(top, text="Bowling Stats", command=BO2021)
  button.pack(pady=5)

def A21():
  top= Toplevel(root)
  top.geometry("750x500")
  C=Label(top,text='DO YOU WANT TO CHECK:',font=('bold',16),anchor=CENTER)
  C.place(x=100,y=0)

  Button(top,text= "Sold Players", command=AUC2021).pack(pady= 5)
  button= Button(top, text="Unsold Players", command=AUCUN2021)
  button.pack(pady=5)

def A20():
  top= Toplevel(root)
  top.geometry("750x500")
  C=Label(top,text='DO YOU WANT TO CHECK:',font=('bold',16),anchor=CENTER)
  C.place(x=100,y=0)

  Button(top,text= "Sold Players", command=AUC2020).pack(pady= 5)
  button= Button(top, text="Unsold Players", command=AUCUN2020)
  button.pack(pady=5)

def A19():
  top= Toplevel(root)
  top.geometry("750x500")
  C=Label(top,text='DO YOU WANT TO CHECK:',font=('bold',16),anchor=CENTER)
  C.place(x=100,y=0)

  Button(top,text= "Sold Players", command=AUC2019).pack(pady= 5)
  button= Button(top, text="Unsold Players", command=AUCUN2019)
  button.pack(pady=5)



MVP=Label(root,text='MVP(Most Valuable Player)',font=('bold',10),bg="yellow",anchor=CENTER)
MVP.place(x=335,y=300)
PT=Label(root,text='Points Table',font=('bold',10),bg="yellow",anchor=CENTER)
PT.place(x=600,y=300)
CS=Label(root,text='Player Stats',font=('bold',10),bg="yellow",anchor=CENTER)
CS.place(x=800,y=300)
CSS=Label(root,text='Auction Data',font=('bold',10),bg="yellow",anchor=CENTER)
CSS.place(x=1000,y=300)
MVPOne=Button(root,text="2019",font=("italic",12),bg="light blue", command=MVP1)
MVPOne.place(x=390,y=400)
MVPTwo=Button(root,text="2020",font=("italic",12),bg="light blue", command=MVP2)
MVPTwo.place(x=390,y=500)

MVPTwo=Button(root,text="2021",font=("italic",12),bg="light blue", command=MVP3)
MVPTwo.place(x=390,y=600)
PTOne=Button(root,text="2019",font=("italic",12),bg="light blue", command=PT2019)
```

```
PTOne.place(x=610,y=400)
PTTwo=Button(root,text="2020",font=("italic",12),bg="light blue", command=PT2020)
PTTwo.place(x=610,y=500)
PTThree=Button(root,text="2021",font=("italic",12),bg="light blue", command=PT2021)
PTThree.place(x=610,y=600)
PSOne=Button(root,text="2019",font=("italic",12),bg="light blue", command=PP19)
PSOne.place(x=810,y=400)
PSOne=Button(root,text="2020",font=("italic",12),bg="light blue", command=PP20)
PSOne.place(x=810,y=500)
PSOne=Button(root,text="2021",font=("italic",12),bg="light blue", command=PP21)
PSOne.place(x=810,y=600)
AucOne=Button(root,text="2019",font=("italic",12),bg="light blue", command=A19)
AucOne.place(x=1010,y=400)
AucTwo=Button(root,text="2020",font=("italic",12),bg="light blue", command=A20)
AucTwo.place(x=1010,y=500)
AucThree=Button(root,text="2021",font=("italic",12),bg="light blue", command=A21)
AucThree.place(x=1010,y=600)
songpl=Button(root,text="Play Music",font=("italic",12),bg="light blue", command=play)
songpl.place(x=610,y=800)
songpa=Button(root,text="Pause Music",font=("italic",12),bg="light blue", command=pause)
songpa.place(x=810,y=800)
def close():
    root.destroy()
    pygame.mixer.music.stop()
exit_button = Button(root, text="Exit",bg="red", command=close)
exit_button.pack(pady=10)
i=PIL.Image.open(r"C:\Users\vsais\IPL\LOGO\logo.png")
r=i.resize((350,205),PIL.Image.Resampling.LANCZOS)
s=ImageTk.PhotoImage(r)
img=Label(root, image=s)
img.place(x=600, y=60)
```

```
root.mainloop()
```

i) 2019 Auction(Sold Players):

```
from tkinter import *
import tkinter as tk
from tkinter import ttk
import mysql.connector as pro
e=''
def AUC2019():
   def update(rows):
        tree.delete(*tree.get_children())
        for i in rows:
           tree.insert('','end',values=i)
   def default():
        global e
        e=''
        ab=Ent.get()
        cur.execute("SELECT * from auction2019")
        rows=cur.fetchall()
        update(rows)
   def bowl():
        ab=Ent.get()
        cur.execute("SELECT * from auction2019 where Player LIKE '%"+ab+"%' and
Type LIKE '%Bowl%' and Team LIKE '%"+e+"%' ")
        rows=cur.fetchall()
        update(rows)
   def bat():
        ab=Ent.get()
        cur.execute("SELECT * from auction2019 where Player LIKE '%"+ab+"%' and
Type LIKE '%Bat%' and Team LIKE '%"+e+"%' ")
        rows=cur.fetchall()
        update(rows)
   def allr():
        ab=Ent.get()
        cur.execute("SELECT * from auction2019 where Player LIKE '%"+ab+"%' and
Type LIKE '%All%' and Team LIKE '%"+e+"%'")
        rows=cur.fetchall()
        update(rows)



   def wk():
        ab=Ent.get()
        cur.execute("SELECT * from auction2019 where Player LIKE '%"+ab+"%' and
Type LIKE '%Keep%' and Team LIKE '%"+e+"%'")
        rows=cur.fetchall()
        update(rows)
   def price():
        ab=Ent.get()
```

```
      cur.execute("SELECT * from auction2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' order by length(Price) desc,Price desc;")
      rows=cur.fetchall()
      update(rows)
  def player():
      ab=Ent.get()
      cur.execute("SELECT * from auction2019 where Player LIKE '%"+ab+"%'")
      player1()
  def player1():
      rows=cur.fetchall()
      c=cur.rowcount
      if c==1:
        k=0
        co=[["Player","Team","Type","Price"]]
        op=Tk()
        w=LabelFrame(op,text="Single player stats")
        w.pack(fill="both",expand="yes",padx=20,pady=10)
        ree = ttk.Treeview(w, column=("Attributes","Stats"), show='headings',
height=4)
        ree.pack()
        ree.column("# 1", anchor=CENTER)
        ree.heading("# 1", text="Attributes")
        ree.column("# 2", anchor=CENTER)
        ree.heading("# 2", text="Stats")
        for i in rows:
          while k<=3:
            co.append(i)
            co[1]=list(co[1])
            ree.insert('','end',values=(co[0][k],co[1][k]))
            k+=1
        def close():
          op.destroy()
        exit_button = Button(op, text="Exit", command=close)
        exit_button.pack(pady=20)
      else:
        update(rows)



  def CSK():
      global e
      e='Chennai Super Kings'
      ab=Ent.get()
      cur.execute("SELECT * from auction2019 where Team LIKE '%Chennai%'")
      rows=cur.fetchall()
      update(rows)
  def MI():
      global e
      e='Mumbai Indians'
```

```
    ab=Ent.get()
    cur.execute("SELECT * from auction2019 where Team LIKE '%Mumbai%'")
    rows=cur.fetchall()
    update(rows)
def DC():
    global e
    e='Delhi Capitals'
    ab=Ent.get()
    cur.execute("SELECT * from auction2019 where Team LIKE '%Delhi%'")
    rows=cur.fetchall()
    update(rows)
def KKR():
    global e
    e='Kolkata Knight Riders'
    ab=Ent.get()
    cur.execute("SELECT * from auction2019 where Team LIKE '%Kolkata%'")
    rows=cur.fetchall()
    update(rows)
def KXIP():
    global e
    e='Kings XI Punjab'
    ab=Ent.get()
    cur.execute("SELECT * from auction2019 where Team LIKE '%Punjab%'")
    rows=cur.fetchall()
    update(rows)
def RR():
    global e
    e='Rajasthan Royals'
    ab=Ent.get()
    cur.execute("SELECT * from auction2019 where Team LIKE '%Rajasthan%'")
    rows=cur.fetchall()
    update(rows)
def RCB():
    global e
    e='Royal Challengers Bangalore'
    ab=Ent.get()
    cur.execute("SELECT * from auction2019 where Team LIKE '%Bangalore%'")
    rows=cur.fetchall()
    update(rows)


def SRH():
    global e
    e='Sunrisers Hyderabad'
    ab=Ent.get()
    cur.execute("SELECT * from auction2019 where Team LIKE '%Hyderabad%'")
    rows=cur.fetchall()
    update(rows)
    con=pro.connect(host='localhost',user='root',passwd='VSS@07',database='ipl')
    cur=con.cursor()

    ws=Tk()
    b=StringVar()
    ws.geometry("700x350")
    style = ttk.Style()
    style.theme_use('clam')
    w1=LabelFrame(ws,text="Sold Players")
    w2=LabelFrame(ws,text="Search Player")
    w3=LabelFrame(ws,text="Sort Team")
    w1.pack(fill="both",expand="yes",padx=20,pady=10)
    w2.pack(fill="both",expand="yes",padx=20,pady=10)
    w3.pack(fill="both",expand="yes",padx=20,pady=10)
    tree = ttk.Treeview(w1, column=("Player","Team","Type","Price"),
   show='headings', height=30)
    tree.pack()
    tree.column("# 1", anchor=CENTER)
    tree.heading("# 1", text="Player")
    tree.column("# 2", anchor=CENTER)
    tree.heading("# 2", text="Team")
    tree.column("# 3", anchor=CENTER)
    tree.heading("# 3", text="Type")
    tree.column("# 4", anchor=CENTER)
    tree.heading("# 4", text="Price")
    cur=con.cursor()
    cur.execute("select * from auction2019")
    rows=cur.fetchall()
    update(rows)
    Lab=Label(w2,text="Search")
    Lab.pack(side=tk.LEFT,padx=10)
    Ent=Entry(w2,textvariable=b)
    Ent.pack(side=tk.LEFT,padx=6)
    bt=Button(w2,text="Search Player",command=player)
    bt.pack(side=tk.LEFT,padx=6)
    bt=Button(w2,text="Default",command=default)
    bt.pack(side=tk.LEFT,padx=10)
    bt=Button(w2,text="Sort by Bowlers",command=bowl)
    bt.pack(side=tk.LEFT,padx=6)
    bt=Button(w2,text="Sort by Batsmen",command=bat)
    bt.pack(side=tk.LEFT,padx=6)
    bt=Button(w2,text="Sort by All-Rounders",command=allr)
    bt.pack(side=tk.LEFT,padx=6)
    bt=Button(w2,text="Sort by Wicket Keepers",command=wk)
    bt.pack(side=tk.LEFT,padx=6)
    bt=Button(w2,text="Sort by Price",command=price)
    bt.pack(side=tk.LEFT,padx=6)
```

```
  but=Button(w3,text="Default",command=default)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Chennai Super Kings",command=CSK)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Mumbai Indians",command=MI)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Delhi Capitals",command=DC)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Kolkata Knight Riders",command=KKR)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Kings XI Punjab",command=KXIP)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Rajasthan Royals",command=RR)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Royal Challengers Bangalore",command=RCB)
  but.pack(side=tk.LEFT,padx=6)
  but=Button(w3,text="Sunrisers Hyderabad",command=SRH)
  but.pack(side=tk.LEFT,padx=6)
  ws.attributes('-fullscreen', True)
  def close():
    ws.destroy()
  exit_button = Button(ws, text="Exit", command=close)
  exit_button.pack(pady=20)
  ws.mainloop()
```

ii) 2019 Auction(Unsold Players):

```
from tkinter import *
import tkinter as tk
from tkinter import ttk
import mysql.connector as pro

def AUCUN2019():
    def update(rows):
        tree.delete(*tree.get_children())
            for i in rows:
                tree.insert('','end',values=i)

def default():
    ab=Ent.get()
    cur.execute("SELECT * from aucunsold2019")
    rows=cur.fetchall()
    update(rows)

def bowl():
     ab=Ent.get()
     cur.execute("SELECT * from aucunsold2019 where Player LIKE '%"+ab+"%' and 
Type LIKE '%Bowl%' ")
     rows=cur.fetchall()
     update(rows)
```

```
                def bat():
                    ab=Ent.get()
                    cur.execute("SELECT * from aucunsold2019 where Player LIKE '%"+ab+"%' and 
Type LIKE '%Bat%' ")
                    rows=cur.fetchall()
                    update(rows)

                def allr():
                    ab=Ent.get()
                    cur.execute("SELECT * from aucunsold2019 where Player LIKE '%"+ab+"%' and 
Type LIKE '%All%'")
                    rows=cur.fetchall()
                    update(rows)

                def wk():
                    ab=Ent.get()
                    cur.execute("SELECT * from aucunsold2019 where Player LIKE '%"+ab+"%' and 
Type LIKE '%Keep%'")
                    rows=cur.fetchall()
                    update(rows)

                def price():
                    ab=Ent.get()
                    cur.execute("SELECT * from aucunsold2019 where Player LIKE '%"+ab+"%' order 
by length(Price) desc,Price desc")
                    rows=cur.fetchall()
                    update(rows)


                def player():
                    ab=Ent.get()
                    cur.execute("SELECT * from aucunsold2019 where Player LIKE '%"+ab+"%'")
                    player1()

                def player1():
                    rows=cur.fetchall()
                    c=cur.rowcount
                    if c==1:
                      k=0
                      co=[["Player","Type","Base Price"]]
                      op=Tk()
                      w=LabelFrame(op,text="Single player stats")
                      w.pack(fill="both",expand="yes",padx=20,pady=10)
                      ree = ttk.Treeview(w, column=("Attributes","Stats"), show='headings', 
height=3)
                      ree.pack()
                      ree.column("# 1", anchor=CENTER)
                      ree.heading("# 1", text="Attributes")
                      ree.column("# 2", anchor=CENTER)
                      ree.heading("# 2", text="Stats")
```

```
        for i in rows:
          while k<=2:
             co.append(i)
             co[1]=list(co[1])
             ree.insert('','end',values=(co[0][k],co[1][k]))
             k+=1
        def close():
          op.destroy()
          exit_button = Button(op, text="Exit", command=close)
          exit_button.pack(pady=20)
    else:
        update(rows)



con=pro.connect(host='localhost',user='root',passwd='VSS@07',database='ipl')
cur=con.cursor()

ws=Tk()
b=StringVar()
ws.geometry("700x350")
style = ttk.Style()
style.theme_use('clam')



w1=LabelFrame(ws,text="Unsold Players")
w2=LabelFrame(ws,text="Search Player")

w1.pack(fill="both",expand="yes",padx=20,pady=10)
w2.pack(fill="both",expand="yes",padx=20,pady=10)



tree = ttk.Treeview(w1, column=("Player","Type","Base Price"), show='headings', height=30)

tree.pack()
tree.column("# 1", anchor=CENTER)
tree.heading("# 1", text="Player")
tree.column("# 2", anchor=CENTER)
tree.heading("# 2", text="Type")
tree.column("# 3", anchor=CENTER)
tree.heading("# 3", text="Base Price")




cur=con.cursor()
cur.execute("select * from aucunsold2019")
rows=cur.fetchall()
update(rows)
```

```
Lab=Label(w2,text="Search")
Lab.pack(side=tk.LEFT,padx=10)
Ent=Entry(w2,textvariable=b)
Ent.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Search Player",command=player)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Default",command=default)
bt.pack(side=tk.LEFT,padx=10)
bt=Button(w2,text="Sort by Bowlers",command=bowl)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Batsmen",command=bat)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by All-Rounders",command=allr)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Wicket Keepers",command=wk)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Base Price",command=price)
bt.pack(side=tk.LEFT,padx=6)


ws.attributes('-fullscreen', True)
def close():
   ws.destroy()

exit_button = Button(ws, text="Exit", command=close)
exit_button.pack(pady=20)
ws.mainloop()
```

iii) 2019 MVP:

```
from tkinter import *
import tkinter as tk
from tkinter import ttk
import mysql.connector as pro
e=''
def MVP1():
   def update(rows):
         tree.delete(*tree.get_children())
```

```
      for i in rows:
        tree.insert('','end',values=i)
  def default():
      global e
      e=''
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019")
      rows=cur.fetchall()
      update(rows)
  def sixes():
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' Order by 6s DESC")
      rows=cur.fetchall()
      update(rows)
  def fours():
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' Order by 4s DESC")
      rows=cur.fetchall()
      update(rows)
  def wickets():
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' Order by wickets DESC")
      rows=cur.fetchall()
      update(rows)
  def dot():
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' Order by DotBalls DESC")
      rows=cur.fetchall()
      update(rows)
  def player():
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Player LIKE '%"+ab+"%'")
      player1()

  def player1():
      rows=cur.fetchall()
      c=cur.rowcount
      if c==1:
        k=0
        co=[["Position", "Player","Team","Matches","Wickets","Dot
Balls","Fours","Sixes","Catches","Stumpings","Run Outs","Points"]]
        op=Tk()
        w=LabelFrame(op,text="Single player stats")
        w.pack(fill="both",expand="yes",padx=20,pady=10)
        ree = ttk.Treeview(w, column=("Attributes","Stats"), show='headings',
height=12)
        ree.pack()
        ree.column("# 1", anchor=CENTER)
        ree.heading("# 1", text="Attributes")
        ree.column("# 2", anchor=CENTER)
        ree.heading("# 2", text="Stats")
        for i in rows:
          while k<=11:
            co.append(i)
            co[1]=list(co[1])
            ree.insert('','end',values=(co[0][k],co[1][k]))
            k+=1
        def close():
          op.destroy()
        exit_button = Button(op, text="Exit", command=close)
        exit_button.pack(pady=20)
      else:
        update(rows)
  def CSK():
      global e
      e='Chennai Super Kings'
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Team LIKE '%Chennai%'")
      rows=cur.fetchall()
      update(rows)
  def MI():
      global e
      e='Mumbai Indians'
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Team LIKE '%Mumbai%'")
      rows=cur.fetchall()
      update(rows)
  def DC():
      global e
      e='Delhi Capitals'
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Team LIKE '%Delhi%'")
      rows=cur.fetchall()
      update(rows)
  def KKR():
      global e
      e='Kolkata Knight Riders'
      ab=Ent.get()
      cur.execute("SELECT * from mvp2019 where Team LIKE '%Kolkata%'")
      rows=cur.fetchall()
      update(rows)
  def KXIP():
        global e
        e='Kings XI Punjab'
        ab=Ent.get()
        cur.execute("SELECT * from mvp2019 where Team LIKE '%Punjab%'")
        rows=cur.fetchall()
        update(rows)
  def RR():
        global e
        e='Rajasthan Royals'
        ab=Ent.get()
        cur.execute("SELECT * from mvp2019 where Team LIKE '%Rajasthan%'")
        rows=cur.fetchall()
        update(rows)
  def RCB():
        global e
        e='Royal Challengers Bangalore'
        ab=Ent.get()
        cur.execute("SELECT * from mvp2019 where Team LIKE '%Bangalore%'")
        rows=cur.fetchall()
        update(rows)
  def SRH():
        global e
        e='Sunrisers Hyderabad'
        ab=Ent.get()
        cur.execute("SELECT * from mvp2019 where Team LIKE '%Hyderabad%'")
        rows=cur.fetchall()
        update(rows)

  con=pro.connect(host='localhost',user='root',passwd='VSS@07',database='ipl')

  cur=con.cursor()
  ws=Tk()
  b=StringVar()
  ws.geometry("700x350")
  style = ttk.Style()
  style.theme_use('clam')
  w1=LabelFrame(ws,text="Most Valuble Player")
  w2=LabelFrame(ws,text="Search Player")
  w3=LabelFrame(ws,text="Sort Team")
  w1.pack(fill="both",expand="yes",padx=20,pady=10)
  w2.pack(fill="both",expand="yes",padx=20,pady=10)
  w3.pack(fill="both",expand="yes",padx=20,pady=10)
  tree = ttk.Treeview(w1, column=( "Position",
"Player","Team","Matches","Wickets","Dot
Balls","Fours","Sixes","Catches","Stumpings","Run Outs","Points"), show='headings',
height=30)
  tree.pack()
  tree.column("# 1", anchor=CENTER)
  tree.heading("# 1", text="Position")
tree.column("# 2", anchor=CENTER)
tree.heading("# 2", text="Player")
tree.column("# 3", anchor=CENTER)
tree.heading("# 3", text="Team")
tree.column("# 4", anchor=CENTER)
tree.heading("# 4", text="Matches")
tree.column("# 5", anchor=CENTER)
tree.heading("# 5", text="Wickets")
tree.column("# 6", anchor=CENTER)
tree.heading("# 6", text="Dot Balls")
tree.column("# 7", anchor=CENTER)
tree.heading("# 7", text="Fours")
tree.column("# 8", anchor=CENTER)
tree.heading("# 8", text="Sixes")
tree.column("# 9", anchor=CENTER)
tree.heading("# 9", text="Catches")
tree.column("# 10", anchor=CENTER)
tree.heading("# 10", text="Stumpings")
tree.column("# 11", anchor=CENTER)
tree.heading("# 11", text="Run Outs")
tree.column("# 12", anchor=CENTER)
tree.heading("# 12", text="Points")
cur=con.cursor()
cur.execute("select * from mvp2019")
rows=cur.fetchall()
update(rows)
Lab=Label(w2,text="Search")
Lab.pack(side=tk.LEFT,padx=10)
Ent=Entry(w2,textvariable=b)
Ent.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Search Player",command=player)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Default",command=default)
bt.pack(side=tk.LEFT,padx=10)
bt=Button(w2,text="Sort by Sixes",command=sixes)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Fours",command=fours)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Wickets",command=wickets)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by DotBalls",command=dot)
bt.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Default",command=default)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Chennai Super Kings",command=CSK)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Mumbai Indians",command=MI)
but.pack(side=tk.LEFT,padx=6)
```

```
    but=Button(w3,text="Delhi Capitals",command=DC)
    but.pack(side=tk.LEFT,padx=6)
    but=Button(w3,text="Kolkata Knight Riders",command=KKR)
    but.pack(side=tk.LEFT,padx=6)
    but=Button(w3,text="Kings XI Punjab",command=KXIP)
    but.pack(side=tk.LEFT,padx=6)
    but=Button(w3,text="Rajasthan Royals",command=RR)
    but.pack(side=tk.LEFT,padx=6)
    but=Button(w3,text="Royal Challengers Bangalore",command=RCB)
    but.pack(side=tk.LEFT,padx=6)
    but=Button(w3,text="Sunrisers Hyderabad",command=SRH)
    but.pack(side=tk.LEFT,padx=6)
    ws.attributes('-fullscreen', True)
    def close():
      ws.destroy()
    exit_button = Button(ws, text="Exit", command=close)
    exit_button.pack(pady=20)
    ws.mainloop()
```

iv) 2019 Batting(Player Performance):

```
from tkinter import *
import tkinter as tk
from tkinter import ttk
import mysql.connector as pro
e=''
def BT2019():
    def update(rows):
            tree.delete(*tree.get_children())
            for i in rows:
                tree.insert('','end',values=i)
    def default():
            global e
            e=''
            ab=Ent.get()
            cur.execute("SELECT * from batting2019")
            rows=cur.fetchall()
            update(rows)
    def centuries():
            ab=Ent.get()
            cur.execute("SELECT * from batting2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' Order by Centuries DESC")
            rows=cur.fetchall()
            update(rows
    def fifties():
            ab=Ent.get()
```

```
      cur.execute("SELECT * from batting2019 where Player LIKE '%"+ab+"%' and Team LIKE '%"+e+"%' Order by Fifties DESC")
      rows=cur.fetchall()
      update(rows)
  def strate():
      ab=Ent.get()
      cur.execute("SELECT * from batting2019 where Player LIKE '%"+ab+"%' and Team LIKE '%"+e+"%' Order by StrikeRate DESC")
      rows=cur.fetchall()
      update(rows)
  def avg():
      ab=Ent.get()
      cur.execute("SELECT * from batting2019 where Player LIKE '%"+ab+"%' and Team LIKE '%"+e+"%' Order by Average DESC")
      rows=cur.fetchall()
      update(rows)
  def player():
      ab=Ent.get()
      cur.execute("SELECT * from batting2019 where Player LIKE '%"+ab+"%'")
      player1()
  def player1():
      rows=cur.fetchall()
      c=cur.rowcount
      if c==1:
        k=0
        co=[["Position", "Player","Team","Matches","Innings","Not Outs","Runs","Balls Faced","Centuries","Fifties","Strike Rate","Average"]]
        op=Tk()
        w=LabelFrame(op,text="Single player stats")
        w.pack(fill="both",expand="yes",padx=20,pady=10)
        ree = ttk.Treeview(w, column=("Attributes","Stats"), show='headings', height=12)
        ree.pack()
        ree.column("# 1", anchor=CENTER)
        ree.heading("# 1", text="Attributes")
        ree.column("# 2", anchor=CENTER)
        ree.heading("# 2", text="Stats")
        for i in rows:
          while k<=11:
            co.append(i)
            co[1]=list(co[1])
            ree.insert('','end',values=(co[0][k],co[1][k]))
            k+=1
        def close():
          op.destroy()
        exit_button = Button(op, text="Exit", command=close)
        exit_button.pack(pady=20)
      else:
          update(rows)
   def CSK():
        global e
        e='Chennai Super Kings'
        ab=Ent.get()
        cur.execute("SELECT * from batting2019 where Team LIKE '%Chennai%' and
Player LIKE '%"+ab+"%'")
        rows=cur.fetchall()
        update(rows)
   def MI():
        global e
        e='Mumbai Indians'
        ab=Ent.get()
        cur.execute("SELECT * from batting2019 where Team LIKE '%Mumbai%'and
Player LIKE '%"+ab+"%'")
        rows=cur.fetchall()
        update(rows)
   def DC():
        global e
        e='Delhi Capitals'
        ab=Ent.get()
        cur.execute("SELECT * from batting2019 where Team LIKE '%Delhi%' and
Player LIKE '%"+ab+"%'")
        rows=cur.fetchall()
        update(rows)
   def KKR():
        global e
        e='Kolkata Knight Riders'
        ab=Ent.get()
        cur.execute("SELECT * from batting2019 where Team LIKE '%Kolkata%' and
Player LIKE '%"+ab+"%'")
        rows=cur.fetchall()
        update(rows)
   def KXIP():
        global e
        e='Kings XI Punjab'
        ab=Ent.get()
        cur.execute("SELECT * from batting2019 where Team LIKE '%Punjab%' and
Player LIKE '%"+ab+"%'")
        rows=cur.fetchall()
        update(rows)
   def RR():
        global e
        e='Rajasthan Royals'
        ab=Ent.get()
        cur.execute("SELECT * from batting2019 where Team LIKE '%Rajasthan%' and
Player LIKE '%"+ab+"%'")
        rows=cur.fetchall()
```

```
            update(rows)
    def RCB():
            global e
            e='Royal Challengers Bangalore'
            ab=Ent.get()
            cur.execute("SELECT * from batting2019 where Team LIKE '%Bangalore%' and 
Player LIKE '%"+ab+"%'")
            rows=cur.fetchall()
            update(rows)
    def SRH():
            global e
            e='Sunrisers Hyderabad'
            ab=Ent.get()
            cur.execute("SELECT * from batting2019 where Team LIKE '%Hyderabad%' 
and Player LIKE '%"+ab+"%'")
            rows=cur.fetchall()
            update(rows)


    con=pro.connect(host='localhost',user='root',passwd='VSS@07',database='ipl')
    cur=con.cursor()
    ws=Tk()
    b=StringVar()
    ws.geometry("700x350")
    style = ttk.Style()
    style.theme_use('clam')
    w1=LabelFrame(ws,text="Batting Stats")
    w2=LabelFrame(ws,text="Search Player")
    w3=LabelFrame(ws,text="Sort Player")
    w1.pack(fill="both",expand="yes",padx=20,pady=10)
    w2.pack(fill="both",expand="yes",padx=20,pady=10)
    w3.pack(fill="both",expand="yes",padx=20,pady=10)
    tree = ttk.Treeview(w1, column=( "Position", 
"Player","Team","Matches","Innings","Not Outs","Runs","Balls 
Faced","Centuries","Fifties","Strike Rate","Average"), show='headings', height=30)
    tree.pack()
    tree.column("# 1", anchor=CENTER)
    tree.heading("# 1", text="Position")
    tree.column("# 2", anchor=CENTER)
    tree.heading("# 2", text="Player")
    tree.column("# 3", anchor=CENTER)
    tree.heading("# 3", text="Team")
    tree.column("# 4", anchor=CENTER)
    tree.heading("# 4", text="Matches")
    tree.column("# 5", anchor=CENTER)
    tree.heading("# 5", text="Innings")
    tree.column("# 6", anchor=CENTER)
    tree.heading("# 6", text="Not Outs")
tree.column("# 7", anchor=CENTER)
tree.heading("# 7", text="Runs")
tree.column("# 8", anchor=CENTER)
tree.heading("# 8", text="Balls Faced")
tree.column("# 9", anchor=CENTER)
tree.heading("# 9", text="Centuries")
tree.column("# 10", anchor=CENTER)
tree.heading("# 10", text="Fifties")
tree.column("# 11", anchor=CENTER)
tree.heading("# 11", text="Strike Rate")
tree.column("# 12", anchor=CENTER)
tree.heading("# 12", text="Average")
cur=con.cursor()
cur.execute("select * from batting2019")
rows=cur.fetchall()
update(rows)
Lab=Label(w2,text="Search")
Lab.pack(side=tk.LEFT,padx=10)
Ent=Entry(w2,textvariable=b)
Ent.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Search Player",command=player)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Default",command=default)
bt.pack(side=tk.LEFT,padx=10)
bt=Button(w2,text="Sort by Centuries",command=centuries)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Fifties",command=fifties)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Strike Rate",command=strate)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Average",command=avg)
bt.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Default",command=default)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Chennai Super Kings",command=CSK)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Mumbai Indians",command=MI)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Delhi Capitals",command=DC)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Kolkata Knight Riders",command=KKR)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Kings XI Punjab",command=KXIP)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Rajasthan Royals",command=RR)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Royal Challengers Bangalore",command=RCB)
but.pack(side=tk.LEFT,padx=6)
```

```
    but=Button(w3,text="Sunrisers Hyderabad",command=SRH)
    but.pack(side=tk.LEFT,padx=6)
    ws.attributes('-fullscreen', True)
    def close():
      ws.destroy()

    exit_button = Button(ws, text="Exit", command=close)
    exit_button.pack(pady=20)
    ws.mainloop()
```

v) 2019 Bowling(Player Performance):

```
from tkinter import *
import tkinter as tk
from tkinter import ttk
import mysql.connector as pro
e=''
def BO2019():
    def update(rows):
            tree.delete(*tree.get_children())
            for i in rows:
                tree.insert('','end',values=i)
    def default():
            global e
            e=''
            ab=Ent.get()
            cur.execute("SELECT * from bowling2019")
            rows=cur.fetchall()
            update(rows)
    def overs():
            ab=Ent.get()
            cur.execute("SELECT * from bowling2019 where Player LIKE '%"+ab+"%' and 
Team LIKE '%"+e+"%' Order by Overs DESC")
            rows=cur.fetchall()
            update(rows)
    def runs():
            ab=Ent.get()
            cur.execute("SELECT * from bowling2019 where Player LIKE '%"+ab+"%' and 
Team LIKE '%"+e+"%' Order by Runs DESC")
            rows=cur.fetchall()
            update(rows)
    def econ():
            ab=Ent.get()
            cur.execute("SELECT * from bowling2019 where Player LIKE '%"+ab+"%' and 
Team LIKE '%"+e+"%' Order by Econ DESC")
            rows=cur.fetchall()
            update(rows)
```

```
  def strate():
      ab=Ent.get()
      cur.execute("SELECT * from bowling2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' Order by SR DESC")
      rows=cur.fetchall()
      update(rows)
  def avg():
      ab=Ent.get()
      cur.execute("SELECT * from bowling2019 where Player LIKE '%"+ab+"%' and
Team LIKE '%"+e+"%' Order by Average DESC")
      rows=cur.fetchall()
      update(rows)
  def player():
      ab=Ent.get()
      cur.execute("SELECT * from bowling2019 where Player LIKE '%"+ab+"%'")
      player1()
  def player1():
      rows=cur.fetchall()
      c=cur.rowcount
      if c==1:
        k=0
        co=[["Position",
"Player","Team","Matches","Innings","Overs","Runs","Wickets","Economy","Strike
Rate","Average"]]
        op=Tk()
        w=LabelFrame(op,text="Single player stats")
        w.pack(fill="both",expand="yes",padx=20,pady=10)
        ree = ttk.Treeview(w, column=("Attributes","Stats"), show='headings',
height=11)
        ree.pack()
        ree.column("# 1", anchor=CENTER)
        ree.heading("# 1", text="Attributes")
        ree.column("# 2", anchor=CENTER)
        ree.heading("# 2", text="Stats")
        for i in rows:
          while k<=10:
            co.append(i)
            co[1]=list(co[1])
            ree.insert('','end',values=(co[0][k],co[1][k]))
            k+=1
        def close():
          op.destroy()
        exit_button = Button(op, text="Exit", command=close)
        exit_button.pack(pady=20)
      else:
        update(rows)
  def CSK():
      global e
```

```
    e='Chennai Super Kings'
    ab=Ent.get()
    cur.execute("SELECT * from bowling2019 where Team LIKE '%Chennai%'")
    rows=cur.fetchall()
    update(rows)
def MI():
    global e
    e='Mumbai Indians'
    ab=Ent.get()
    cur.execute("SELECT * from bowling2019 where Team LIKE '%Mumbai%'")
    rows=cur.fetchall()
    update(rows)
def DC():
    global e
    e='Delhi Capitals'
    ab=Ent.get()
    cur.execute("SELECT * from bowling2019 where Team LIKE '%Delhi%'")
    rows=cur.fetchall()
    update(rows)
def KKR():
    global e
    e='Kolkata Knight Riders'
    ab=Ent.get()
    cur.execute("SELECT * from bowling2019 where Team LIKE '%Kolkata%'")
    rows=cur.fetchall()
    update(rows)
def KXIP():
    global e
    e='Kings XI Punjab'
    ab=Ent.get()
    cur.execute("SELECT * from bowling2019 where Team LIKE '%Punjab%'")
    rows=cur.fetchall()
    update(rows)
def RR():
    global e
    e='Rajasthan Royals'
    ab=Ent.get()
    cur.execute("SELECT * from bowling2019 where Team LIKE '%Rajasthan%'")
    rows=cur.fetchall()
    update(rows)
def RCB():
    global e
    e='Royal Challengers Bangalore'
    ab=Ent.get()
    cur.execute("SELECT * from bowling2019 where Team LIKE '%Bangalore%'")
    rows=cur.fetchall()
    update(rows)
def SRH():
        global e
        e='Sunrisers Hyderabad'
        ab=Ent.get()
        cur.execute("SELECT * from bowling2019 where Team LIKE '%Hyderabad%'")
        rows=cur.fetchall()
        update(rows)



    con=pro.connect(host='localhost',user='root',passwd='VSS@07',database='ipl')
    cur=con.cursor()

    ws=Tk()
    b=StringVar()
    ws.geometry("700x350")
    style = ttk.Style()
    style.theme_use('clam')
    w1=LabelFrame(ws,text="Bowling Stats")
    w2=LabelFrame(ws,text="Search Player")
    w3=LabelFrame(ws,text="Sort Team")
    w1.pack(fill="both",expand="yes",padx=20,pady=10)
    w2.pack(fill="both",expand="yes",padx=20,pady=10)
    w3.pack(fill="both",expand="yes",padx=20,pady=10)
    tree = ttk.Treeview(w1, column=( "Position",
"Player","Team","Matches","Innings","Overs","Runs","Wickets","Economy","Strike
Rate","Average"), show='headings', height=30)
    tree.pack()
    tree.column("# 1", anchor=CENTER)
    tree.heading("# 1", text="Position")
    tree.column("# 2", anchor=CENTER)
    tree.heading("# 2", text="Player")
    tree.column("# 3", anchor=CENTER)
    tree.heading("# 3", text="Team")
    tree.column("# 4", anchor=CENTER)
    tree.heading("# 4", text="Matches")
    tree.column("# 5", anchor=CENTER)
    tree.heading("# 5", text="Innings")
    tree.column("# 6", anchor=CENTER)
    tree.heading("# 6", text="Overs")
    tree.column("# 7", anchor=CENTER)
    tree.heading("# 7", text="Runs")
    tree.column("# 8", anchor=CENTER)
    tree.heading("# 8", text="Wickets")
    tree.column("# 9", anchor=CENTER)
    tree.heading("# 9", text="Economy")
    tree.column("# 10", anchor=CENTER)
    tree.heading("# 10", text="Strike Rate")
    tree.column("# 11", anchor=CENTER)
tree.heading("# 11", text="Average")


cur=con.cursor()
cur.execute("select * from bowling2019")
rows=cur.fetchall()
update(rows)


Lab=Label(w2,text="Search")
Lab.pack(side=tk.LEFT,padx=10)
Ent=Entry(w2,textvariable=b)
Ent.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Search Player",command=player)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Default",command=default)
bt.pack(side=tk.LEFT,padx=10)
bt=Button(w2,text="Sort by Overs",command=overs)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Runs",command=runs)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Economy",command=econ)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Strike Rate",command=strate)
bt.pack(side=tk.LEFT,padx=6)
bt=Button(w2,text="Sort by Average",command=avg)
bt.pack(side=tk.LEFT,padx=6)


but=Button(w3,text="Default",command=default)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Chennai Super Kings",command=CSK)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Mumbai Indians",command=MI)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Delhi Capitals",command=DC)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Kolkata Knight Riders",command=KKR)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Kings XI Punjab",command=KXIP)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Rajasthan Royals",command=RR)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Royal Challengers Bangalore",command=RCB)
but.pack(side=tk.LEFT,padx=6)
but=Button(w3,text="Sunrisers Hyderabad",command=SRH)
but.pack(side=tk.LEFT,padx=6)
```

```
   ws.attributes('-fullscreen', True)
   def close():
     ws.destroy()

   exit_button = Button(ws, text="Exit", command=close)
   exit_button.pack(pady=20)
   ws.mainloop()
```

vi) 2019 Points Table:

```
 #WAP to accept the inputs and print the NRR(Net Run Rate) of a team(If a team gets all out, the total overs is considered to be 20 overs)
#Tuple having the data form[Runs,wickets,overs,runs,wickets,overs]. 1st half comprises the own team and 2nd half for the opponent team
#Format for points table: Position, Team , Matches, Wins, Losses ,Points, NRR
from tkinter import *
from tkinter import ttk
import math
from operator import itemgetter
def PT2021():
   CSK=[];MI=[];DC=[];KKR=[];PBKS=[];RCB=[];RR=[];SRH=[]
   CSK.append("Chennai Super Kings")
   MI.append("Mumbai Indians")
   SRH.append("Sunrises Hyderabad")
   KKR.append("Kolkata Knight Riders")
   PBKS.append("Punjab Kings")
   RR.append("Rajasthan Royals")
   RCB.append("Royal Challengers Bangalore")
   DC.append("Delhi Capitals")


   M=0;W=0;L=0

Che=[[188,7,20,190,3,18.4],[107,4,15.4,106,8,20],[188,9,20,143,9,20],[220,3,20,202,10,19.1],[191,4,20,122,9,20],[173,3,18.3,171,3,20],[218,4,20,219,6,20],[156,6,20,136,8,20],[157,4,18.1,156,6,20],[172,8,20,171,6,20],[139,4,19.4,134,7,20],[189,4,20,190,3,17.3],[136,5,20,139,7,19.4],[134,6,20,139,4,13]]

Mum=[[159,9,20,160,8,20],[152,10,20,142,7,20],[150,5,20,137,10,19.4],[137,9,20,138,4,19.1],[131,6,20,132,1,17.4],[172,3,18.3,171,4,20],[219,6,20,218,4,20],[136,8,20,156,6,20],[155,6,20,159,3,15.1],[111,10,18.1,165,6,20],[137,4,19,135,6,20],[129,8,20,132,6,19.1],[94,2,8.2,90,9,20],[235,9,20,193,8,20]]

Del=[[190,3,18.4,188,7,20],[147,8,20,150,7,19.4],[198,4,18.2,195,4,20],[138,4,19.1,137,9,20],[159,4,20,159,7,20],[170,4,20,171,5,20],[156,3,16.3,154,6,20],[167,3,17.4,166,6,20],[139,2,17.5,134,9,20],[154,6,20,121,6,20],[127,9,20,130,7,18.2],[132,6,19.1,129,8,20],[139,7,19.4,136,5,20],[164,5,20,166,3,20]]
```

```
Kol=[[187,6,20,177,5,20],[142,7,20,152,10,20],[166,8,20,204,4,20],[202,10,19.1,220,3,20],[133,9,20,134,4,18.5],[126,5,16.4,123,9,20],[154,6,20,156,3,16.3],[94,1,10,92,10,19],[159,3,15.1,155,6,20],[171,6,20,172,8,20],[130,7,18.2,127,9,20],[165,7,20,168,5,19.3],[119,4,19.4,115,8,20],[171,4,20,85,10,16.1]]

Pun=[[221,6,20,217,7,20],[106,8,20,107,4,15.4],[195,4,20,198,4,18.2],[120,10,19.4,121,1,18.4],[132,1,17.4,131,6,20],[123,9,20,126,5,16.4],[179,5,20,145,8,20],[166,6,20,167,3,17.4],[183,4,20,185,10,20],[125,7,20,120,7,20],[135,6,20,137,4,19],[168,5,19.3,165,7,20],[158,6,20,164,7,20],[139,4,13,134,6,20]]

Roy=[[160,8,20,159,9,20],[149,8,20,143,9,20],[204,4,20,166,8,20],[181,0,16.3,177,9,20],[122,9,20,191,4,20],[171,5,20,170,4,20],[145,8,20,179,5,20],[92,10,19,94,1,10],[156,6,20,157,4,18.1],[165,6,20,111,10,18.1],[153,3,17.1,149,9,20],[164,7,20,158,6,20],[137,6,20,141,7,20],[166,3,20,164,5,20]]

Raj=[[217,7,20,221,6,20],[150,7,19.4,147,8,20],[143,9,20,188,9,20],[177,9,20,181,0,16.3],[134,4,18.5,133,9,20],[171,4,20,172,3,18.3],[220,3,20,165,8,20],[185,10,20,183,4,20],[121,6,20,154,6,20],[164,5,20,167,3,18.3],[149,9,20,153,3,17.1],[190,3,17.3,189,4,20],[90,9,20,94,2,8.2],[85,10,16.1,171,4,20]]

Sun=[[177,5,20,187,6,20],[143,9,20,149,8,20],[137,10,19.4,150,5,20],[121,1,18.4,120,10,19.4],[159,7,20,159,4,20],[171,3,20,173,3,18.3],[165,8,20,220,3,20],[134,9,20,139,2,17.5],[120,7,20,125,7,20],[167,3,18.3,164,5,20],[134,7,20,139,4,19.4],[115,8,20,119,4,19.4],[141,7,20,137,6,20],[193,8,20,235,9,20]]

 #CSK
 NoR=0
 aa=[];bb=[];cc=[];dd=[];ee=[];ff=[]
 for i in Che:
     if sum(i)>0:
         M+=1
     if i[3]<i[0]:
         W+=1
     if i[3]>i[0]:
         L+=1
     a=i[0];b=i[1];c=i[2];d=i[3];e=i[4];f=i[5]
     aa.append(a)
     bb.append(b)
     cc.append(c)
     dd.append(d)
     ee.append(e)
     ff.append(f)
 A=[]
 for i in range(len(bb)):
    if bb[i] == 10:
       A.append(i)
       cc[i]=20
```

```
B=[]
for i in range(len(ee)):
  if ee[i] == 10:
    B.append(i)
    ff[i]=20
nl=[];nm=[]
g=sum(aa)
h=sum(dd)
for i in cc:
    TOTY= int(int(i)*6+round((i-int(i))*10,0))
    nm.append(TOTY)
for i in ff:
    TOTO=int(int(i)*6+round ((i-int(i))*10,0))
    nl.append(TOTO)
nn=sum(nm)/6
no=sum(nl)/6
NRR=round((g/nn)-(h/no),3)
P=W*2
sup=0
wi=0;l=0
W+=wi
L+=l
Ext=(wi*2)+P
CSK.append(M)
CSK.append(W)
CSK.append(L)
CSK.append(NoR)
CSK.append(NRR)
CSK.append(Ext)


#MI
NoR=0
gg=[];hh=[];ii=[];jj=[];kk=[];ll=[];M=0;W=0;L=0;NRR=0
for i in Mum:
   if sum(i)>0:
      M+=1
   if i[3]<i[0]:
      W+=1
   if i[3]>i[0]:
      L+=1
   g=i[0];h=i[1];ik=i[2];k=i[3];l=i[4];m=i[5]
   gg.append(g)
   hh.append(h)
   ii.append(ik)
   jj.append(k)
   kk.append(l)
   ll.append(m)
```

```
O=[]
for i in range(len(hh)):
  if hh[i] == 10:
    O.append(i)
    ii[i]=20
I=[]
for i in range(len(kk)):
  if kk[i] == 10:
    I.append(i)
    ll[i]=20
np=[];nq=[]
gh=sum(gg)
hg=sum(jj)
for i in ii:
    TOTX= int(int(i)*6+round((i-int(i))*10,0))
    np.append(TOTX)
for i in ll:
    TOTN=int(int(i)*6+round ((i-int(i))*10,0))
    nq.append(TOTN)
nr=sum(np)/6
ns=sum(nq)/6
NRR=round((gh/nr)-(hg/ns),3)
P=W*2
sup=0
wi=0;l=0
Ext=(wi*2)+P
W+=wi
L+=l
MI.append(M)
MI.append(W)
MI.append(L)
MI.append(NoR)
MI.append(NRR)
MI.append(Ext)

#DC
NoR=0
mm=[];nn=[];oo=[];pp=[];qq=[];rr=[];M=0;W=0;L=0;NRR=0
for i in Del:
   if sum(i)>0:
      M+=1
   if i[3]<i[0]:
      W+=1
   if i[3]>i[0]:
      L+=1
   g=i[0];h=i[1];ik=i[2];k=i[3];l=i[4];m=i[5]
   mm.append(g)
   nn.append(h)
```

```
    oo.append(ik)
    pp.append(k)
    qq.append(l)
    rr.append(m)
O=[]
for i in range(len(nn)):
   if nn[i] == 10:
      O.append(i)
      oo[i]=20
I=[]
for i in range(len(qq)):
   if qq[i] == 10:
      I.append(i)
      rr[i]=20
np=[];nq=[]
gh=sum(mm)
hg=sum(pp)
for i in oo:
     TOTX= int(int(i)*6+round((i-int(i))*10,0))
     np.append(TOTX)
for i in rr:
     TOTN=int(int(i)*6+round ((i-int(i))*10,0))
     nq.append(TOTN)
nr=sum(np)/6
ns=sum(nq)/6
NRR=round((gh/nr)-(hg/ns),3)
P=W*2
sup=1
wi=1;l=0
L+=l
W+=wi
Ext=(wi*2)+P
DC.append(M)
DC.append(W)
DC.append(L)
DC.append(NoR)
DC.append(NRR)
DC.append(Ext)
NR=0
#KKR
NoR=0
gg=[];hh=[];ii=[];jj=[];kk=[];ll=[];M=0;W=0;L=0;NRR=0
for i in Kol:
    if sum(i)>0:
        M+=1
    if i[3]<i[0]:
        W+=1
    if i[3]>i[0]:
        L+=1
    g=i[0];h=i[1];ik=i[2];k=i[3];l=i[4];m=i[5]
    gg.append(g)
    hh.append(h)
    ii.append(ik)
    jj.append(k)
    kk.append(l)
    ll.append(m)
O=[]
for i in range(len(hh)):
   if hh[i] == 10:
      O.append(i)
      ii[i]=20
I=[]
for i in range(len(kk)):
   if kk[i] == 10:
      I.append(i)
      ll[i]=20
np=[];nq=[]
gh=sum(gg)
hg=sum(jj)
for i in ii:
     TOTX= int(int(i)*6+round((i-int(i))*10,0))
     np.append(TOTX)
for i in ll:
     TOTN=int(int(i)*6+round ((i-int(i))*10,0))
     nq.append(TOTN)
nr=sum(np)/6
ns=sum(nq)/6
NRR=round((gh/nr)-(hg/ns),3)
P=W*2
sup=0
wi=0;l=0
L+=l
W+=wi
Ext=(wi*2)+P
KKR.append(M)
KKR.append(W)
KKR.append(L)
KKR.append(NoR)
KKR.append(NRR)
KKR.append(Ext)

#PBKS
NoR=0
gg=[];hh=[];ii=[];jj=[];kk=[];ll=[];M=0;W=0;L=0;NRR=0
for i in Pun:
    if sum(i)>0:
        M+=1
    if i[3]<i[0]:
        W+=1
    if i[3]>i[0]:
        L+=1
    g=i[0];h=i[1];ik=i[2];k=i[3];l=i[4];m=i[5]
    gg.append(g)
    hh.append(h)
    ii.append(ik)
    jj.append(k)
    kk.append(l)
    ll.append(m)
O=[]
for i in range(len(hh)):
   if hh[i] == 10:
      O.append(i)
      ii[i]=20
l=[]
for i in range(len(kk)):
   if kk[i] == 10:
      l.append(i)
      ll[i]=20
np=[];nq=[]
gh=sum(gg)
hg=sum(jj)
for i in ii:
     TOTX= int(int(i)*6+round((i-int(i))*10,0))
     np.append(TOTX)
for i in ll:
     TOTN=int(int(i)*6+round ((i-int(i))*10,0))
     nq.append(TOTN)
nr=sum(np)/6
ns=sum(nq)/6
NRR=round((gh/nr)-(hg/ns),3)
P=W*2
sup=0;l=0
L+=l
wi=0
W+=wi
Ext=(wi*2)+P
PBKS.append(M)
PBKS.append(W)
PBKS.append(L)
PBKS.append(NoR)
PBKS.append(NRR)
PBKS.append(Ext)

#RCB
```

```
NoR=0
gg=[];hh=[];ii=[];jj=[];kk=[];ll=[];M=0;W=0;L=0;NRR=0
for i in Roy:
    if sum(i)>0:
        M+=1
    if i[3]<i[0]:
        W+=1
    if i[3]>i[0]:
        L+=1
    g=i[0];h=i[1];ik=i[2];k=i[3];l=i[4];m=i[5]
    gg.append(g)
    hh.append(h)
    ii.append(ik)
    jj.append(k)
    kk.append(l)
    ll.append(m)
O=[]
for i in range(len(hh)):
   if hh[i] == 10:
      O.append(i)
      ii[i]=20
I=[]
for i in range(len(kk)):
   if kk[i] == 10:
      I.append(i)
      ll[i]=20
np=[];nq=[]
gh=sum(gg)
hg=sum(jj)
for i in ii:
     TOTX= int(int(i)*6+round((i-int(i))*10,0))
     np.append(TOTX)
for i in ll:
     TOTN=int(int(i)*6+round ((i-int(i))*10,0))
     nq.append(TOTN)
nr=sum(np)/6
ns=sum(nq)/6
NRR=round((gh/nr)-(hg/ns),3)
P=W*2
sup=0
wi=0;l=0
L+=l
W+=wi
Ext=(wi*2)+P
RCB.append(M)
RCB.append(W)
RCB.append(L)
RCB.append(NoR)
RCB.append(NRR)
RCB.append(Ext)


#RR
NoR=0
gg=[];hh=[];ii=[];jj=[];kk=[];ll=[];M=0;W=0;L=0;NRR=0
for i in Raj:
    if sum(i)>0:
        M+=1
    if i[3]<i[0]:
        W+=1
    if i[3]>i[0]:
        L+=1
    g=i[0];h=i[1];ik=i[2];k=i[3];l=i[4];m=i[5]
    gg.append(g)
    hh.append(h)
    ii.append(ik)
    jj.append(k)
    kk.append(l)
    ll.append(m)
O=[]
for i in range(len(hh)):
   if hh[i] == 10:
      O.append(i)
      ii[i]=20
I=[]
for i in range(len(kk)):
   if kk[i] == 10:
      I.append(i)
      ll[i]=20
np=[];nq=[]
gh=sum(gg)
hg=sum(jj)
for i in ii:
     TOTX= int(int(i)*6+round((i-int(i))*10,0))
     np.append(TOTX)
for i in ll:
     TOTN=int(int(i)*6+round ((i-int(i))*10,0))
     nq.append(TOTN)
nr=sum(np)/6
ns=sum(nq)/6
NRR=round((gh/nr)-(hg/ns),3)
P=W*2
sup=0
wi=0;l=0
L+=l
W+=wi
```

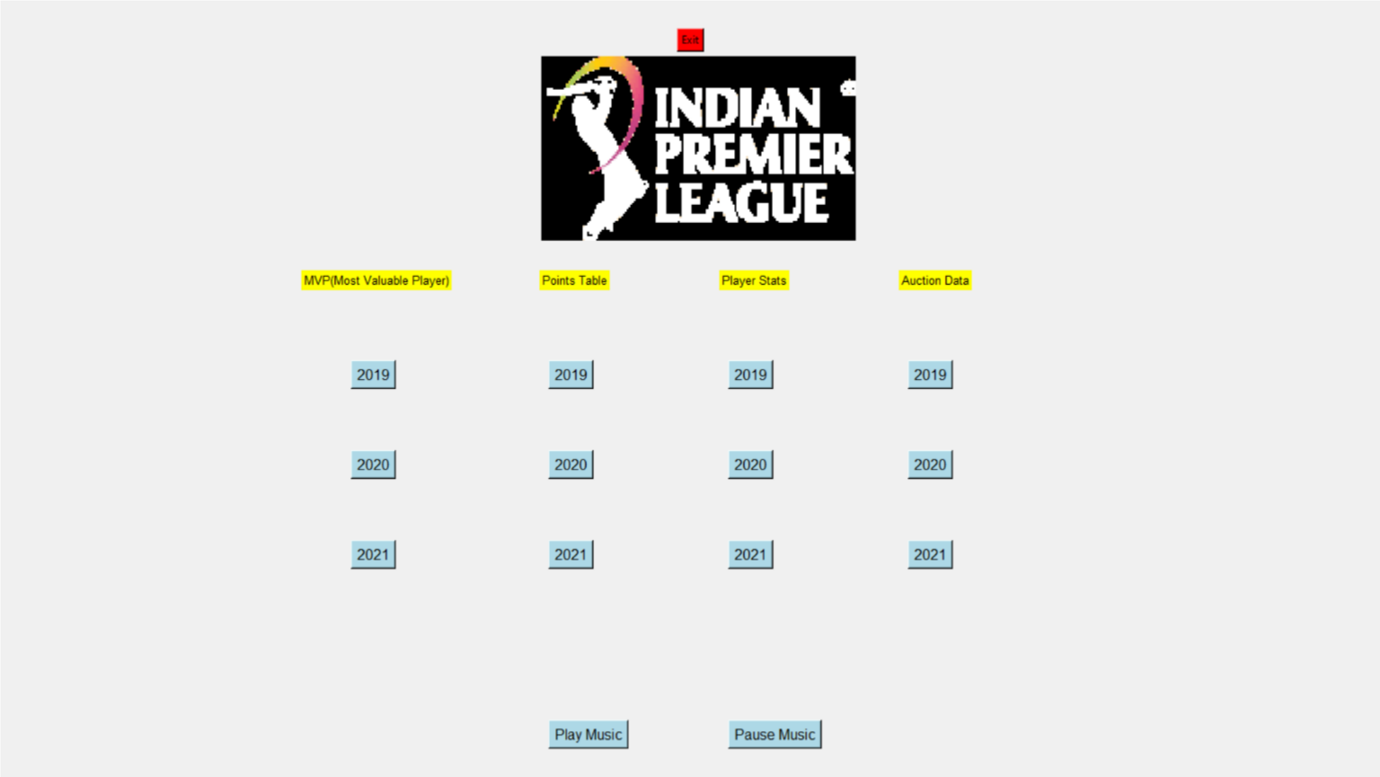
```
Ext=(wi*2)+P
RR.append(M)
RR.append(W)
RR.append(L)
RR.append(NoR)
RR.append(NRR)
RR.append(Ext)


#SRH
NoR=0
gg=[];hh=[];ii=[];jj=[];kk=[];ll=[];M=0;W=0;L=0;NRR=0
for i in Sun:
    if sum(i)>0:
        M+=1
    if i[3]<i[0]:
        W+=1
    if i[3]>i[0]:
        L+=1
    g=i[0];h=i[1];ik=i[2];k=i[3];l=i[4];m=i[5]
    gg.append(g)
    hh.append(h)
    ii.append(ik)
    jj.append(k)
    kk.append(l)
    ll.append(m)
O=[]
for i in range(len(hh)):
   if hh[i] == 10:
      O.append(i)
      ii[i]=20
I=[]
for i in range(len(kk)):
   if kk[i] == 10:
      I.append(i)
      ll[i]=20
np=[];nq=[]
gh=sum(gg)
hg=sum(jj)
for i in ii:
     TOTX= int(int(i)*6+round((i-int(i))*10,0))
     np.append(TOTX)
for i in ll:
     TOTN=int(int(i)*6+round ((i-int(i))*10,0))
     nq.append(TOTN)
nr=sum(np)/6
ns=sum(nq)/6
NRR=round((gh/nr)-(hg/ns),3)
```

```
 P=W*2
 sup=1
 wi=0;l=1
 L+=l
 W+=wi
 Ext=(wi*2)+P
 SRH.append(M)
 SRH.append(W)
 SRH.append(L)
 SRH.append(NoR)
 SRH.append(NRR)
 SRH.append(Ext)





 OVR=[]
 OVR.append(CSK)
 OVR.append(DC)
 OVR.append(KKR)
 OVR.append(MI)
 OVR.append(PBKS)
 OVR.append(RCB)
 OVR.append(RR)
 OVR.append(SRH)

 abc=sorted(OVR, key=itemgetter(6,5),reverse=True)
 Pos=[1,2,3,4,5,6,7,8]
 for i in range(1,9):
   abc[i-1].insert(0,i)

 win = Tk()
 win.geometry("700x350")
 style = ttk.Style()
 style.theme_use('clam')
 tree = ttk.Treeview(win, column=("Pos", "Team", "M","W","L","N/R","NRR","Pts"),
show='headings', height=8)
 tree.column("# 1", anchor=CENTER)
 tree.heading("# 1", text="Pos")
 tree.column("# 2", anchor=CENTER)
 tree.heading("# 2", text="Team")
 tree.column("# 3", anchor=CENTER)
 tree.heading("# 3", text="M")
 tree.column("# 4", anchor=CENTER)
 tree.heading("# 4", text="W")
 tree.column("# 5", anchor=CENTER)
 tree.heading("# 5", text="L")
```

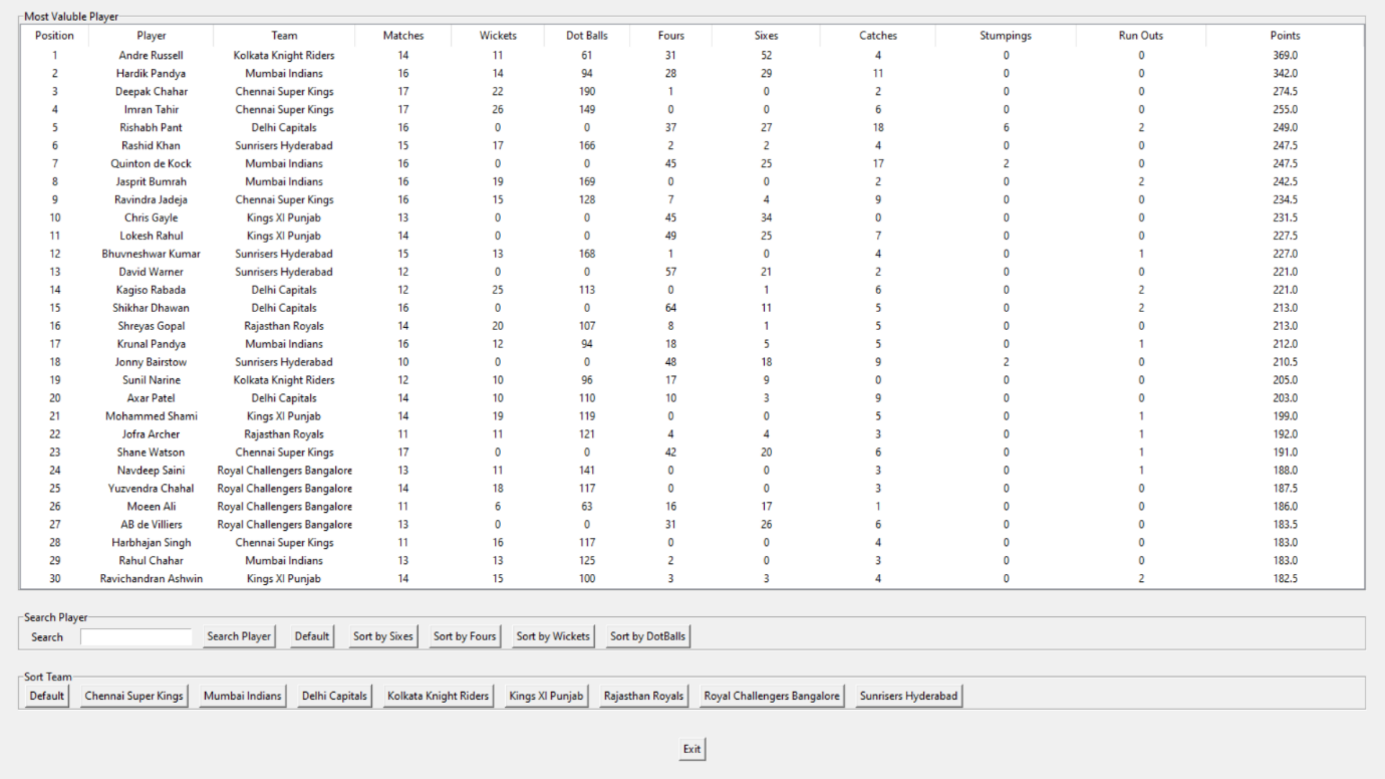
```
  tree.column("# 6", anchor=CENTER)
  tree.heading("# 6", text="N/R")
  tree.column("# 7", anchor=CENTER)
  tree.heading("# 7", text="NRR")
  tree.column("# 8", anchor=CENTER)
  tree.heading("# 8", text="Pts")

  def close():
   win.destroy()

  exit_button = Button(win, text="Exit", command=close)
  exit_button.pack(pady=20)

  tree.insert('', 'end', text="1", values=(abc[0][0],
abc[0][1],abc[0][2],abc[0][3],abc[0][4],abc[0][5],abc[0][6],abc[0][7] ))
  tree.insert('', 'end', text="1", values=(abc[1][0],
abc[1][1],abc[1][2],abc[1][3],abc[1][4],abc[1][5],abc[1][6],abc[1][7] ))
  tree.insert('', 'end', text="1", values=(abc[2][0],
abc[2][1],abc[2][2],abc[2][3],abc[2][4],abc[2][5],abc[2][6],abc[2][7] ))
  tree.insert('', 'end', text="1", values=(abc[3][0],
abc[3][1],abc[3][2],abc[3][3],abc[3][4],abc[3][5],abc[3][6],abc[3][7] ))
  tree.insert('', 'end', text="1", values=(abc[4][0],
abc[4][1],abc[4][2],abc[4][3],abc[4][4],abc[4][5],abc[4][6],abc[4][7] ))
  tree.insert('', 'end', text="1", values=(abc[5][0],
abc[5][1],abc[5][2],abc[5][3],abc[5][4],abc[5][5],abc[5][6],abc[5][7] ))
  tree.insert('', 'end', text="1", values=(abc[6][0],
abc[6][1],abc[6][2],abc[6][3],abc[6][4],abc[6][5],abc[6][6],abc[6][7] ))
  tree.insert('', 'end', text="1", values=(abc[7][0],
abc[7][1],abc[7][2],abc[7][3],abc[7][4],abc[7][5],abc[7][6],abc[7][7] ))

  tree.pack()

  win.mainloop()
```

OUTPUT

Main interface:-



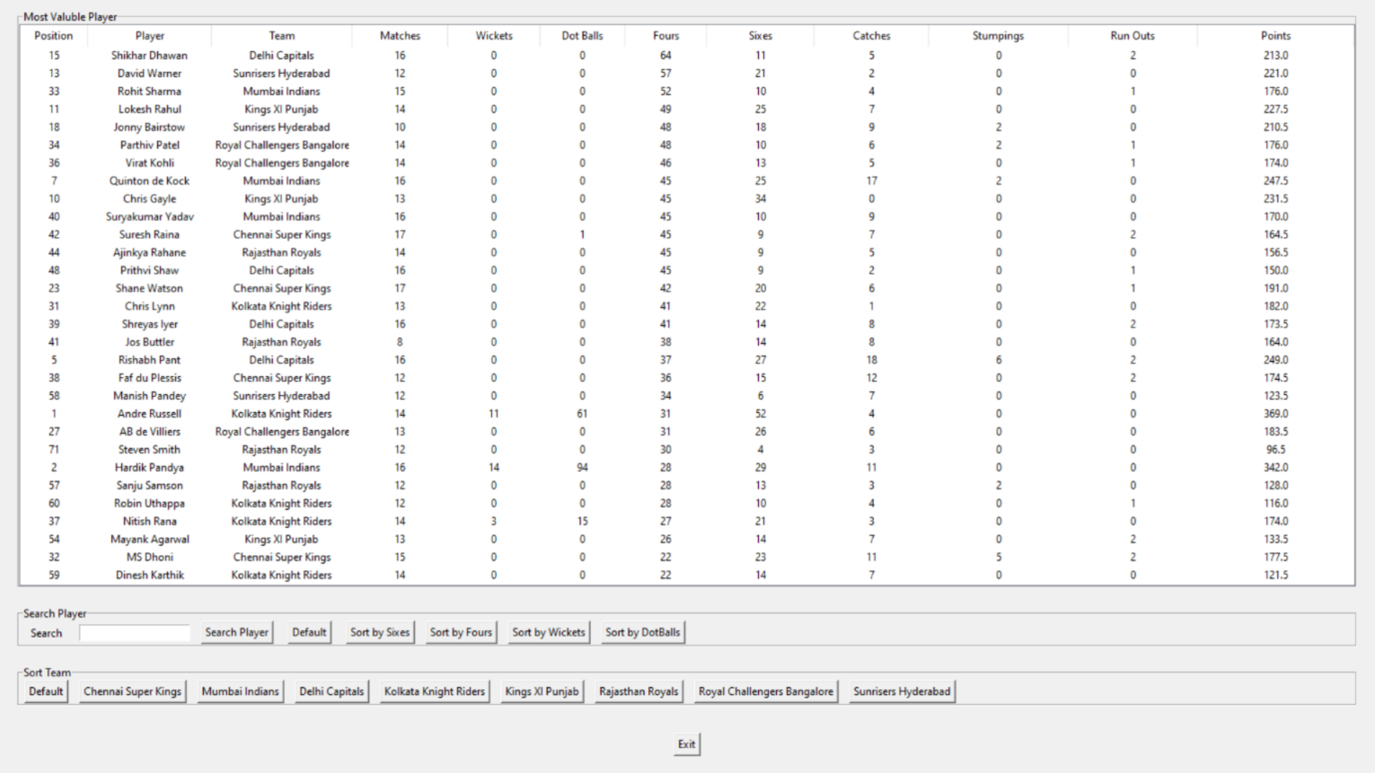
i) MVP 2019:-



Most Valuble Player

Position	Player	Team	Matches	Wickets	Dot Balls	Fours	Sixes	Catches	Stumpings	Run Outs	Points
1	Andre Russell	Kolkata Knight Riders	14	11	61	31	52	4	0	0	369.0
2	Hardik Pandya	Mumbai Indians	16	14	94	28	29	11	0	0	342.0
3	Deepak Chahar	Chennai Super Kings	17	22	190	1	0	2	0	0	274.5
4	Imran Tahir	Chennai Super Kings	17	26	149	0	0	6	0	0	255.0
5	Rishabh Pant	Delhi Capitals	16	0	0	37	27	18	6	2	249.0
6	Rashid Khan	Sunrisers Hyderabad	15	17	166	2	2	4	0	0	247.5
7	Quinton de Kock	Mumbai Indians	16	0	0	45	25	17	2	0	247.5
8	Jasprit Bumrah	Mumbai Indians	16	19	169	0	0	2	0	2	242.5
9	Ravindra Jadeja	Chennai Super Kings	16	15	128	7	4	9	0	0	234.5
10	Chris Gayle	Kings XI Punjab	13	0	0	45	34	0	0	0	231.5
11	Lokesh Rahul	Kings XI Punjab	14	0	0	49	25	7	0	0	227.5
12	Bhuvneshwar Kumar	Sunrisers Hyderabad	15	13	168	1	0	4	0	1	227.0
13	David Warner	Sunrisers Hyderabad	12	0	0	57	21	2	0	0	221.0
14	Kagiso Rabada	Delhi Capitals	12	25	113	0	1	6	0	2	221.0
15	Shikhar Dhawan	Delhi Capitals	16	0	0	64	11	5	0	2	213.0
16	Shreyas Gopal	Rajasthan Royals	14	20	107	8	1	5	0	0	213.0
17	Krunal Pandya	Mumbai Indians	16	12	94	18	5	5	0	1	212.0
18	Jonny Bairstow	Sunrisers Hyderabad	10	0	0	48	18	9	2	0	210.5
19	Sunil Narine	Kolkata Knight Riders	12	10	96	17	9	0	0	0	205.0
20	Axar Patel	Delhi Capitals	14	10	110	10	3	9	0	0	203.0
21	Mohammed Shami	Kings XI Punjab	14	19	119	0	0	5	0	1	199.0
22	Jofra Archer	Rajasthan Royals	11	11	121	4	4	3	0	1	192.0
23	Shane Watson	Chennai Super Kings	17	0	0	42	20	6	0	1	191.0
24	Navdeep Saini	Royal Challengers Bangalore	13	11	141	0	0	3	0	1	188.0
25	Yuzvendra Chahal	Royal Challengers Bangalore	14	18	117	0	0	3	0	0	187.5
26	Moeen Ali	Royal Challengers Bangalore	11	6	63	16	17	1	0	0	186.0
27	AB de Villiers	Royal Challengers Bangalore	13	0	0	31	26	6	0	0	183.5
28	Harbhajan Singh	Chennai Super Kings	11	16	117	0	0	4	0	0	183.0
29	Rahul Chahar	Mumbai Indians	13	13	125	2	0	3	0	0	183.0
30	Ravichandran Ashwin	Kings XI Punjab	14	15	100	3	3	4	0	2	182.5

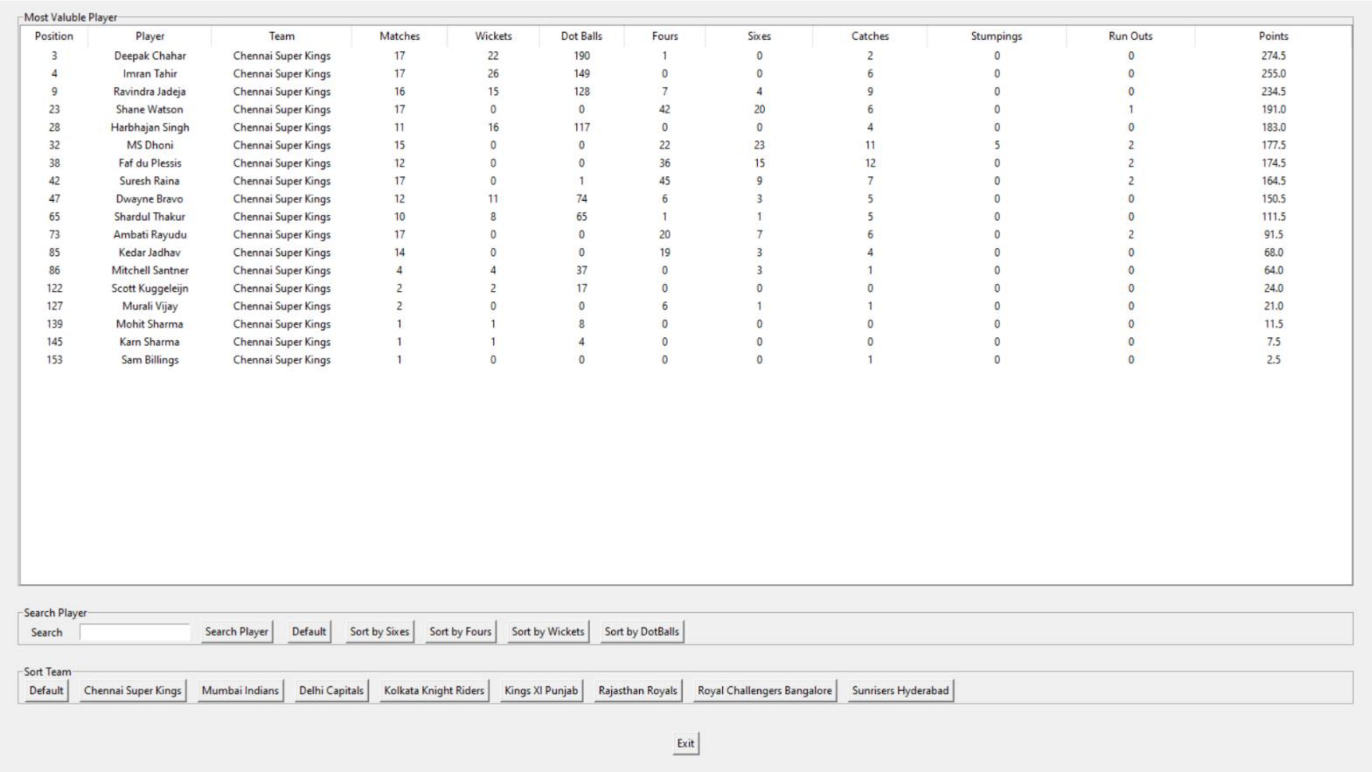
MVP 2019(sorted by Fours):-



Most Valuble Player

Position	Player	Team	Matches	Wickets	Dot Balls	Fours	Sixes	Catches	Stumpings	Run Outs	Points
15	Shikhar Dhawan	Delhi Capitals	16	0	0	64	11	5	0	2	213.0
13	David Warner	Sunrisers Hyderabad	12	0	0	57	21	2	0	0	221.0
33	Rohit Sharma	Mumbai Indians	15	0	0	52	10	4	0	1	176.0
11	Lokesh Rahul	Kings XI Punjab	14	0	0	49	25	7	0	0	227.5
18	Jonny Bairstow	Sunrisers Hyderabad	10	0	0	48	18	9	2	0	210.5
34	Parthiv Patel	Royal Challengers Bangalore	14	0	0	48	10	6	2	1	176.0
36	Virat Kohli	Royal Challengers Bangalore	14	0	0	46	13	5	0	1	174.0
7	Quinton de Kock	Mumbai Indians	16	0	0	45	25	17	2	0	247.5
10	Chris Gayle	Kings XI Punjab	13	0	0	45	34	0	0	0	231.5
40	Suryakumar Yadav	Mumbai Indians	16	0	0	45	10	9	0	0	170.0
42	Suresh Raina	Chennai Super Kings	17	0	1	45	9	7	0	2	164.5
44	Ajinkya Rahane	Rajasthan Royals	14	0	0	45	9	5	0	0	156.5
48	Prithvi Shaw	Delhi Capitals	16	0	0	45	9	2	0	1	150.0
23	Shane Watson	Chennai Super Kings	17	0	0	42	20	6	0	1	191.0
31	Chris Lynn	Kolkata Knight Riders	13	0	0	41	22	1	0	0	182.0
39	Shreyas Iyer	Delhi Capitals	16	0	0	41	14	8	0	2	173.5
41	Jos Buttler	Rajasthan Royals	8	0	0	38	14	8	0	0	164.0
5	Rishabh Pant	Delhi Capitals	16	0	0	37	27	18	6	2	249.0
38	Faf du Plessis	Chennai Super Kings	12	0	0	36	15	12	0	2	174.5
58	Manish Pandey	Sunrisers Hyderabad	12	0	0	34	6	7	0	0	123.5
1	Andre Russell	Kolkata Knight Riders	14	11	61	31	52	4	0	0	369.0
27	AB de Villiers	Royal Challengers Bangalore	13	0	0	31	26	6	0	0	183.5
71	Steven Smith	Rajasthan Royals	12	0	0	30	4	3	0	0	96.5
2	Hardik Pandya	Mumbai Indians	16	14	94	28	29	11	0	0	342.0
57	Sanju Samson	Rajasthan Royals	12	0	0	28	13	3	2	0	128.0
60	Robin Uthappa	Kolkata Knight Riders	12	0	0	28	10	4	0	1	116.0
37	Nitish Rana	Kolkata Knight Riders	14	3	15	27	21	3	0	0	174.0
54	Mayank Agarwal	Kings XI Punjab	13	0	0	26	14	7	0	2	133.5
32	MS Dhoni	Chennai Super Kings	15	0	0	22	23	11	5	2	177.5
59	Dinesh Karthik	Kolkata Knight Riders	14	0	0	22	14	7	0	0	121.5

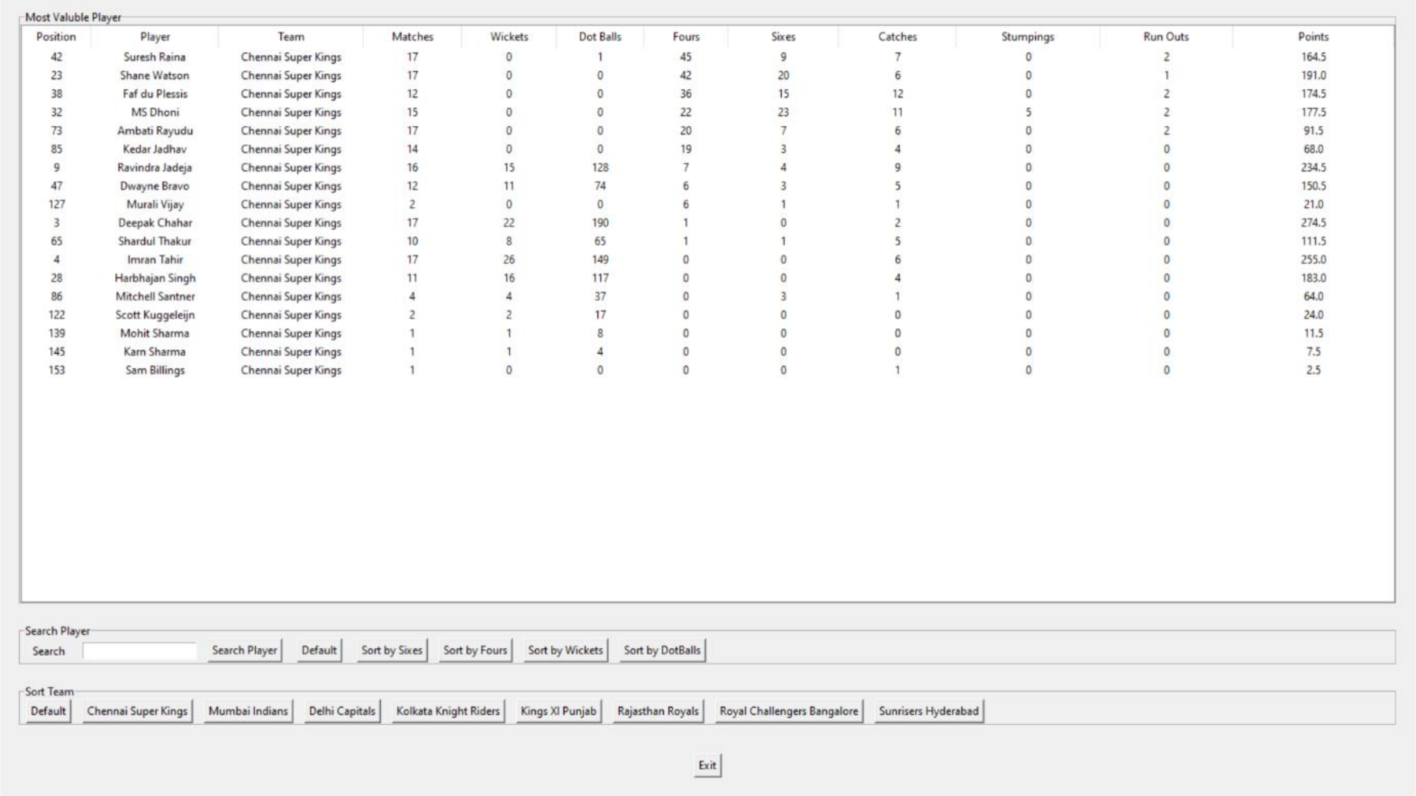
MVP 2019(sorted by Team):-



Most Valuble Player

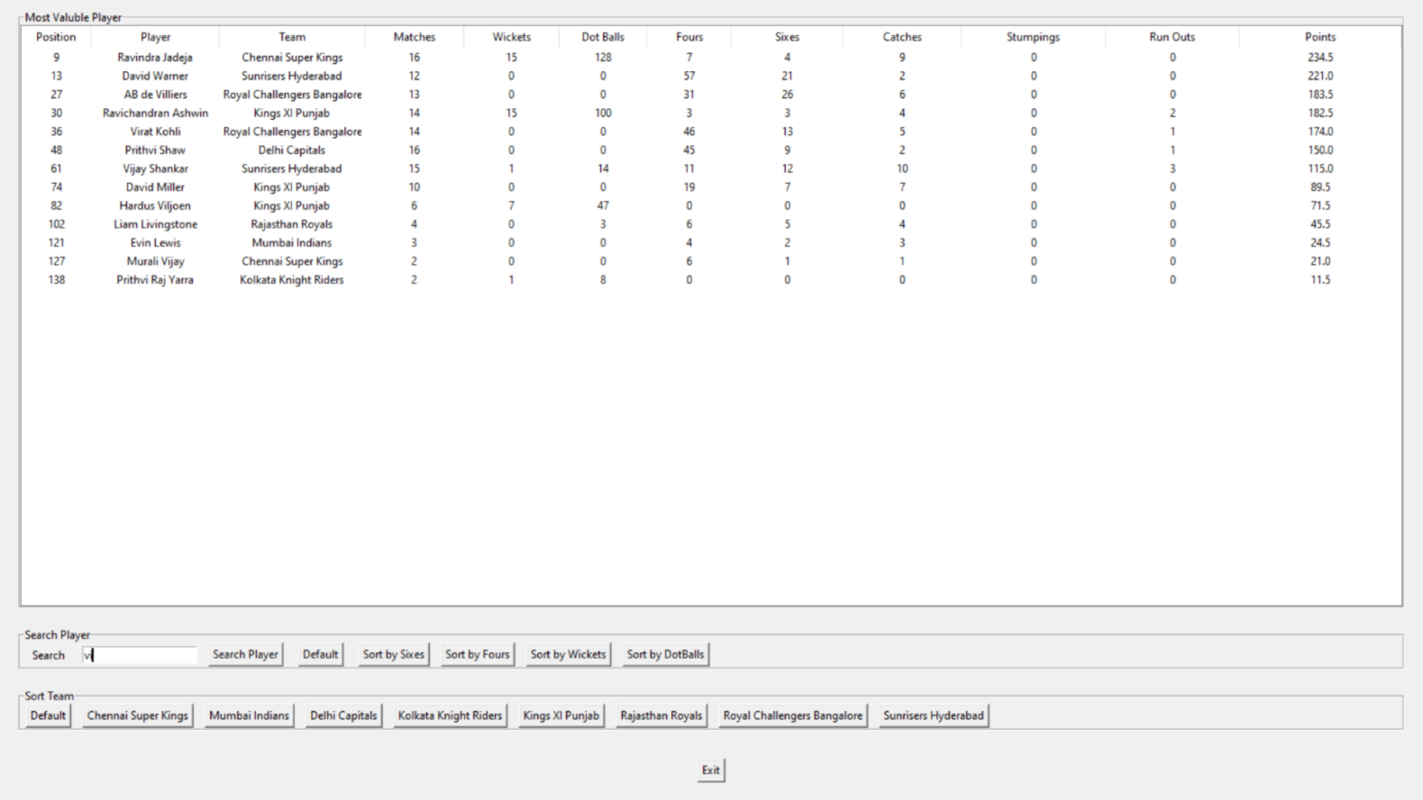
Position	Player	Team	Matches	Wickets	Dot Balls	Fours	Sixes	Catches	Stumpings	Run Outs	Points
3	Deepak Chahar	Chennai Super Kings	17	22	190	1	0	2	0	0	274.5
4	Imran Tahir	Chennai Super Kings	17	26	149	0	0	6	0	0	255.0
9	Ravindra Jadeja	Chennai Super Kings	16	15	128	7	4	9	0	0	234.5
23	Shane Watson	Chennai Super Kings	17	0	0	42	20	6	0	1	191.0
28	Harbhajan Singh	Chennai Super Kings	11	16	117	0	0	4	0	0	183.0
32	MS Dhoni	Chennai Super Kings	15	0	0	22	23	11	5	2	177.5
38	Faf du Plessis	Chennai Super Kings	12	0	0	36	15	12	0	2	174.5
42	Suresh Raina	Chennai Super Kings	17	0	1	45	9	7	0	2	164.5
47	Dwayne Bravo	Chennai Super Kings	12	11	74	6	3	5	0	0	150.5
65	Shardul Thakur	Chennai Super Kings	10	8	65	1	1	5	0	0	111.5
73	Ambati Rayudu	Chennai Super Kings	17	0	0	20	7	6	0	2	91.5
85	Kedar Jadhav	Chennai Super Kings	14	0	0	19	3	4	0	0	68.0
86	Mitchell Santner	Chennai Super Kings	4	4	37	0	3	1	0	0	64.0
122	Scott Kuggeleijn	Chennai Super Kings	2	2	17	0	0	0	0	0	24.0
127	Murali Vijay	Chennai Super Kings	2	0	0	6	1	1	0	0	21.0
139	Mohit Sharma	Chennai Super Kings	1	1	8	0	0	0	0	0	11.5
145	Karn Sharma	Chennai Super Kings	1	1	4	0	0	0	0	0	7.5
153	Sam Billings	Chennai Super Kings	1	0	0	0	0	1	0	0	2.5

MVP 2019(sorted by Team and a column):-



Position	Player	Team	Matches	Wickets	Dot Balls	Fours	Sixes	Catches	Stumpings	Run Outs	Points
42	Suresh Raina	Chennai Super Kings	17	0	1	45	9	7	0	2	164.5
23	Shane Watson	Chennai Super Kings	17	0	0	42	20	6	0	1	191.0
38	Faf du Plessis	Chennai Super Kings	12	0	0	36	15	12	0	2	174.5
32	MS Dhoni	Chennai Super Kings	15	0	0	22	23	11	5	2	177.5
73	Ambati Rayudu	Chennai Super Kings	17	0	0	20	7	6	0	2	91.5
85	Kedar Jadhav	Chennai Super Kings	14	0	0	19	3	4	0	0	68.0
9	Ravindra Jadeja	Chennai Super Kings	16	15	128	7	4	9	0	0	234.5
47	Dwayne Bravo	Chennai Super Kings	12	11	74	6	3	5	0	0	150.5
127	Murali Vijay	Chennai Super Kings	2	0	0	6	1	1	0	0	21.0
3	Deepak Chahar	Chennai Super Kings	17	22	190	1	0	2	0	0	274.5
65	Shardul Thakur	Chennai Super Kings	10	8	65	1	1	5	0	0	111.5
4	Imran Tahir	Chennai Super Kings	17	26	149	0	0	6	0	0	255.0
28	Harbhajan Singh	Chennai Super Kings	11	16	117	0	0	4	0	0	183.0
86	Mitchell Santner	Chennai Super Kings	4	4	37	0	3	1	0	0	64.0
122	Scott Kuggeleijn	Chennai Super Kings	2	2	17	0	0	0	0	0	24.0
139	Mohit Sharma	Chennai Super Kings	1	1	8	0	0	0	0	0	11.5
145	Karn Sharma	Chennai Super Kings	1	1	4	0	0	0	0	0	7.5
153	Sam Billings	Chennai Super Kings	1	0	0	0	0	1	0	0	2.5

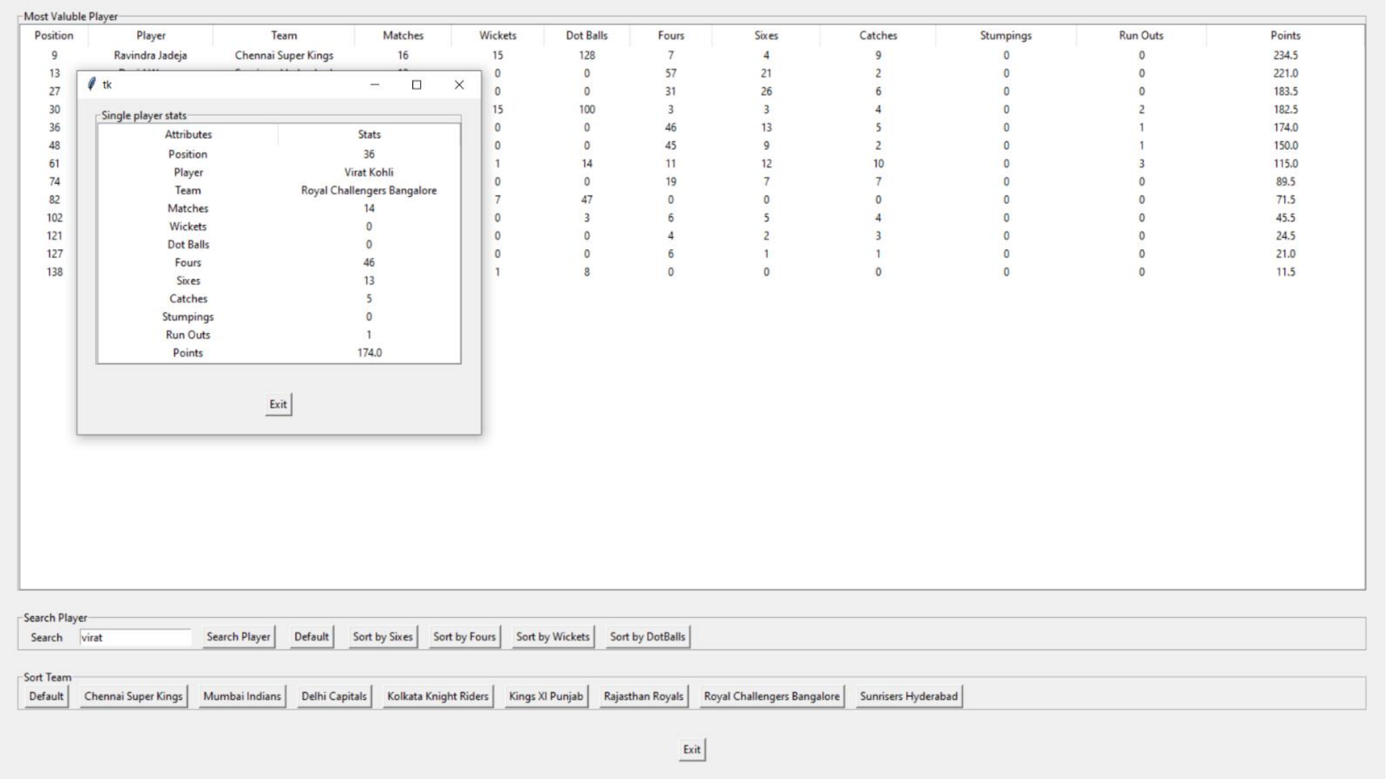
MVP 2019(search by keyword):-



Search vi

Position	Player	Team	Matches	Wickets	Dot Balls	Fours	Sixes	Catches	Stumpings	Run Outs	Points
9	Ravindra Jadeja	Chennai Super Kings	16	15	128	7	4	9	0	0	234.5
13	David Warner	Sunrisers Hyderabad	12	0	0	57	21	2	0	0	221.0
27	AB de Villiers	Royal Challengers Bangalore	13	0	0	31	26	6	0	0	183.5
30	Ravichandran Ashwin	Kings XI Punjab	14	15	100	3	3	4	0	2	182.5
36	Virat Kohli	Royal Challengers Bangalore	14	0	0	46	13	5	0	1	174.0
48	Prithvi Shaw	Delhi Capitals	16	0	0	45	9	2	0	1	150.0
61	Vijay Shankar	Sunrisers Hyderabad	15	1	14	11	12	10	0	3	115.0
74	David Miller	Kings XI Punjab	10	0	0	19	7	7	0	0	89.5
82	Hardus Viljoen	Kings XI Punjab	6	7	47	0	0	0	0	0	71.5
102	Liam Livingstone	Rajasthan Royals	4	0	3	6	5	4	0	0	45.5
121	Evin Lewis	Mumbai Indians	3	0	0	4	2	3	0	0	24.5
127	Murali Vijay	Chennai Super Kings	2	0	0	6	1	1	0	0	21.0
138	Prithvi Raj Yarra	Kolkata Knight Riders	2	1	8	0	0	0	0	0	11.5

MVP 2019(search by single player):-



Most Valuble Player

Position	Player	Team	Matches	Wickets	Dot Balls	Fours	Sixes	Catches	Stumpings	Run Outs	Points
9	Ravindra Jadeja	Chennai Super Kings	16	15	128	7	4	9	0	0	234.5
13				0	0	57	21	2	0	0	221.0
27				0	0	31	26	6	0	0	183.5
30				15	100	3	3	4	0	2	182.5
36				0	0	46	13	5	0	1	174.0
48				0	0	45	9	2	0	1	150.0
61				1	14	11	12	10	0	3	115.0
74				0	0	19	7	7	0	0	89.5
82				7	47	0	0	0	0	0	71.5
102				0	3	6	5	4	0	0	45.5
121				0	0	4	2	3	0	0	24.5
127				0	0	6	1	1	0	0	21.0
138				1	8	0	0	0	0	0	11.5

tk

Single player stats

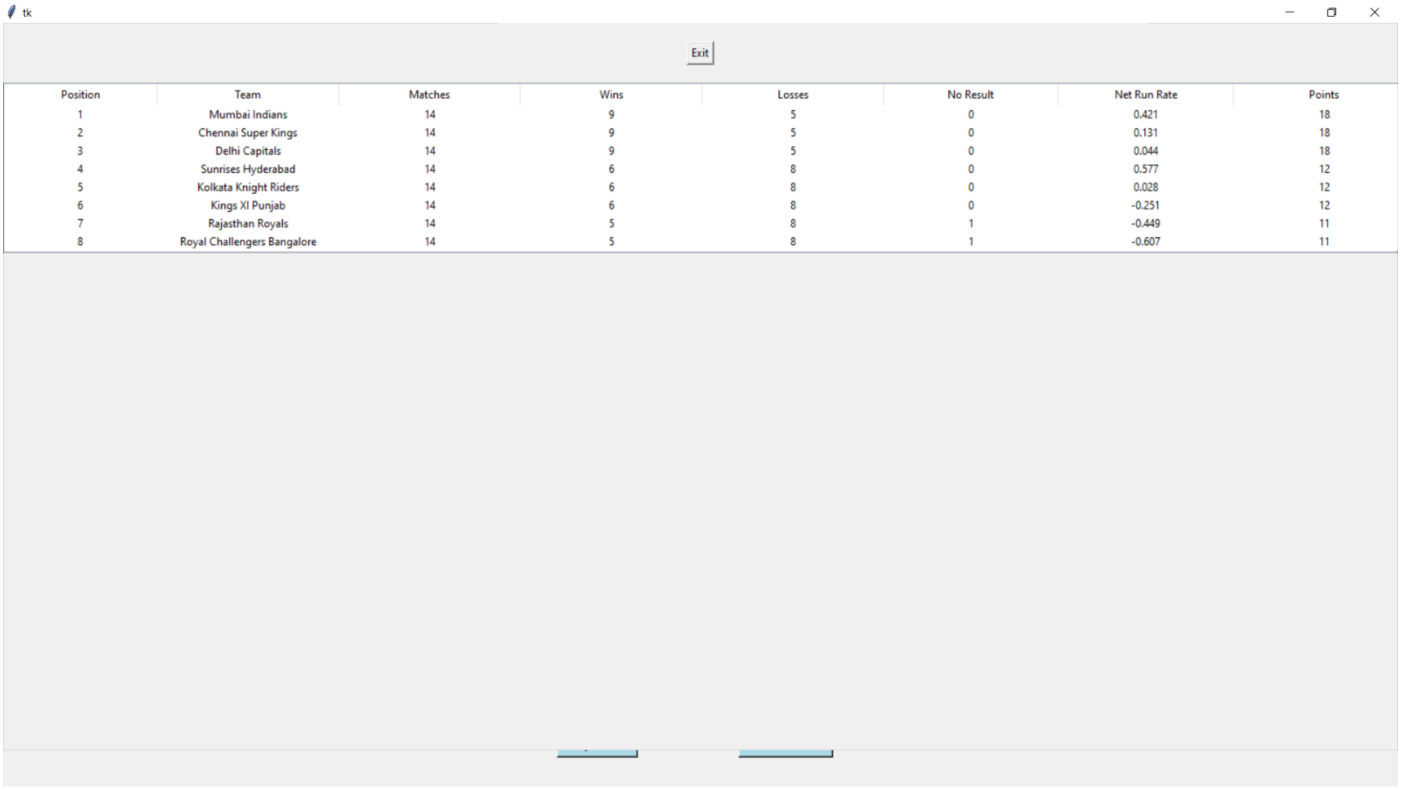
Attributes	Stats
Position	36
Player	Virat Kohli
Team	Royal Challengers Bangalore
Matches	14
Wickets	0
Dot Balls	0
Fours	46
Sixes	13
Catches	5
Stumpings	0
Run Outs	1
Points	174.0

Search Player

Search: virat

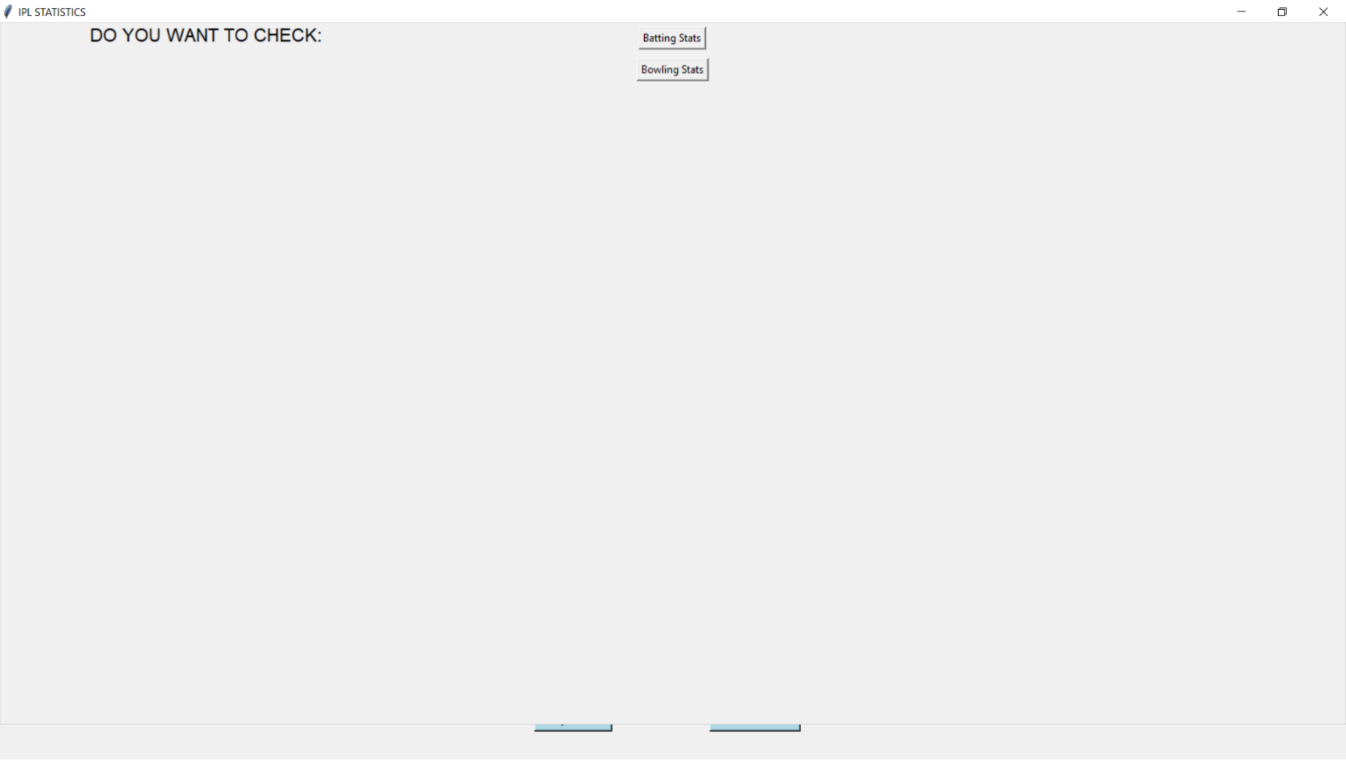
Sort Team

ii) Points Table 2019:-

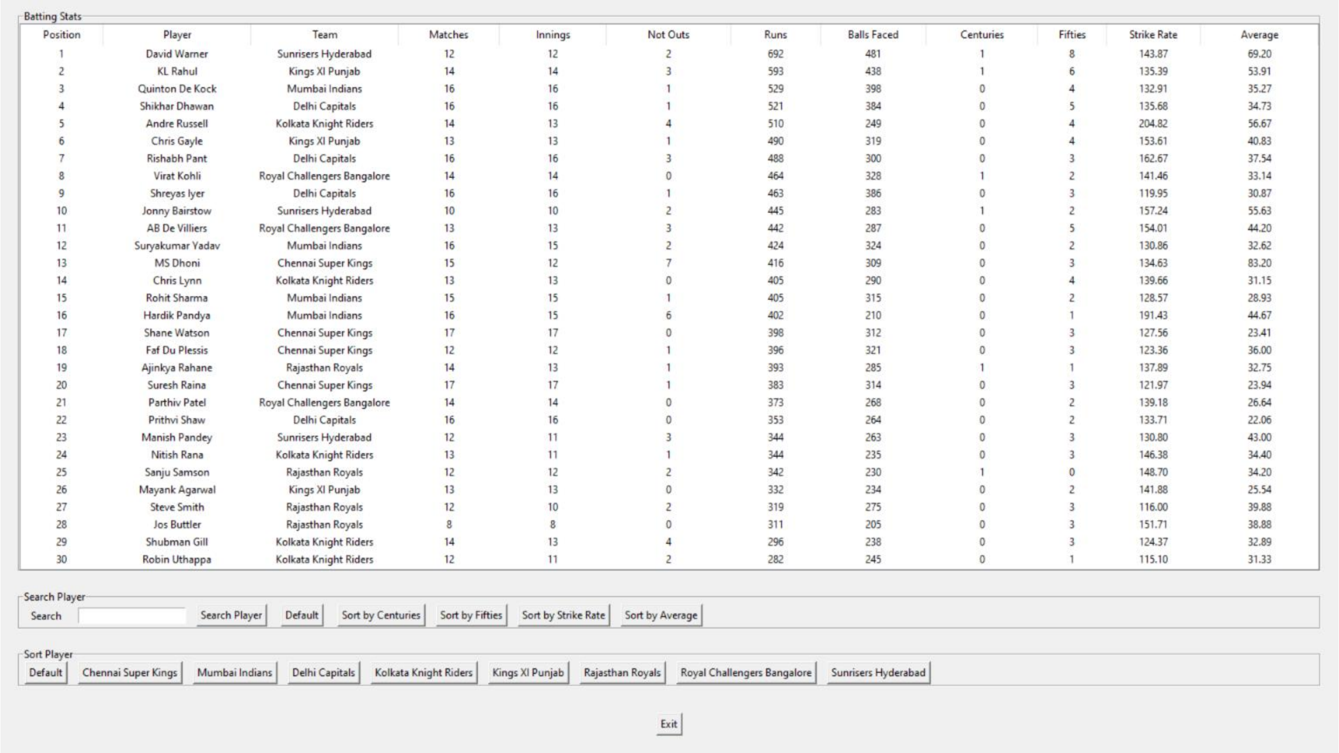


Position	Team	Matches	Wins	Losses	No Result	Net Run Rate	Points
1	Mumbai Indians	14	9	5	0	0.421	18
2	Chennai Super Kings	14	9	5	0	0.131	18
3	Delhi Capitals	14	9	5	0	0.044	18
4	Sunrises Hyderabad	14	6	8	0	0.577	12
5	Kolkata Knight Riders	14	6	8	0	0.028	12
6	Kings XI Punjab	14	6	8	0	-0.251	12
7	Rajasthan Royals	14	5	8	1	-0.449	11
8	Royal Challengers Bangalore	14	5	8	1	-0.607	11

iii) Player Stats 2019 submenu:-

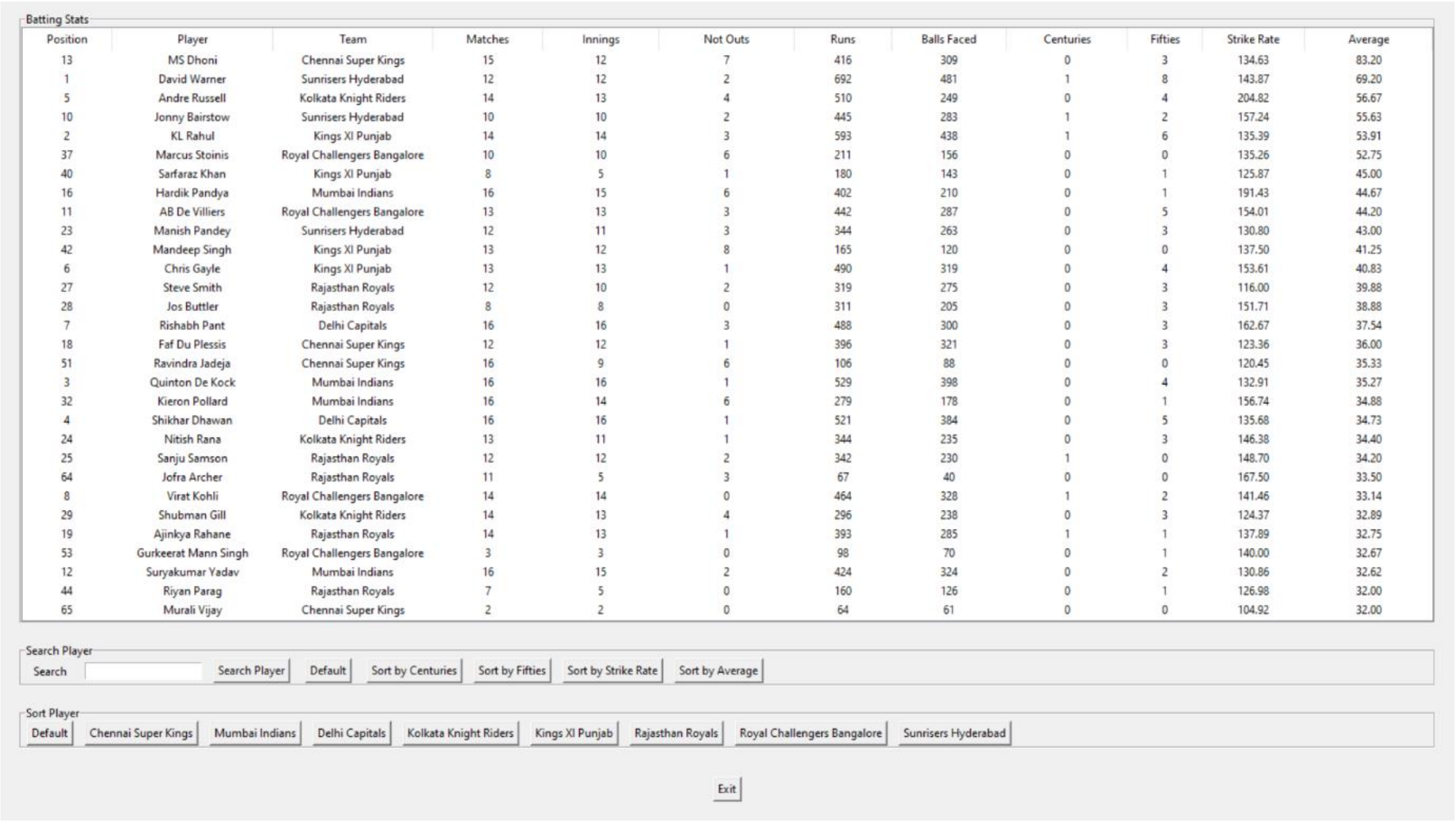


iv) Batting Stats 2019(Player Stats):-



Position	Player	Team	Matches	Innings	Not Outs	Runs	Balls Faced	Centuries	Fifties	Strike Rate	Average
1	David Warner	Sunrisers Hyderabad	12	12	2	692	481	1	8	143.87	69.20
2	KL Rahul	Kings XI Punjab	14	14	3	593	438	1	6	135.39	53.91
3	Quinton De Kock	Mumbai Indians	16	16	1	529	398	0	4	132.91	35.27
4	Shikhar Dhawan	Delhi Capitals	16	16	1	521	384	0	5	135.68	34.73
5	Andre Russell	Kolkata Knight Riders	14	13	4	510	249	0	4	204.82	56.67
6	Chris Gayle	Kings XI Punjab	13	13	1	490	319	0	4	153.61	40.83
7	Rishabh Pant	Delhi Capitals	16	16	3	488	300	0	3	162.67	37.54
8	Virat Kohli	Royal Challengers Bangalore	14	14	0	464	328	1	2	141.46	33.14
9	Shreyas Iyer	Delhi Capitals	16	16	1	463	386	0	3	119.95	30.87
10	Jonny Bairstow	Sunrisers Hyderabad	10	10	2	445	283	1	2	157.24	55.63
11	AB De Villiers	Royal Challengers Bangalore	13	13	3	442	287	0	5	154.01	44.20
12	Suryakumar Yadav	Mumbai Indians	16	15	2	424	324	0	2	130.86	32.62
13	MS Dhoni	Chennai Super Kings	15	12	7	416	309	0	3	134.63	83.20
14	Chris Lynn	Kolkata Knight Riders	13	13	0	405	290	0	4	139.66	31.15
15	Rohit Sharma	Mumbai Indians	15	15	1	405	315	0	2	128.57	28.93
16	Hardik Pandya	Mumbai Indians	16	15	6	402	210	0	1	191.43	44.67
17	Shane Watson	Chennai Super Kings	17	17	0	398	312	0	3	127.56	23.41
18	Faf Du Plessis	Chennai Super Kings	12	12	1	396	321	0	3	123.36	36.00
19	Ajinkya Rahane	Rajasthan Royals	14	13	1	393	285	1	1	137.89	32.75
20	Suresh Raina	Chennai Super Kings	17	17	1	383	314	0	3	121.97	23.94
21	Parthiv Patel	Royal Challengers Bangalore	14	14	0	373	268	0	2	139.18	26.64
22	Prithvi Shaw	Delhi Capitals	16	16	0	353	264	0	2	133.71	22.06
23	Manish Pandey	Sunrisers Hyderabad	12	11	3	344	263	0	3	130.80	43.00
24	Nitish Rana	Kolkata Knight Riders	13	11	1	344	235	0	3	146.38	34.40
25	Sanju Samson	Rajasthan Royals	12	12	2	342	230	1	0	148.70	34.20
26	Mayank Agarwal	Kings XI Punjab	13	13	0	332	234	0	2	141.88	25.54
27	Steve Smith	Rajasthan Royals	12	10	2	319	275	0	3	116.00	39.88
28	Jos Buttler	Rajasthan Royals	8	8	0	311	205	0	3	151.71	38.88
29	Shubman Gill	Kolkata Knight Riders	14	13	4	296	238	0	3	124.37	32.89
30	Robin Uthappa	Kolkata Knight Riders	12	11	2	282	245	0	1	115.10	31.33

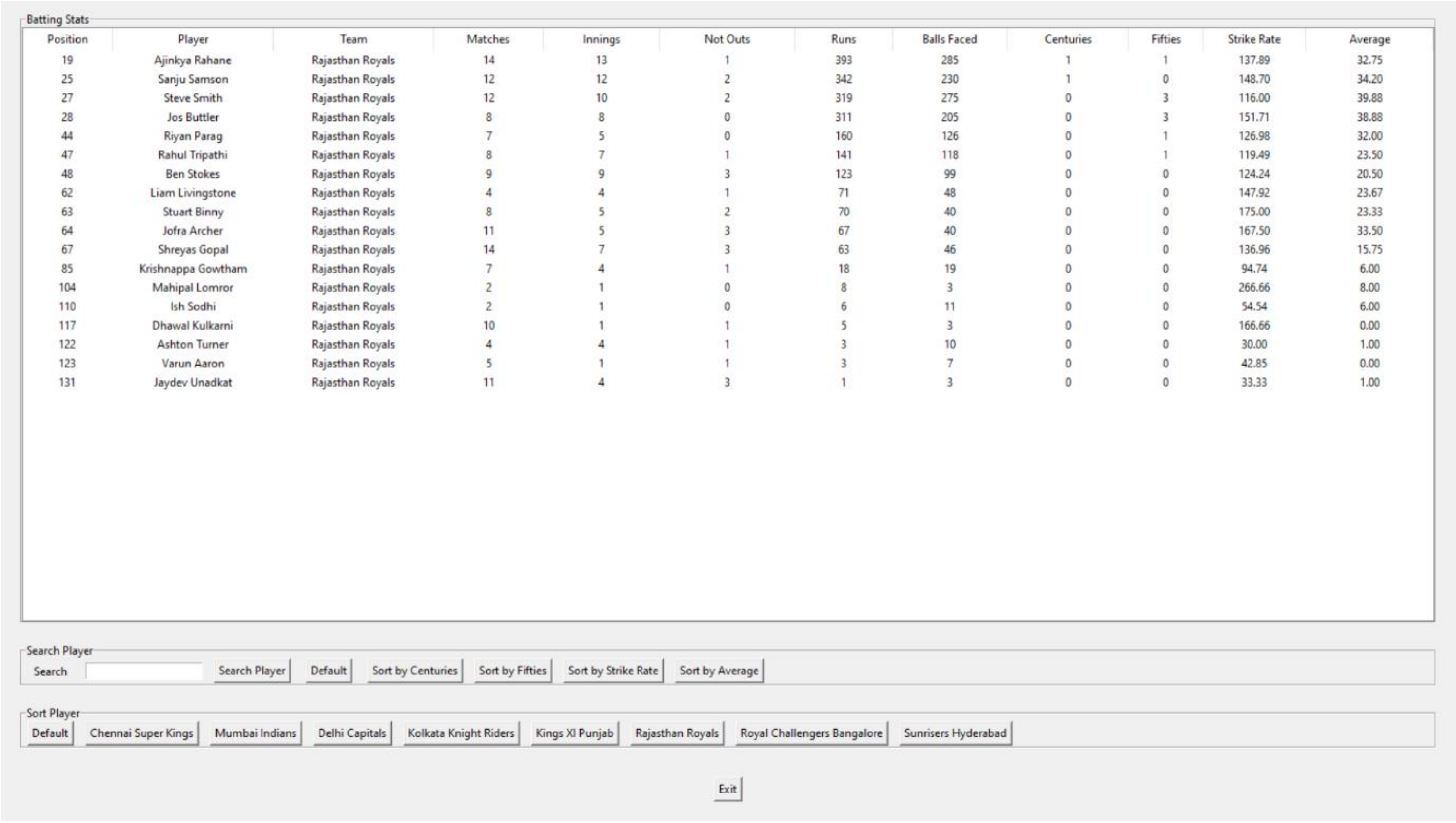
Batting Stats 2019 (sorted by Average):-



Batting Stats

Position	Player	Team	Matches	Innings	Not Outs	Runs	Balls Faced	Centuries	Fifties	Strike Rate	Average
13	MS Dhoni	Chennai Super Kings	15	12	7	416	309	0	3	134.63	83.20
1	David Warner	Sunrisers Hyderabad	12	12	2	692	481	1	8	143.87	69.20
5	Andre Russell	Kolkata Knight Riders	14	13	4	510	249	0	4	204.82	56.67
10	Jonny Bairstow	Sunrisers Hyderabad	10	10	2	445	283	1	2	157.24	55.63
2	KL Rahul	Kings XI Punjab	14	14	3	593	438	1	6	135.39	53.91
37	Marcus Stoinis	Royal Challengers Bangalore	10	10	6	211	156	0	0	135.26	52.75
40	Sarfaraz Khan	Kings XI Punjab	8	5	1	180	143	0	1	125.87	45.00
16	Hardik Pandya	Mumbai Indians	16	15	6	402	210	0	1	191.43	44.67
11	AB De Villiers	Royal Challengers Bangalore	13	13	3	442	287	0	5	154.01	44.20
23	Manish Pandey	Sunrisers Hyderabad	12	11	3	344	263	0	3	130.80	43.00
42	Mandeep Singh	Kings XI Punjab	13	12	8	165	120	0	0	137.50	41.25
6	Chris Gayle	Kings XI Punjab	13	13	1	490	319	0	4	153.61	40.83
27	Steve Smith	Rajasthan Royals	12	10	2	319	275	0	3	116.00	39.88
28	Jos Buttler	Rajasthan Royals	8	8	0	311	205	0	3	151.71	38.88
7	Rishabh Pant	Delhi Capitals	16	16	3	488	300	0	3	162.67	37.54
18	Faf Du Plessis	Chennai Super Kings	12	12	1	396	321	0	3	123.36	36.00
51	Ravindra Jadeja	Chennai Super Kings	16	9	6	106	88	0	0	120.45	35.33
3	Quinton De Kock	Mumbai Indians	16	16	1	529	398	0	4	132.91	35.27
32	Kieron Pollard	Mumbai Indians	16	14	6	279	178	0	1	156.74	34.88
4	Shikhar Dhawan	Delhi Capitals	16	16	1	521	384	0	5	135.68	34.73
24	Nitish Rana	Kolkata Knight Riders	13	11	1	344	235	0	3	146.38	34.40
25	Sanju Samson	Rajasthan Royals	12	12	2	342	230	1	0	148.70	34.20
64	Jofra Archer	Rajasthan Royals	11	5	3	67	40	0	0	167.50	33.50
8	Virat Kohli	Royal Challengers Bangalore	14	14	0	464	328	1	2	141.46	33.14
29	Shubman Gill	Kolkata Knight Riders	14	13	4	296	238	0	3	124.37	32.89
19	Ajinkya Rahane	Rajasthan Royals	14	13	1	393	285	1	1	137.89	32.75
53	Gurkeerat Mann Singh	Royal Challengers Bangalore	3	3	0	98	70	0	1	140.00	32.67
12	Suryakumar Yadav	Mumbai Indians	16	15	2	424	324	0	2	130.86	32.62
44	Riyan Parag	Rajasthan Royals	7	5	0	160	126	0	1	126.98	32.00
65	Murali Vijay	Chennai Super Kings	2	2	0	64	61	0	0	104.92	32.00

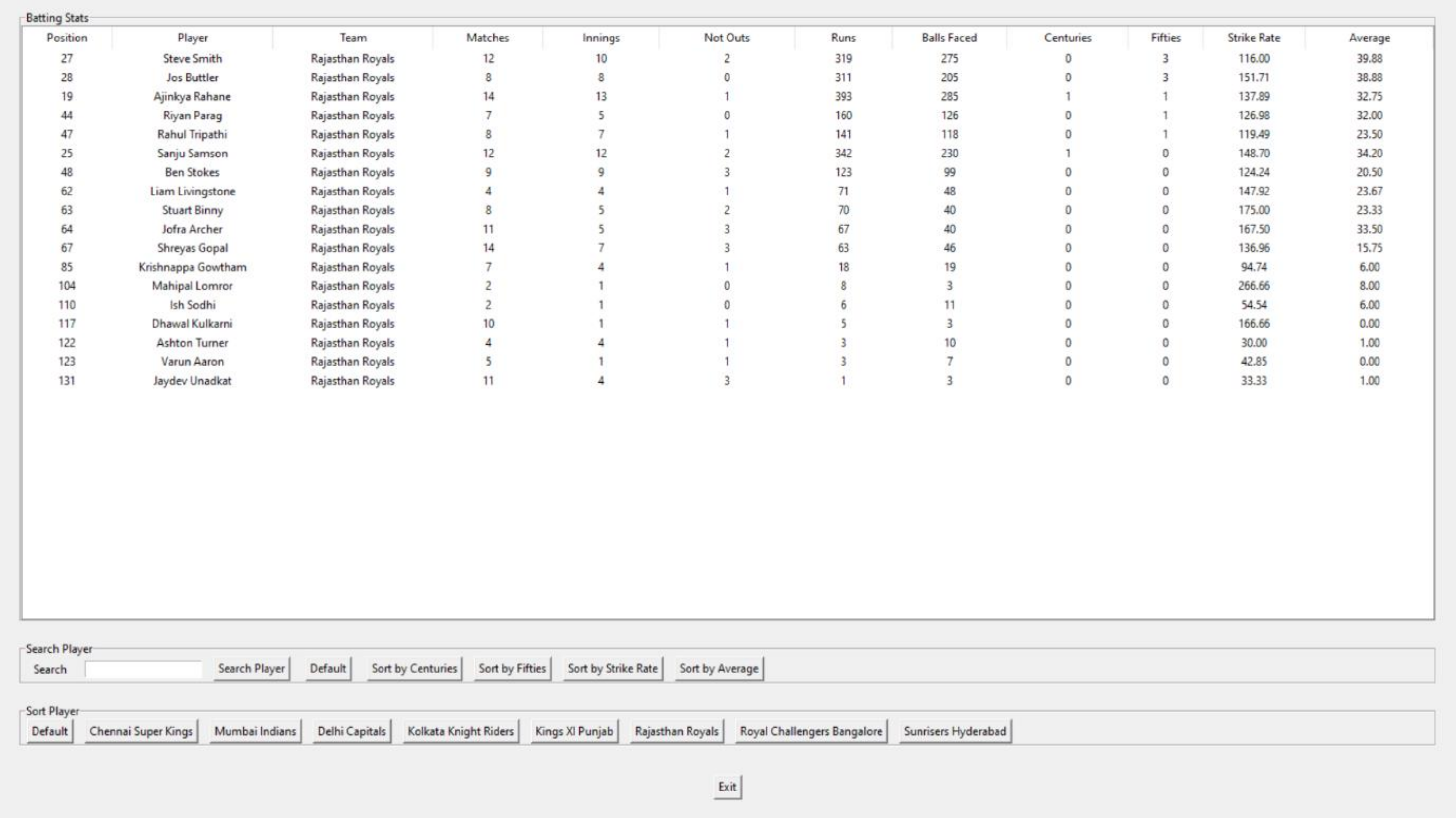
Batting Stats 2019(Sorted by Team):-



Batting Stats

Position	Player	Team	Matches	Innings	Not Outs	Runs	Balls Faced	Centuries	Fifties	Strike Rate	Average
19	Ajinkya Rahane	Rajasthan Royals	14	13	1	393	285	1	1	137.89	32.75
25	Sanju Samson	Rajasthan Royals	12	12	2	342	230	1	0	148.70	34.20
27	Steve Smith	Rajasthan Royals	12	10	2	319	275	0	3	116.00	39.88
28	Jos Buttler	Rajasthan Royals	8	8	0	311	205	0	3	151.71	38.88
44	Riyan Parag	Rajasthan Royals	7	5	0	160	126	0	1	126.98	32.00
47	Rahul Tripathi	Rajasthan Royals	8	7	1	141	118	0	1	119.49	23.50
48	Ben Stokes	Rajasthan Royals	9	9	3	123	99	0	0	124.24	20.50
62	Liam Livingstone	Rajasthan Royals	4	4	1	71	48	0	0	147.92	23.67
63	Stuart Binny	Rajasthan Royals	8	5	2	70	40	0	0	175.00	23.33
64	Jofra Archer	Rajasthan Royals	11	5	3	67	40	0	0	167.50	33.50
67	Shreyas Gopal	Rajasthan Royals	14	7	3	63	46	0	0	136.96	15.75
85	Krishnappa Gowtham	Rajasthan Royals	7	4	1	18	19	0	0	94.74	6.00
104	Mahipal Lomror	Rajasthan Royals	2	1	0	8	3	0	0	266.66	8.00
110	Ish Sodhi	Rajasthan Royals	2	1	0	6	11	0	0	54.54	6.00
117	Dhawal Kulkarni	Rajasthan Royals	10	1	1	5	3	0	0	166.66	0.00
122	Ashton Turner	Rajasthan Royals	4	4	1	3	10	0	0	30.00	1.00
123	Varun Aaron	Rajasthan Royals	5	1	1	3	7	0	0	42.85	0.00
131	Jaydev Unadkat	Rajasthan Royals	11	4	3	1	3	0	0	33.33	1.00

Batting Stats(Sorted by Team and a column):-



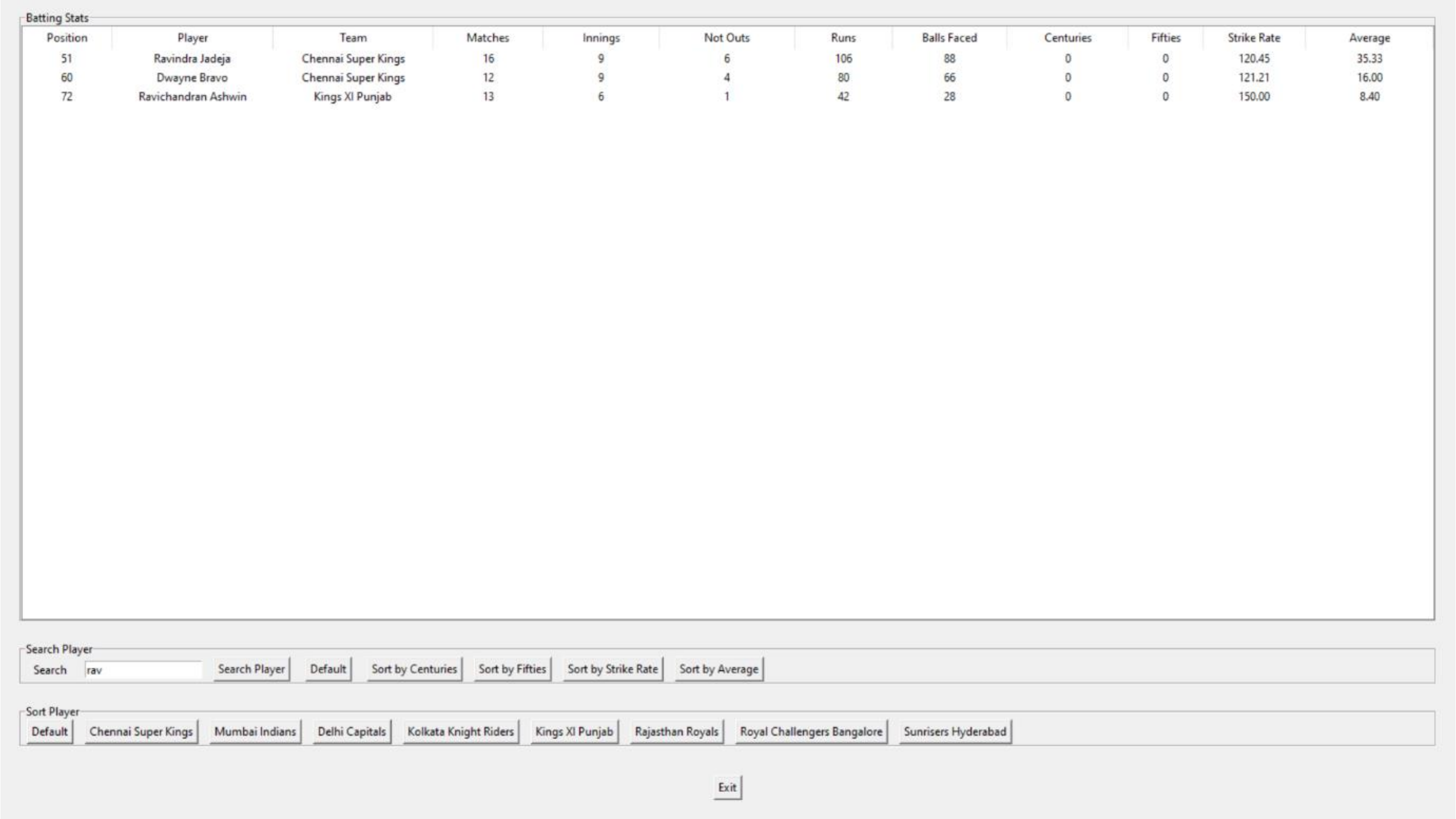
Batting Stats

Position	Player	Team	Matches	Innings	Not Outs	Runs	Balls Faced	Centuries	Fifties	Strike Rate	Average
27	Steve Smith	Rajasthan Royals	12	10	2	319	275	0	3	116.00	39.88
28	Jos Buttler	Rajasthan Royals	8	8	0	311	205	0	3	151.71	38.88
19	Ajinkya Rahane	Rajasthan Royals	14	13	1	393	285	1	1	137.89	32.75
44	Riyan Parag	Rajasthan Royals	7	5	0	160	126	0	1	126.98	32.00
47	Rahul Tripathi	Rajasthan Royals	8	7	1	141	118	0	1	119.49	23.50
25	Sanju Samson	Rajasthan Royals	12	12	2	342	230	1	0	148.70	34.20
48	Ben Stokes	Rajasthan Royals	9	9	3	123	99	0	0	124.24	20.50
62	Liam Livingstone	Rajasthan Royals	4	4	1	71	48	0	0	147.92	23.67
63	Stuart Binny	Rajasthan Royals	8	5	2	70	40	0	0	175.00	23.33
64	Jofra Archer	Rajasthan Royals	11	5	3	67	40	0	0	167.50	33.50
67	Shreyas Gopal	Rajasthan Royals	14	7	3	63	46	0	0	136.96	15.75
85	Krishnappa Gowtham	Rajasthan Royals	7	4	1	18	19	0	0	94.74	6.00
104	Mahipal Lomror	Rajasthan Royals	2	1	0	8	3	0	0	266.66	8.00
110	Ish Sodhi	Rajasthan Royals	2	1	0	6	11	0	0	54.54	6.00
117	Dhawal Kulkarni	Rajasthan Royals	10	1	1	5	3	0	0	166.66	0.00
122	Ashton Turner	Rajasthan Royals	4	4	1	3	10	0	0	30.00	1.00
123	Varun Aaron	Rajasthan Royals	5	1	1	3	7	0	0	42.85	0.00
131	Jaydev Unadkat	Rajasthan Royals	11	4	3	1	3	0	0	33.33	1.00

Search Player

Search

Batting Stats(Search by keyword):-



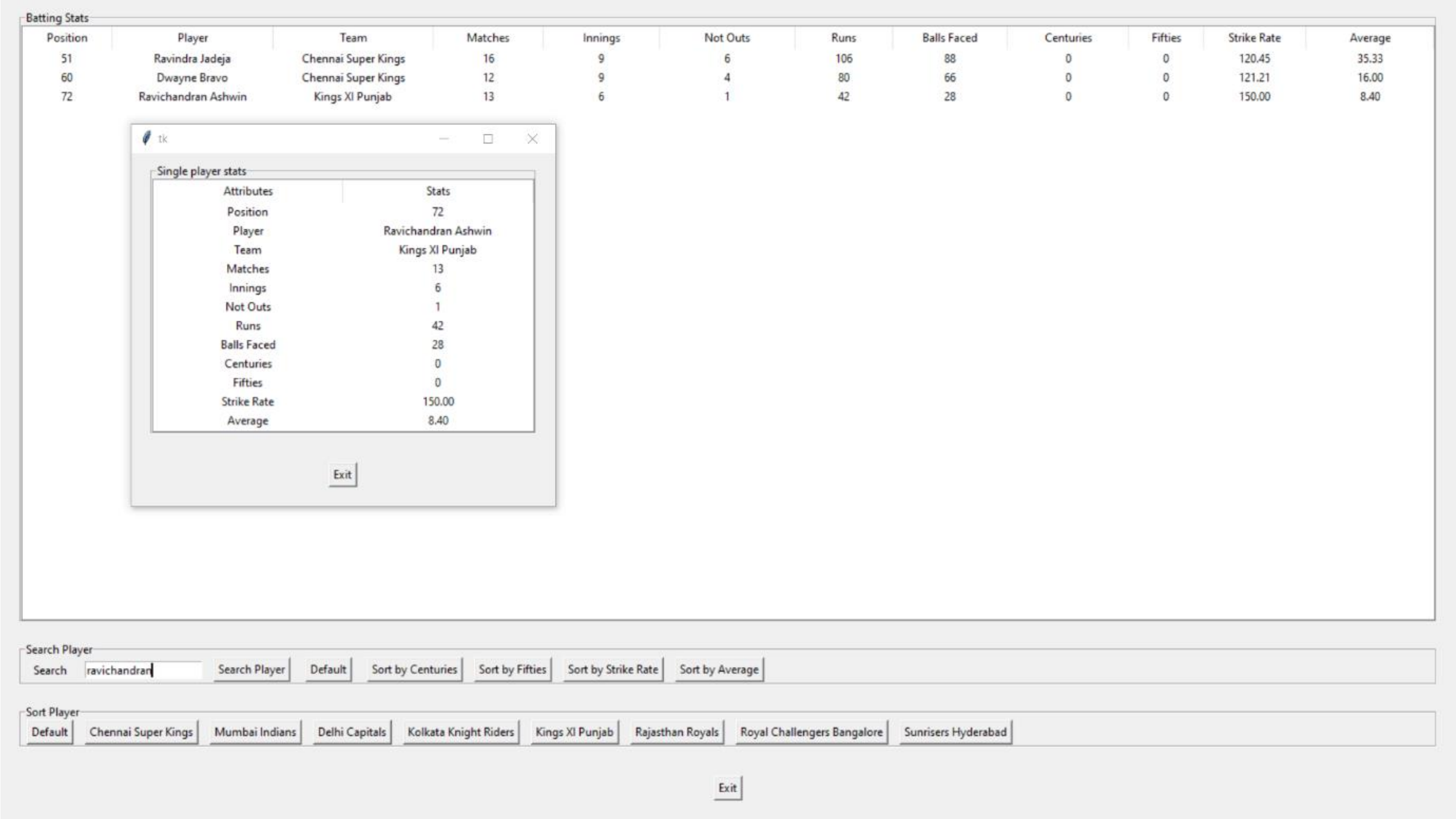
Batting Stats

Position	Player	Team	Matches	Innings	Not Outs	Runs	Balls Faced	Centuries	Fifties	Strike Rate	Average
51	Ravindra Jadeja	Chennai Super Kings	16	9	6	106	88	0	0	120.45	35.33
60	Dwayne Bravo	Chennai Super Kings	12	9	4	80	66	0	0	121.21	16.00
72	Ravichandran Ashwin	Kings XI Punjab	13	6	1	42	28	0	0	150.00	8.40

Search Player

Search rav

Batting Stats 2019(Search by single player):-



Batting Stats

Position	Player	Team	Matches	Innings	Not Outs	Runs	Balls Faced	Centuries	Fifties	Strike Rate	Average
51	Ravindra Jadeja	Chennai Super Kings	16	9	6	106	88	0	0	120.45	35.33
60	Dwayne Bravo	Chennai Super Kings	12	9	4	80	66	0	0	121.21	16.00
72	Ravichandran Ashwin	Kings XI Punjab	13	6	1	42	28	0	0	150.00	8.40

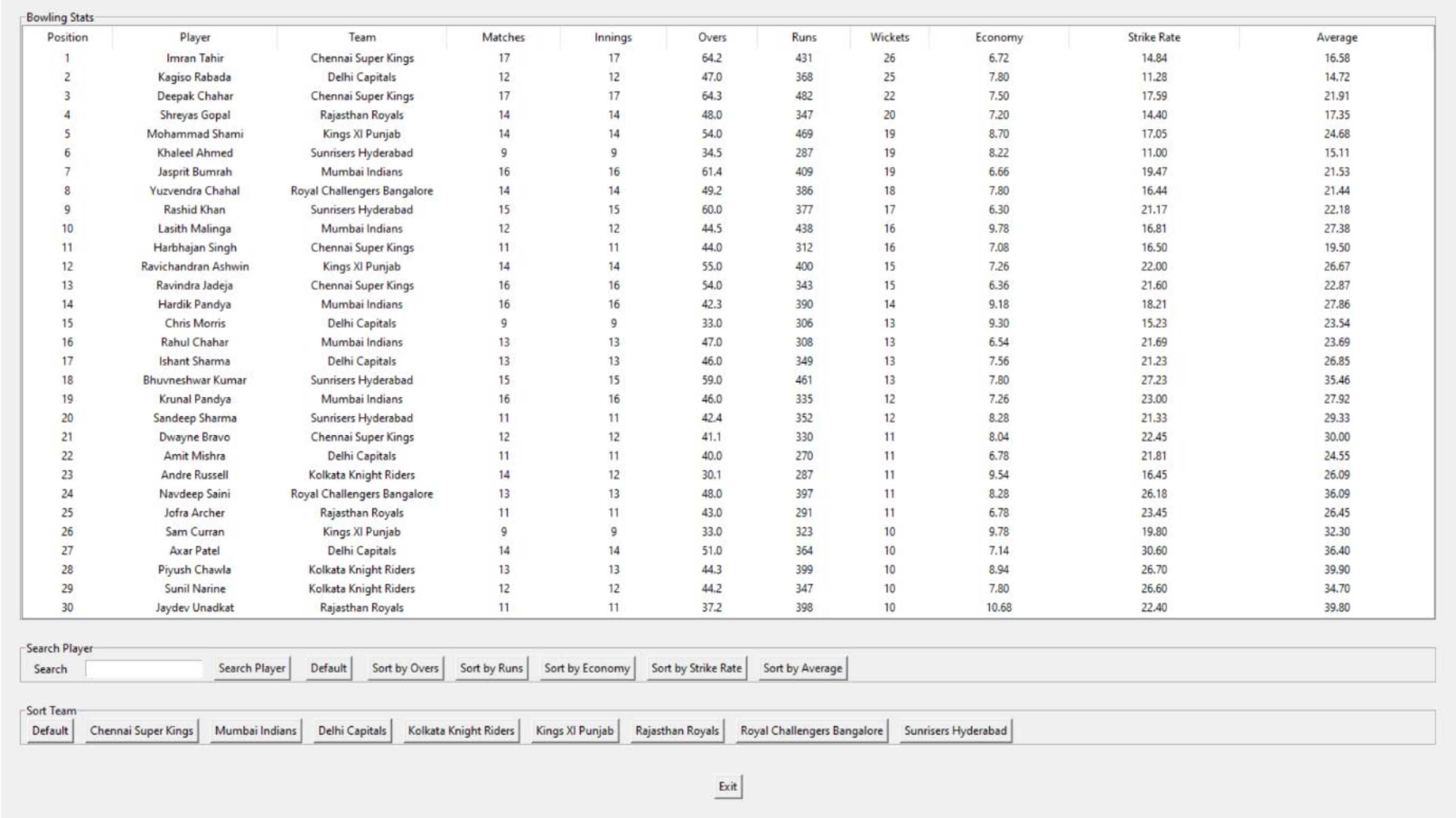
Single player stats

Attributes	Stats
Position	72
Player	Ravichandran Ashwin
Team	Kings XI Punjab
Matches	13
Innings	6
Not Outs	1
Runs	42
Balls Faced	28
Centuries	0
Fifties	0
Strike Rate	150.00
Average	8.40

Search Player

Search: ravichandran

v) Bowling Stats 2019(Player Stats):-



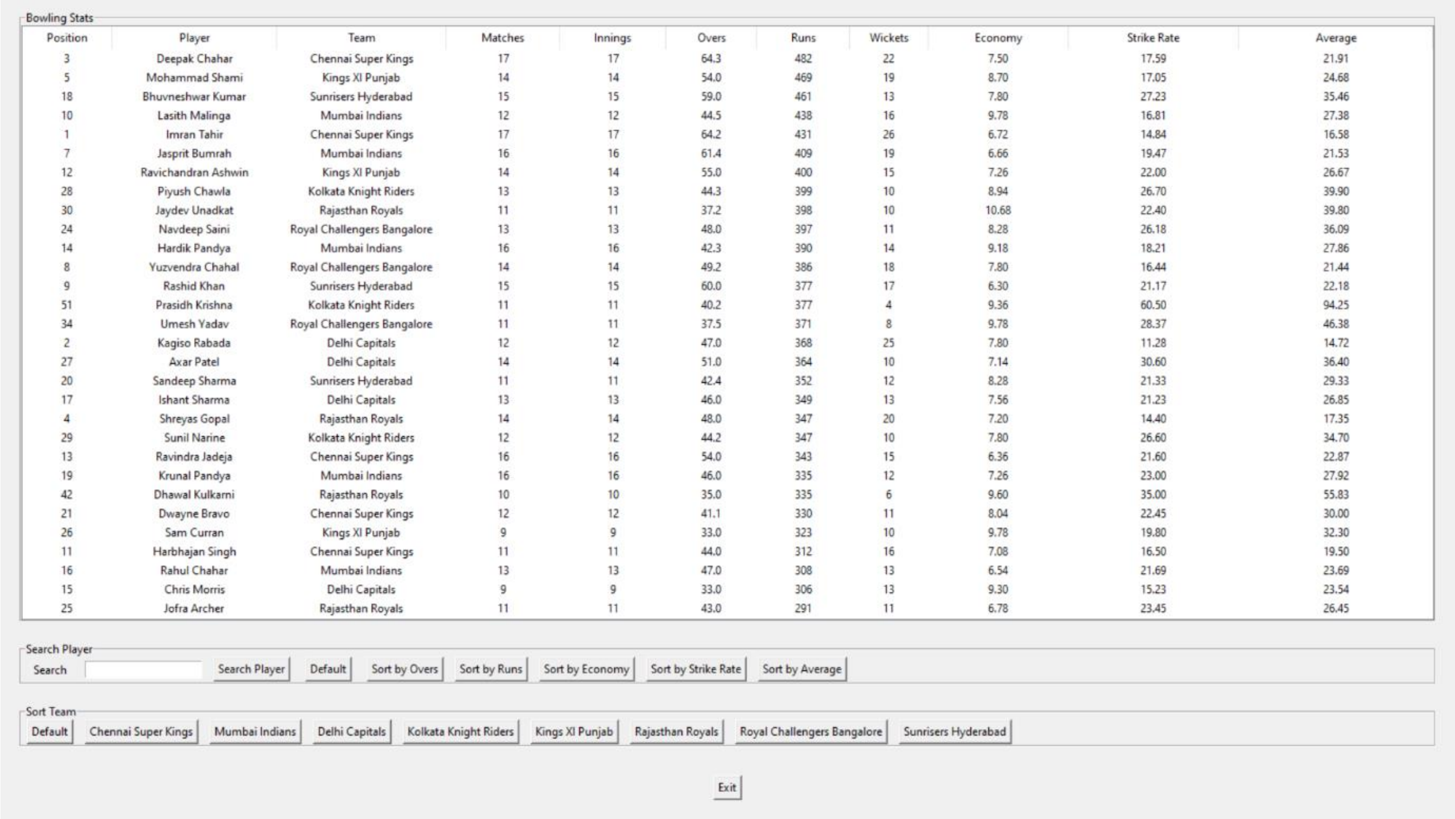
Bowling Stats

Position	Player	Team	Matches	Innings	Overs	Runs	Wickets	Economy	Strike Rate	Average
1	Imran Tahir	Chennai Super Kings	17	17	64.2	431	26	6.72	14.84	16.58
2	Kagiso Rabada	Delhi Capitals	12	12	47.0	368	25	7.80	11.28	14.72
3	Deepak Chahar	Chennai Super Kings	17	17	64.3	482	22	7.50	17.59	21.91
4	Shreyas Gopal	Rajasthan Royals	14	14	48.0	347	20	7.20	14.40	17.35
5	Mohammad Shami	Kings XI Punjab	14	14	54.0	469	19	8.70	17.05	24.68
6	Khaleel Ahmed	Sunrisers Hyderabad	9	9	34.5	287	19	8.22	11.00	15.11
7	Jasprit Bumrah	Mumbai Indians	16	16	61.4	409	19	6.66	19.47	21.53
8	Yuzvendra Chahal	Royal Challengers Bangalore	14	14	49.2	386	18	7.80	16.44	21.44
9	Rashid Khan	Sunrisers Hyderabad	15	15	60.0	377	17	6.30	21.17	22.18
10	Lasith Malinga	Mumbai Indians	12	12	44.5	438	16	9.78	16.81	27.38
11	Harbhajan Singh	Chennai Super Kings	11	11	44.0	312	16	7.08	16.50	19.50
12	Ravichandran Ashwin	Kings XI Punjab	14	14	55.0	400	15	7.26	22.00	26.67
13	Ravindra Jadeja	Chennai Super Kings	16	16	54.0	343	15	6.36	21.60	22.87
14	Hardik Pandya	Mumbai Indians	16	16	42.3	390	14	9.18	18.21	27.86
15	Chris Morris	Delhi Capitals	9	9	33.0	306	13	9.30	15.23	23.54
16	Rahul Chahar	Mumbai Indians	13	13	47.0	308	13	6.54	21.69	23.69
17	Ishant Sharma	Delhi Capitals	13	13	46.0	349	13	7.56	21.23	26.85
18	Bhuvneshwar Kumar	Sunrisers Hyderabad	15	15	59.0	461	13	7.80	27.23	35.46
19	Krunal Pandya	Mumbai Indians	16	16	46.0	335	12	7.26	23.00	27.92
20	Sandeep Sharma	Sunrisers Hyderabad	11	11	42.4	352	12	8.28	21.33	29.33
21	Dwayne Bravo	Chennai Super Kings	12	12	41.1	330	11	8.04	22.45	30.00
22	Amit Mishra	Delhi Capitals	11	11	40.0	270	11	6.78	21.81	24.55
23	Andre Russell	Kolkata Knight Riders	14	12	30.1	287	11	9.54	16.45	26.09
24	Navdeep Saini	Royal Challengers Bangalore	13	13	48.0	397	11	8.28	26.18	36.09
25	Jofra Archer	Rajasthan Royals	11	11	43.0	291	11	6.78	23.45	26.45
26	Sam Curran	Kings XI Punjab	9	9	33.0	323	10	9.78	19.80	32.30
27	Axar Patel	Delhi Capitals	14	14	51.0	364	10	7.14	30.60	36.40
28	Piyush Chawla	Kolkata Knight Riders	13	13	44.3	399	10	8.94	26.70	39.90
29	Sunil Narine	Kolkata Knight Riders	12	12	44.2	347	10	7.80	26.60	34.70
30	Jaydev Unadkat	Rajasthan Royals	11	11	37.2	398	10	10.68	22.40	39.80

Search Player

Search:

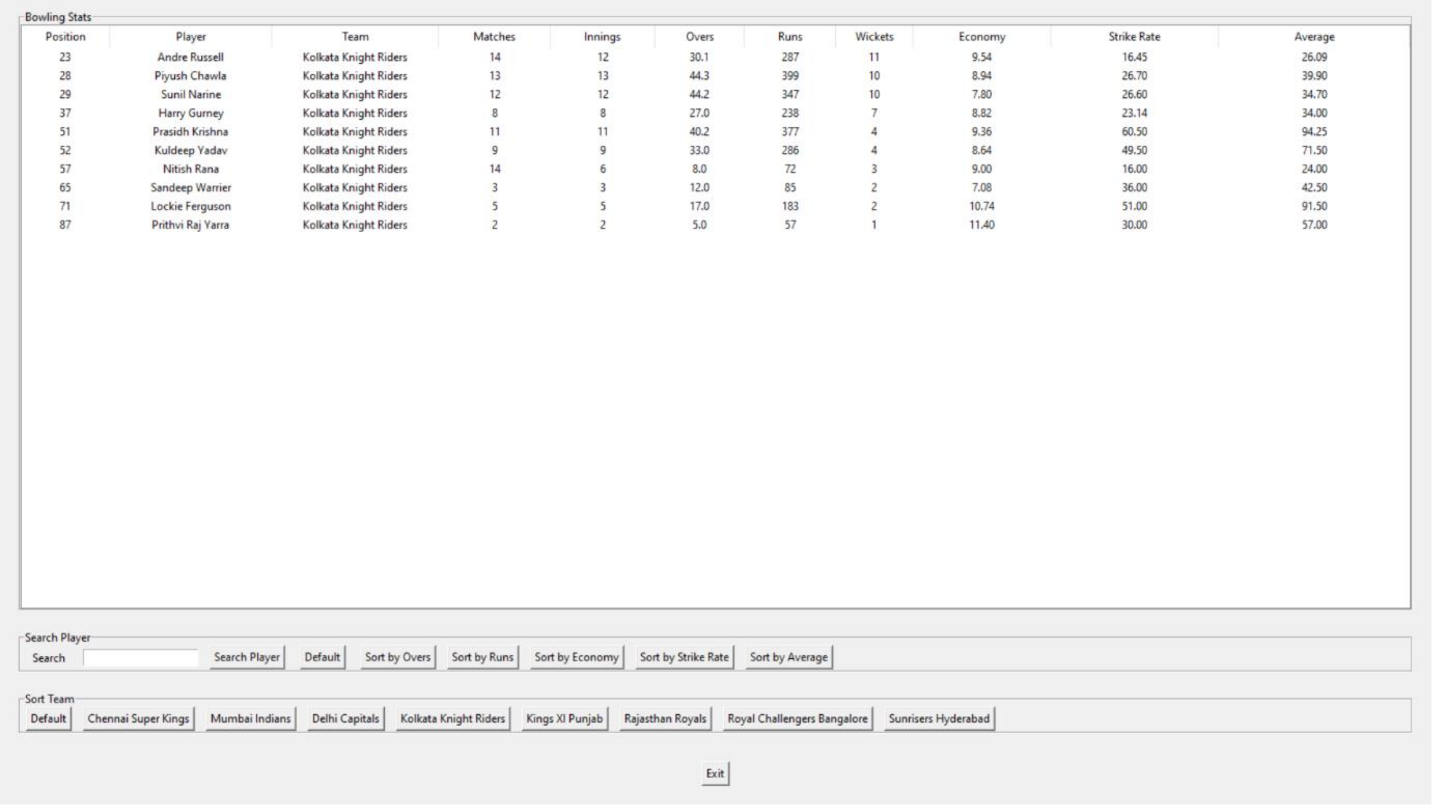
Bowling Stats 2019(Sorted By Runs):-



Bowling Stats

Position	Player	Team	Matches	Innings	Overs	Runs	Wickets	Economy	Strike Rate	Average
3	Deepak Chahar	Chennai Super Kings	17	17	64.3	482	22	7.50	17.59	21.91
5	Mohammad Shami	Kings XI Punjab	14	14	54.0	469	19	8.70	17.05	24.68
18	Bhuvneshwar Kumar	Sunrisers Hyderabad	15	15	59.0	461	13	7.80	27.23	35.46
10	Lasith Malinga	Mumbai Indians	12	12	44.5	438	16	9.78	16.81	27.38
1	Imran Tahir	Chennai Super Kings	17	17	64.2	431	26	6.72	14.84	16.58
7	Jasprit Bumrah	Mumbai Indians	16	16	61.4	409	19	6.66	19.47	21.53
12	Ravichandran Ashwin	Kings XI Punjab	14	14	55.0	400	15	7.26	22.00	26.67
28	Piyush Chawla	Kolkata Knight Riders	13	13	44.3	399	10	8.94	26.70	39.90
30	Jaydev Unadkat	Rajasthan Royals	11	11	37.2	398	10	10.68	22.40	39.80
24	Navdeep Saini	Royal Challengers Bangalore	13	13	48.0	397	11	8.28	26.18	36.09
14	Hardik Pandya	Mumbai Indians	16	16	42.3	390	14	9.18	18.21	27.86
8	Yuzvendra Chahal	Royal Challengers Bangalore	14	14	49.2	386	18	7.80	16.44	21.44
9	Rashid Khan	Sunrisers Hyderabad	15	15	60.0	377	17	6.30	21.17	22.18
51	Prasidh Krishna	Kolkata Knight Riders	11	11	40.2	377	4	9.36	60.50	94.25
34	Umesh Yadav	Royal Challengers Bangalore	11	11	37.5	371	8	9.78	28.37	46.38
2	Kagiso Rabada	Delhi Capitals	12	12	47.0	368	25	7.80	11.28	14.72
27	Axar Patel	Delhi Capitals	14	14	51.0	364	10	7.14	30.60	36.40
20	Sandeep Sharma	Sunrisers Hyderabad	11	11	42.4	352	12	8.28	21.33	29.33
17	Ishant Sharma	Delhi Capitals	13	13	46.0	349	13	7.56	21.23	26.85
4	Shreyas Gopal	Rajasthan Royals	14	14	48.0	347	20	7.20	14.40	17.35
29	Sunil Narine	Kolkata Knight Riders	12	12	44.2	347	10	7.80	26.60	34.70
13	Ravindra Jadeja	Chennai Super Kings	16	16	54.0	343	15	6.36	21.60	22.87
19	Krunal Pandya	Mumbai Indians	16	16	46.0	335	12	7.26	23.00	27.92
42	Dhawal Kulkarni	Rajasthan Royals	10	10	35.0	335	6	9.60	35.00	55.83
21	Dwayne Bravo	Chennai Super Kings	12	12	41.1	330	11	8.04	22.45	30.00
26	Sam Curran	Kings XI Punjab	9	9	33.0	323	10	9.78	19.80	32.30
11	Harbhajan Singh	Chennai Super Kings	11	11	44.0	312	16	7.08	16.50	19.50
16	Rahul Chahar	Mumbai Indians	13	13	47.0	308	13	6.54	21.69	23.69
15	Chris Morris	Delhi Capitals	9	9	33.0	306	13	9.30	15.23	23.54
25	Jofra Archer	Rajasthan Royals	11	11	43.0	291	11	6.78	23.45	26.45

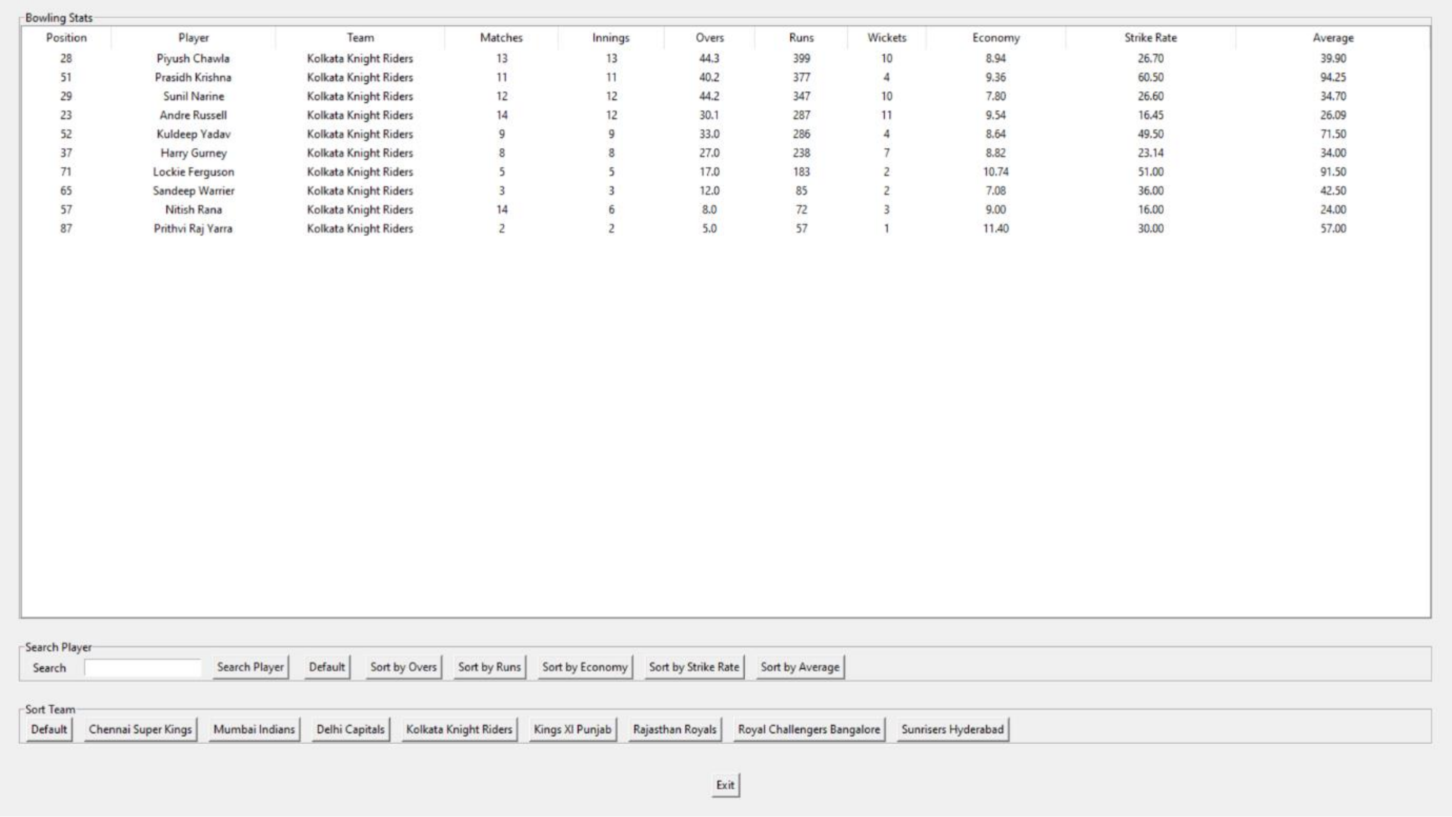
Bowling Stats 2019(Sorted by Team):-



Bowling Stats

Position	Player	Team	Matches	Innings	Overs	Runs	Wickets	Economy	Strike Rate	Average
23	Andre Russell	Kolkata Knight Riders	14	12	30.1	287	11	9.54	16.45	26.09
28	Piyush Chawla	Kolkata Knight Riders	13	13	44.3	399	10	8.94	26.70	39.90
29	Sunil Narine	Kolkata Knight Riders	12	12	44.2	347	10	7.80	26.60	34.70
37	Harry Gurney	Kolkata Knight Riders	8	8	27.0	238	7	8.82	23.14	34.00
51	Prasidh Krishna	Kolkata Knight Riders	11	11	40.2	377	4	9.36	60.50	94.25
52	Kuldeep Yadav	Kolkata Knight Riders	9	9	33.0	286	4	8.64	49.50	71.50
57	Nitish Rana	Kolkata Knight Riders	14	6	8.0	72	3	9.00	16.00	24.00
65	Sandeep Warrier	Kolkata Knight Riders	3	3	12.0	85	2	7.08	36.00	42.50
71	Lockie Ferguson	Kolkata Knight Riders	5	5	17.0	183	2	10.74	51.00	91.50
87	Prithvi Raj Yarra	Kolkata Knight Riders	2	2	5.0	57	1	11.40	30.00	57.00

Bowling Stats 2019(Sorted by Team and a column):-



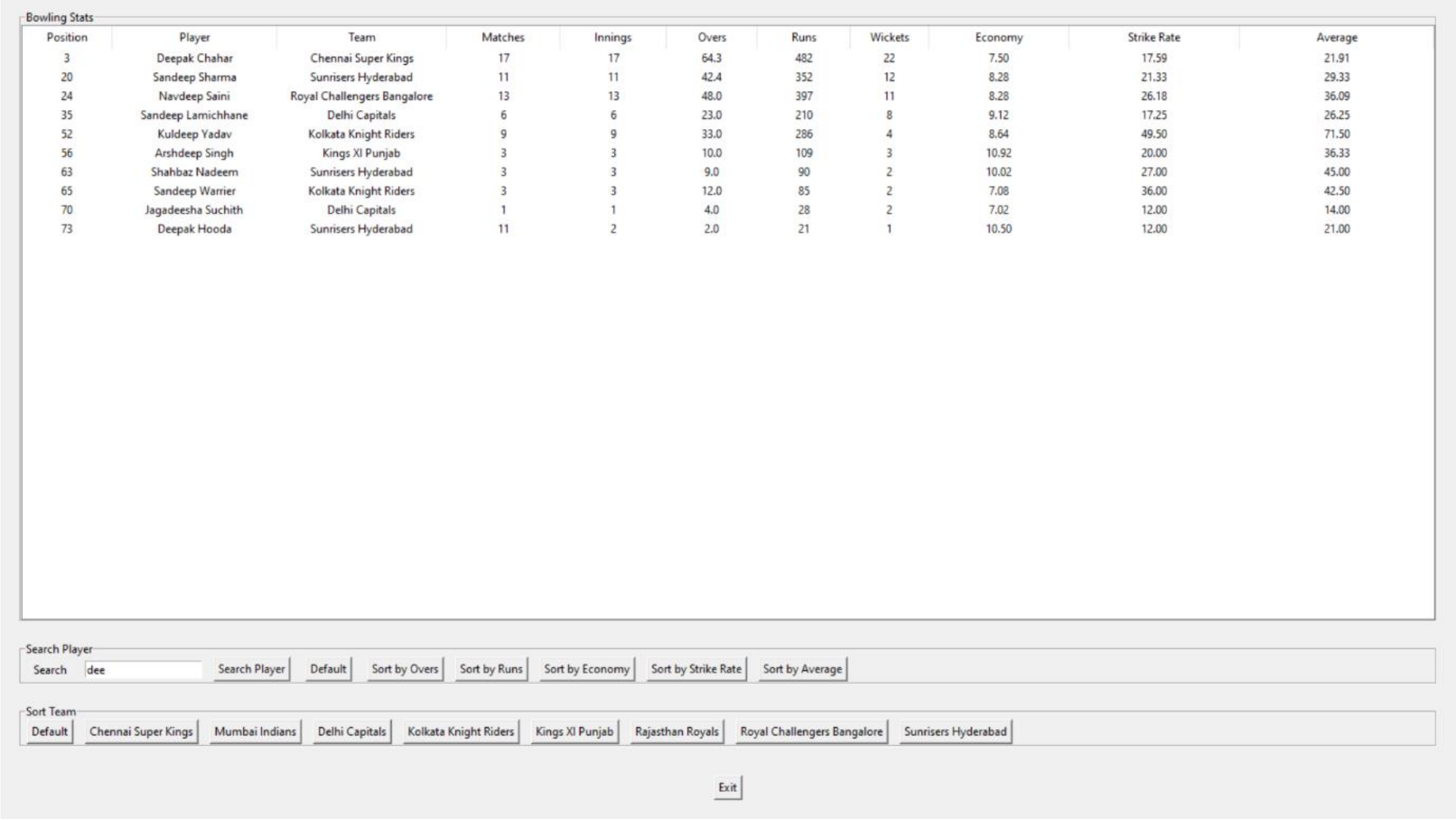
Bowling Stats

Position	Player	Team	Matches	Innings	Overs	Runs	Wickets	Economy	Strike Rate	Average
28	Piyush Chawla	Kolkata Knight Riders	13	13	44.3	399	10	8.94	26.70	39.90
51	Prasidh Krishna	Kolkata Knight Riders	11	11	40.2	377	4	9.36	60.50	94.25
29	Sunil Narine	Kolkata Knight Riders	12	12	44.2	347	10	7.80	26.60	34.70
23	Andre Russell	Kolkata Knight Riders	14	12	30.1	287	11	9.54	16.45	26.09
52	Kuldeep Yadav	Kolkata Knight Riders	9	9	33.0	286	4	8.64	49.50	71.50
37	Harry Gurney	Kolkata Knight Riders	8	8	27.0	238	7	8.82	23.14	34.00
71	Lockie Ferguson	Kolkata Knight Riders	5	5	17.0	183	2	10.74	51.00	91.50
65	Sandeep Warrier	Kolkata Knight Riders	3	3	12.0	85	2	7.08	36.00	42.50
57	Nitish Rana	Kolkata Knight Riders	14	6	8.0	72	3	9.00	16.00	24.00
87	Prithvi Raj Yarra	Kolkata Knight Riders	2	2	5.0	57	1	11.40	30.00	57.00

Search Player

Search

Bowling Stats 2019(Search By keyword):-



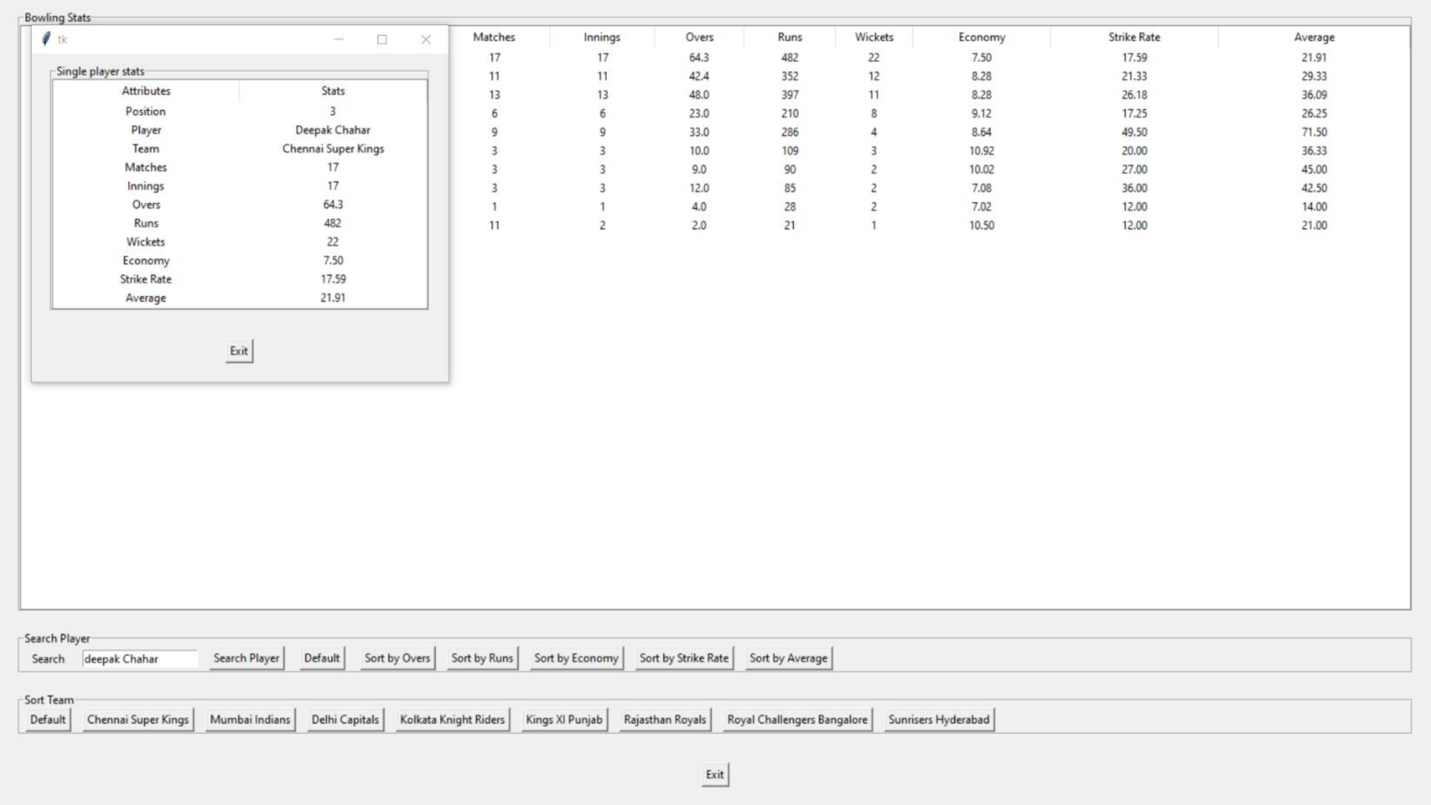
Bowling Stats

Position	Player	Team	Matches	Innings	Overs	Runs	Wickets	Economy	Strike Rate	Average
3	Deepak Chahar	Chennai Super Kings	17	17	64.3	482	22	7.50	17.59	21.91
20	Sandeep Sharma	Sunrisers Hyderabad	11	11	42.4	352	12	8.28	21.33	29.33
24	Navdeep Saini	Royal Challengers Bangalore	13	13	48.0	397	11	8.28	26.18	36.09
35	Sandeep Lamichhane	Delhi Capitals	6	6	23.0	210	8	9.12	17.25	26.25
52	Kuldeep Yadav	Kolkata Knight Riders	9	9	33.0	286	4	8.64	49.50	71.50
56	Arshdeep Singh	Kings XI Punjab	3	3	10.0	109	3	10.92	20.00	36.33
63	Shahbaz Nadeem	Sunrisers Hyderabad	3	3	9.0	90	2	10.02	27.00	45.00
65	Sandeep Warrier	Kolkata Knight Riders	3	3	12.0	85	2	7.08	36.00	42.50
70	Jagadeesha Suchith	Delhi Capitals	1	1	4.0	28	2	7.02	12.00	14.00
73	Deepak Hooda	Sunrisers Hyderabad	11	2	2.0	21	1	10.50	12.00	21.00

Search Player

Search dee

Bowling Stats 2019(Search by Single Player):-

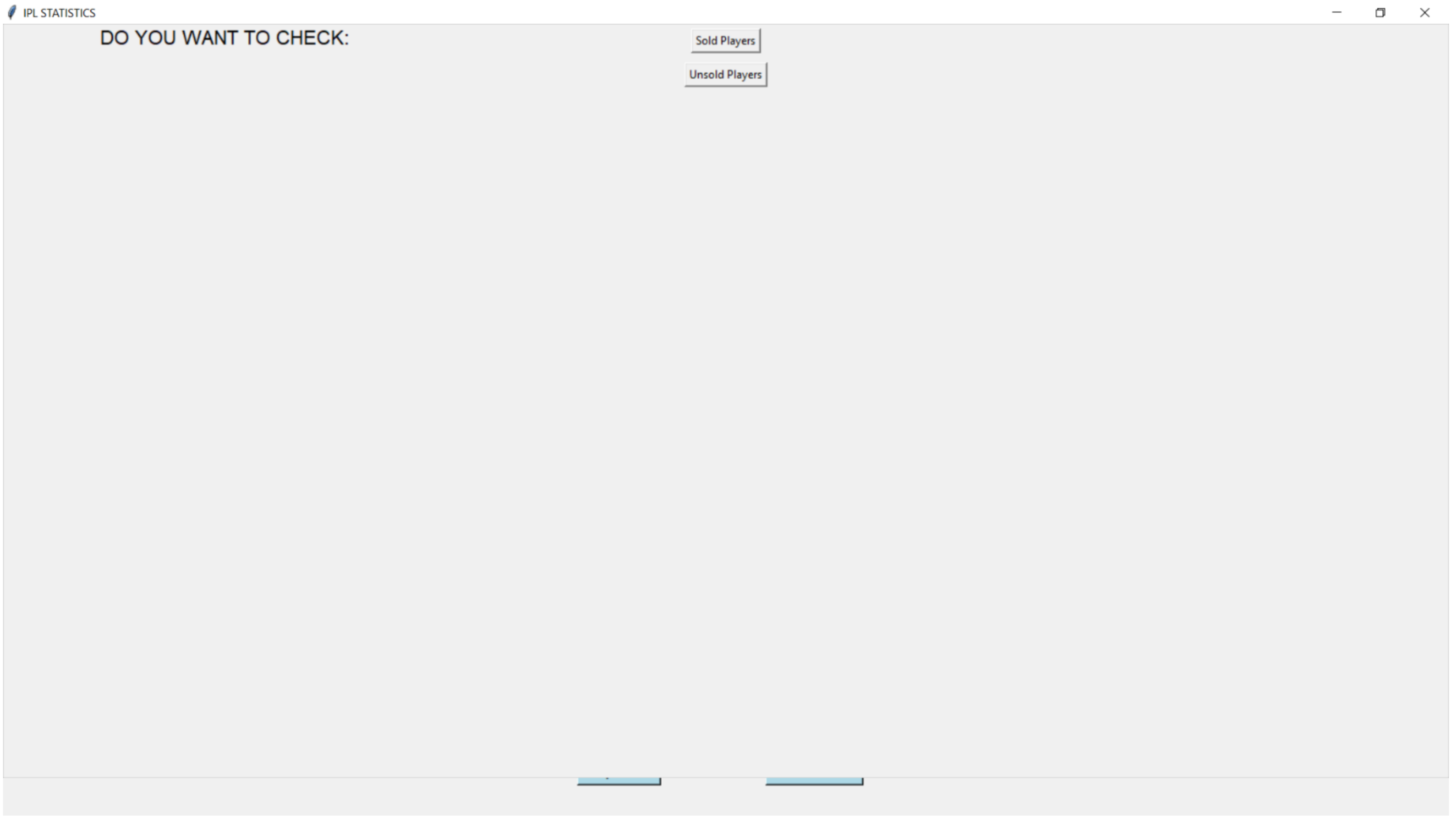


Search: deepak Chahar

Attributes	Stats
Position	3
Player	Deepak Chahar
Team	Chennai Super Kings
Matches	17
Innings	17
Overs	64.3
Runs	482
Wickets	22
Economy	7.50
Strike Rate	17.59
Average	21.91

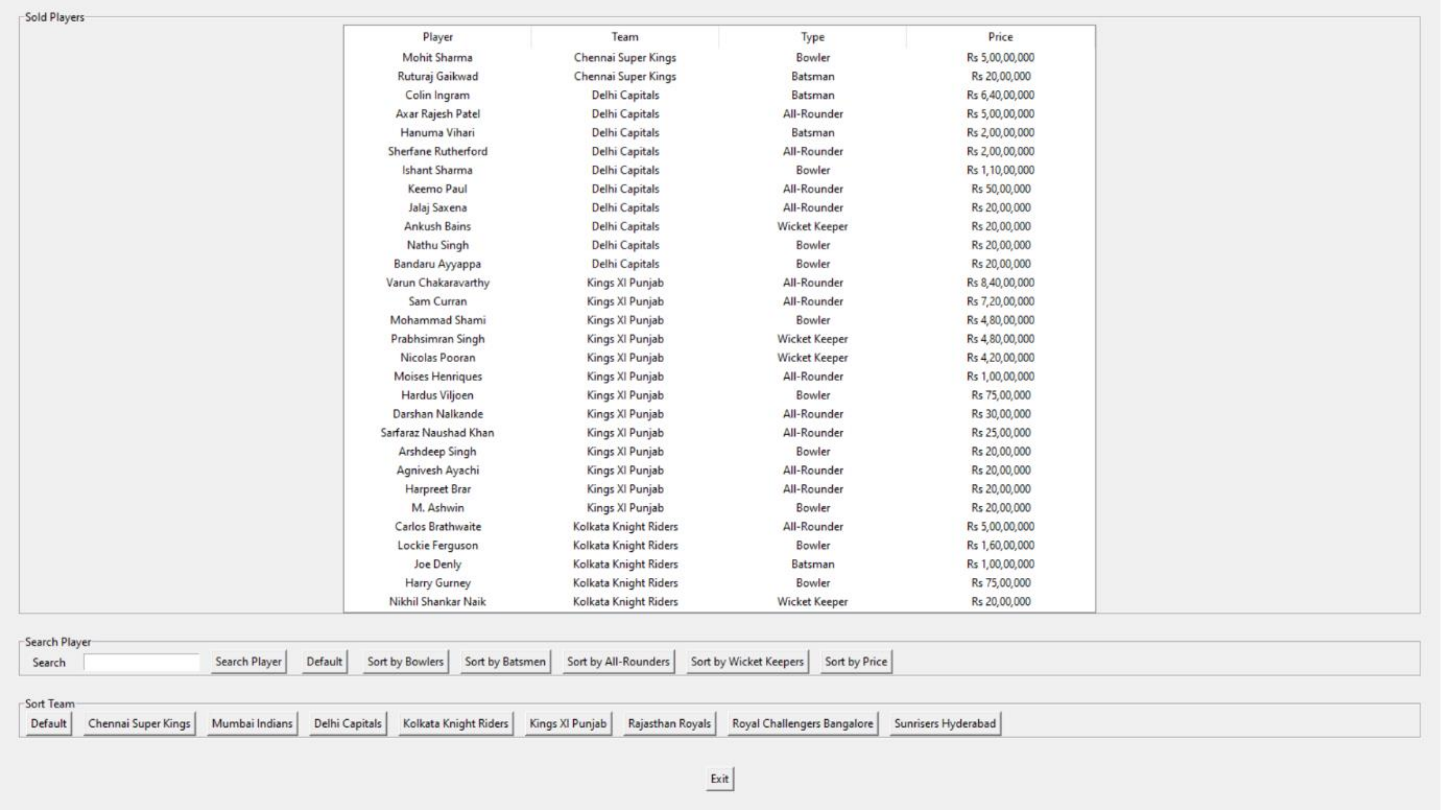
Matches	Innings	Overs	Runs	Wickets	Economy	Strike Rate	Average
17	17	64.3	482	22	7.50	17.59	21.91
11	11	42.4	352	12	8.28	21.33	29.33
13	13	48.0	397	11	8.28	26.18	36.09
6	6	23.0	210	8	9.12	17.25	26.25
9	9	33.0	286	4	8.64	49.50	71.50
3	3	10.0	109	3	10.92	20.00	36.33
3	3	9.0	90	2	10.02	27.00	45.00
3	3	12.0	85	2	7.08	36.00	42.50
1	1	4.0	28	2	7.02	12.00	14.00
11	2	2.0	21	1	10.50	12.00	21.00

vi) Auction Data Submenu:-



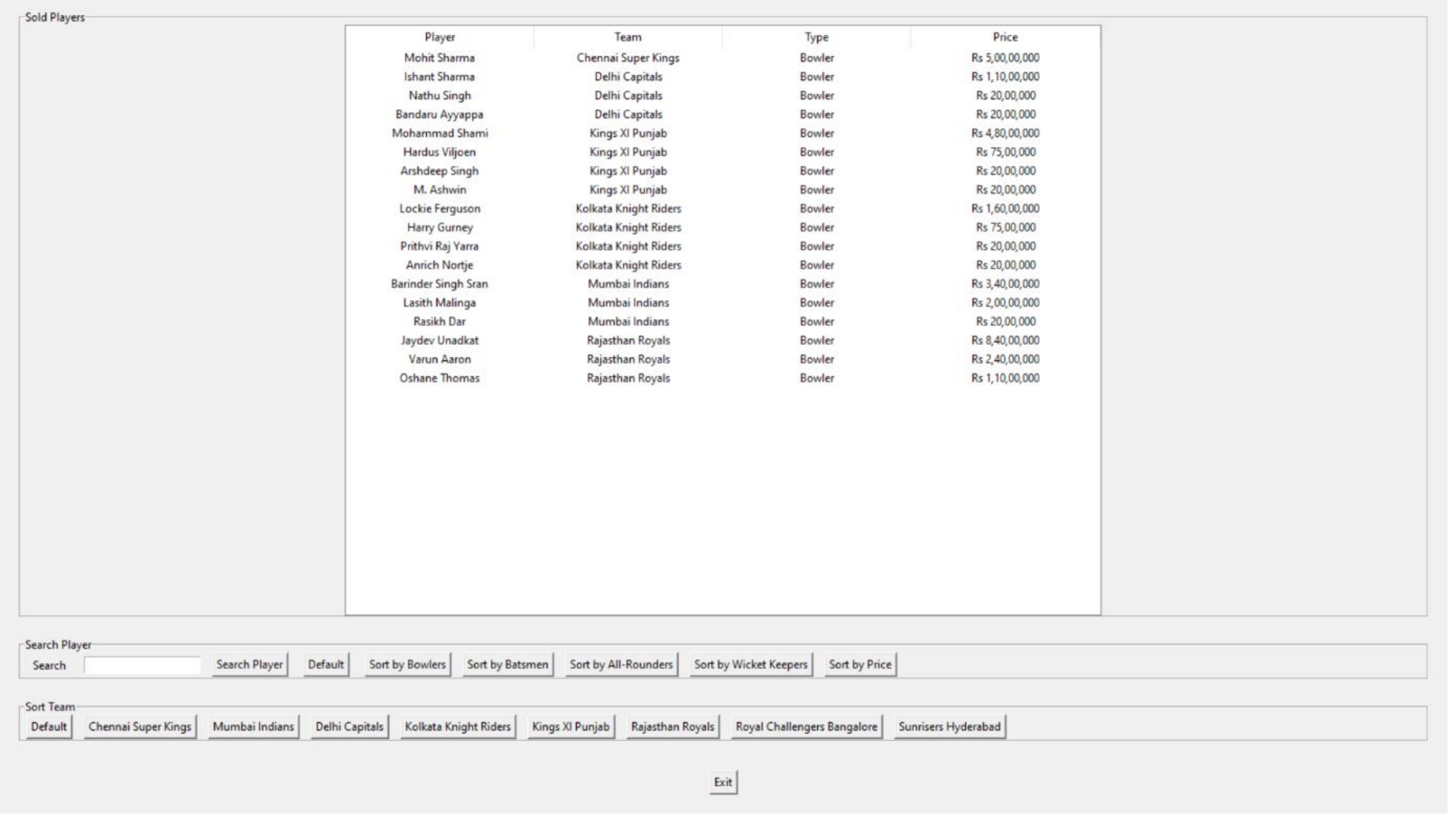
DO YOU WANT TO CHECK:

vii) Sold Players 2019(Auction Data):-



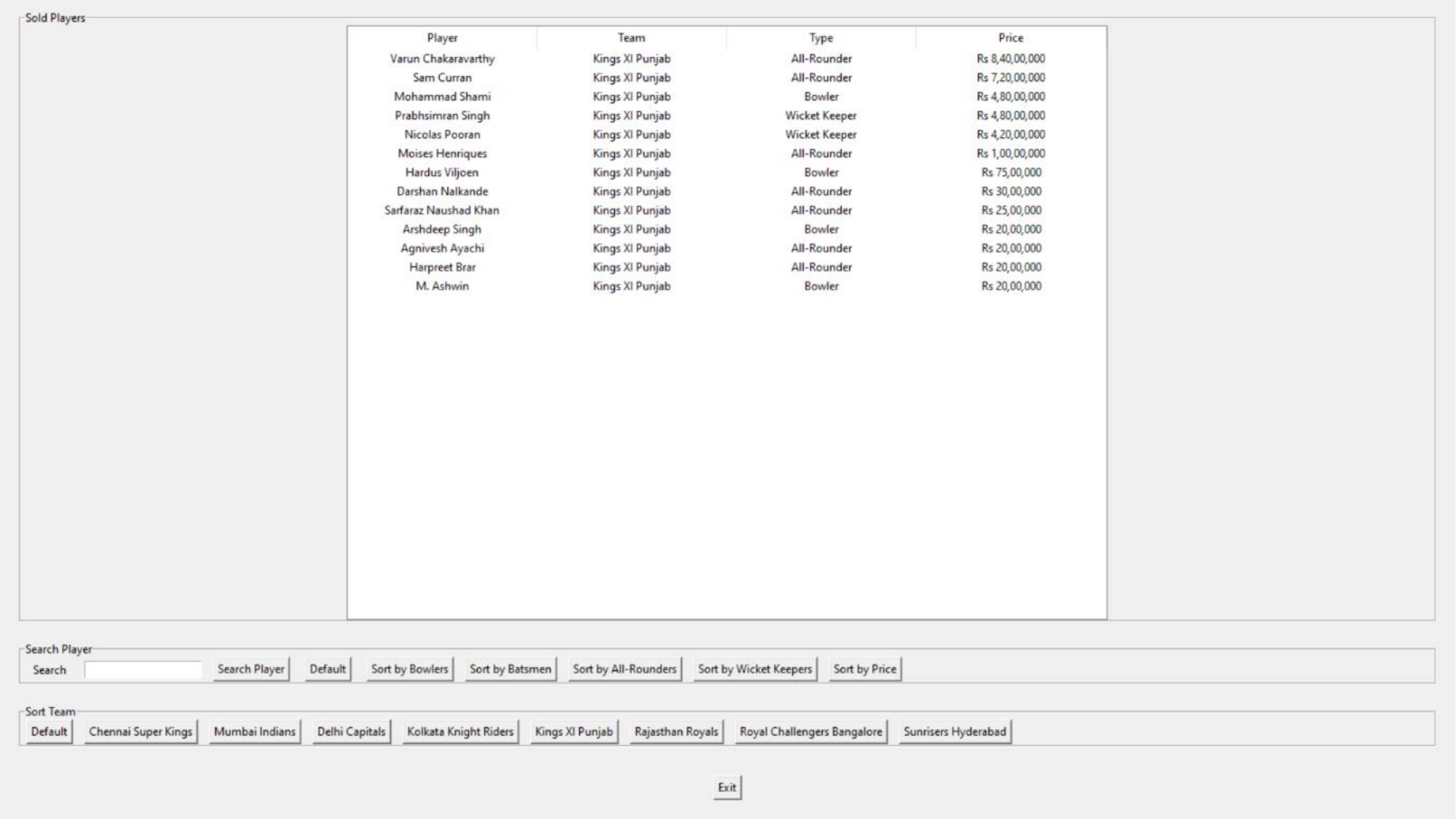
Player	Team	Type	Price
Mohit Sharma	Chennai Super Kings	Bowler	Rs 5,00,00,000
Ruturaj Gaikwad	Chennai Super Kings	Batsman	Rs 20,00,000
Colin Ingram	Delhi Capitals	Batsman	Rs 6,40,00,000
Axar Rajesh Patel	Delhi Capitals	All-Rounder	Rs 5,00,00,000
Hanuma Vihari	Delhi Capitals	Batsman	Rs 2,00,00,000
Sherfane Rutherford	Delhi Capitals	All-Rounder	Rs 2,00,00,000
Ishant Sharma	Delhi Capitals	Bowler	Rs 1,10,00,000
Keemo Paul	Delhi Capitals	All-Rounder	Rs 50,00,000
Jalaj Saxena	Delhi Capitals	All-Rounder	Rs 20,00,000
Ankush Bains	Delhi Capitals	Wicket Keeper	Rs 20,00,000
Nathu Singh	Delhi Capitals	Bowler	Rs 20,00,000
Bandaru Ayyappa	Delhi Capitals	Bowler	Rs 20,00,000
Varun Chakaravarthy	Kings XI Punjab	All-Rounder	Rs 8,40,00,000
Sam Curran	Kings XI Punjab	All-Rounder	Rs 7,20,00,000
Mohammad Shami	Kings XI Punjab	Bowler	Rs 4,80,00,000
Prabhsimran Singh	Kings XI Punjab	Wicket Keeper	Rs 4,80,00,000
Nicolas Pooran	Kings XI Punjab	Wicket Keeper	Rs 4,20,00,000
Moises Henriques	Kings XI Punjab	All-Rounder	Rs 1,00,00,000
Hardus Viljoen	Kings XI Punjab	Bowler	Rs 75,00,000
Darshan Nalkande	Kings XI Punjab	All-Rounder	Rs 30,00,000
Sarfaraz Naushad Khan	Kings XI Punjab	All-Rounder	Rs 25,00,000
Arshdeep Singh	Kings XI Punjab	Bowler	Rs 20,00,000
Agnivesh Ayachi	Kings XI Punjab	All-Rounder	Rs 20,00,000
Harpreet Brar	Kings XI Punjab	All-Rounder	Rs 20,00,000
M. Ashwin	Kings XI Punjab	Bowler	Rs 20,00,000
Carlos Brathwaite	Kolkata Knight Riders	All-Rounder	Rs 5,00,00,000
Lockie Ferguson	Kolkata Knight Riders	Bowler	Rs 1,60,00,000
Joe Denly	Kolkata Knight Riders	Batsman	Rs 1,00,00,000
Harry Gurney	Kolkata Knight Riders	Bowler	Rs 75,00,000
Nikhil Shankar Naik	Kolkata Knight Riders	Wicket Keeper	Rs 20,00,000

Sold Players 2019(Sorted by Bowlers):-



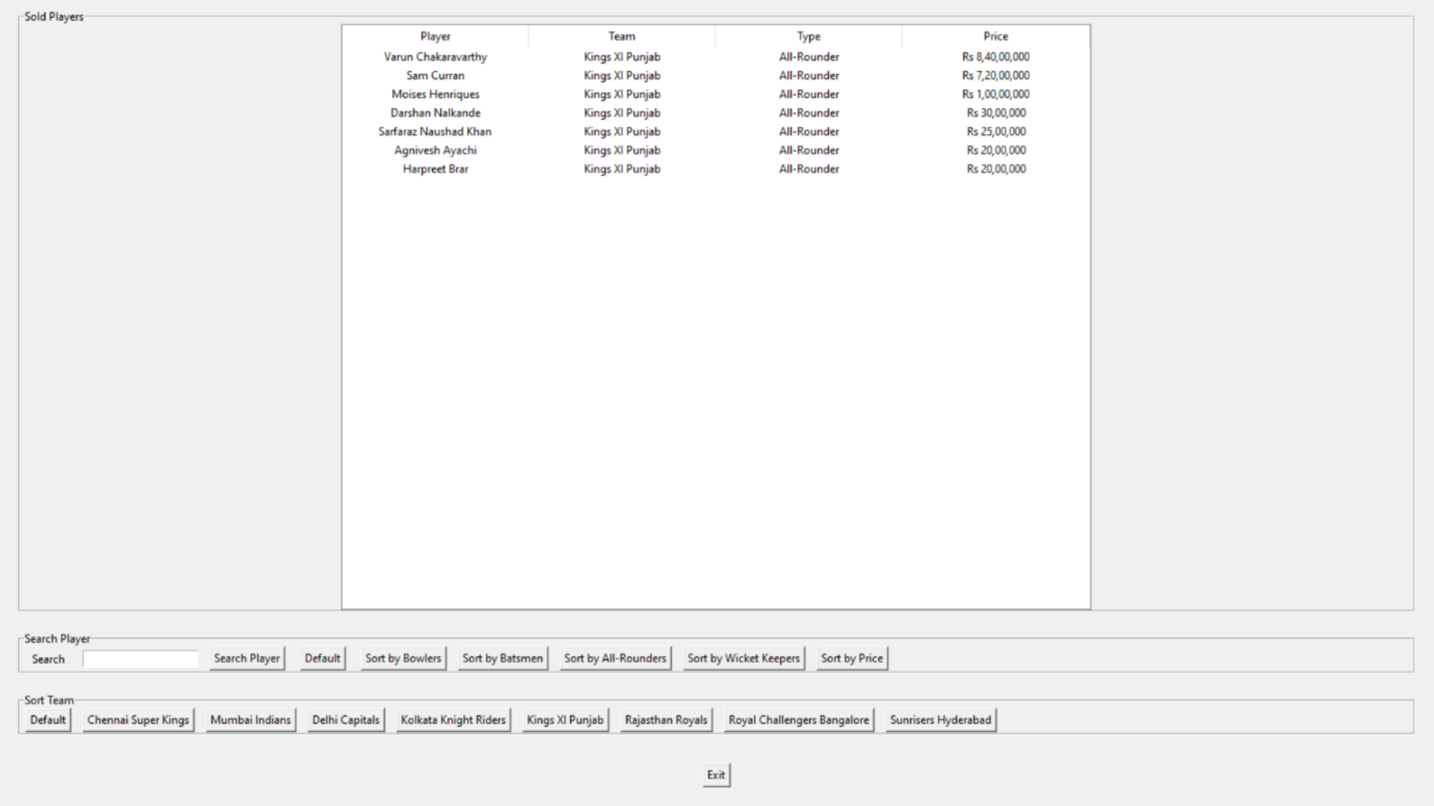
Player	Team	Type	Price
Mohit Sharma	Chennai Super Kings	Bowler	Rs 5,00,00,000
Ishant Sharma	Delhi Capitals	Bowler	Rs 1,10,00,000
Nathu Singh	Delhi Capitals	Bowler	Rs 20,00,000
Bandaru Ayyappa	Delhi Capitals	Bowler	Rs 20,00,000
Mohammad Shami	Kings XI Punjab	Bowler	Rs 4,80,00,000
Hardus Viljoen	Kings XI Punjab	Bowler	Rs 75,00,000
Arshdeep Singh	Kings XI Punjab	Bowler	Rs 20,00,000
M. Ashwin	Kings XI Punjab	Bowler	Rs 20,00,000
Lockie Ferguson	Kolkata Knight Riders	Bowler	Rs 1,60,00,000
Harry Gurney	Kolkata Knight Riders	Bowler	Rs 75,00,000
Prithvi Raj Yarra	Kolkata Knight Riders	Bowler	Rs 20,00,000
Anrich Nortje	Kolkata Knight Riders	Bowler	Rs 20,00,000
Barinder Singh Sran	Mumbai Indians	Bowler	Rs 3,40,00,000
Lasith Malinga	Mumbai Indians	Bowler	Rs 2,00,00,000
Rasikh Dar	Mumbai Indians	Bowler	Rs 20,00,000
Jaydev Unadkat	Rajasthan Royals	Bowler	Rs 8,40,00,000
Varun Aaron	Rajasthan Royals	Bowler	Rs 2,40,00,000
Oshane Thomas	Rajasthan Royals	Bowler	Rs 1,10,00,000

Sold Players 2019(sorted by team):-



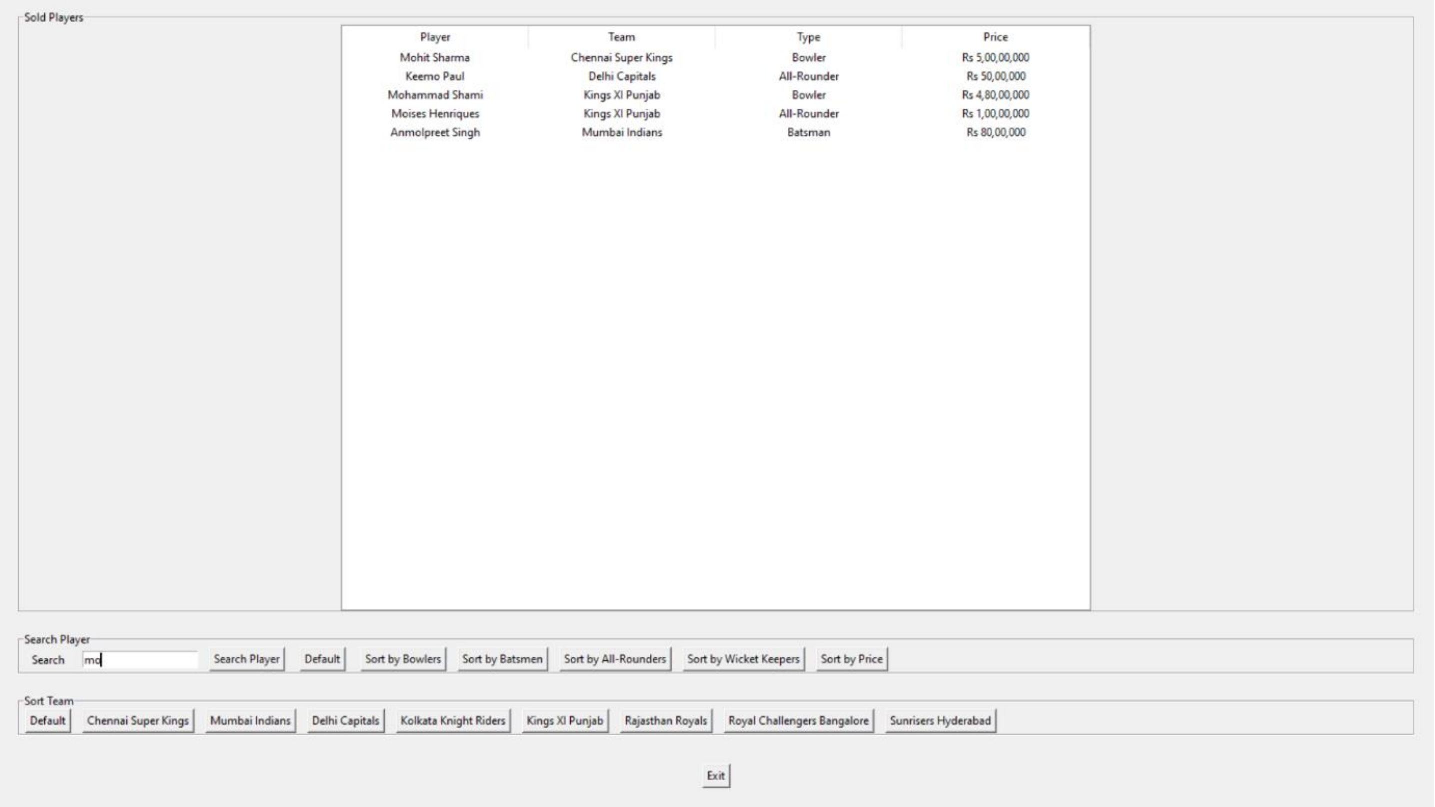
Player	Team	Type	Price
Varun Chakaravarthy	Kings XI Punjab	All-Rounder	Rs 8,40,00,000
Sam Curran	Kings XI Punjab	All-Rounder	Rs 7,20,00,000
Mohammad Shami	Kings XI Punjab	Bowler	Rs 4,80,00,000
Prabhsimran Singh	Kings XI Punjab	Wicket Keeper	Rs 4,80,00,000
Nicolas Pooran	Kings XI Punjab	Wicket Keeper	Rs 4,20,00,000
Moises Henriques	Kings XI Punjab	All-Rounder	Rs 1,00,00,000
Hardus Viljoen	Kings XI Punjab	Bowler	Rs 75,00,000
Darshan Nalkande	Kings XI Punjab	All-Rounder	Rs 30,00,000
Sarfaraz Naushad Khan	Kings XI Punjab	All-Rounder	Rs 25,00,000
Arshdeep Singh	Kings XI Punjab	Bowler	Rs 20,00,000
Agnivesh Ayachi	Kings XI Punjab	All-Rounder	Rs 20,00,000
Harpreet Brar	Kings XI Punjab	All-Rounder	Rs 20,00,000
M. Ashwin	Kings XI Punjab	Bowler	Rs 20,00,000

Sold Players 2019(sorted by team and a column):-



Player	Team	Type	Price
Varun Chakaravarthy	Kings XI Punjab	All-Rounder	Rs 8,40,00,000
Sam Curran	Kings XI Punjab	All-Rounder	Rs 7,20,00,000
Moises Henriques	Kings XI Punjab	All-Rounder	Rs 1,00,00,000
Darshan Nalkande	Kings XI Punjab	All-Rounder	Rs 30,00,000
Sarfaraz Naushad Khan	Kings XI Punjab	All-Rounder	Rs 25,00,000
Agnivesh Ayachi	Kings XI Punjab	All-Rounder	Rs 20,00,000
Harpreet Brar	Kings XI Punjab	All-Rounder	Rs 20,00,000

Sold players 2019(search by keyword):-



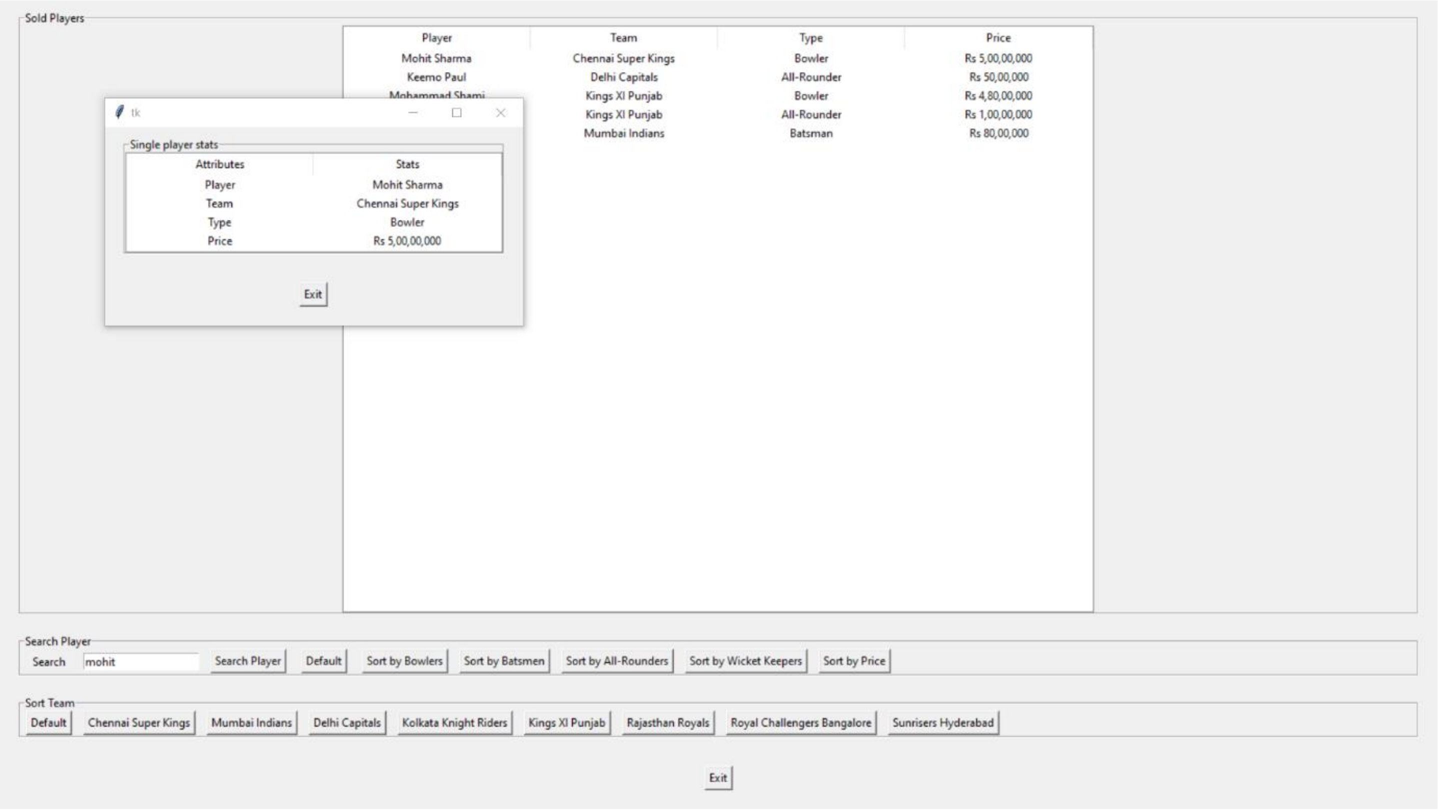
Sold Players

Player	Team	Type	Price
Mohit Sharma	Chennai Super Kings	Bowler	Rs 5,00,00,000
Keemo Paul	Delhi Capitals	All-Rounder	Rs 50,00,000
Mohammad Shami	Kings XI Punjab	Bowler	Rs 4,80,00,000
Moises Henriques	Kings XI Punjab	All-Rounder	Rs 1,00,00,000
Anmolpreet Singh	Mumbai Indians	Batsman	Rs 80,00,000

Search Player

Search: mo

Sold players 2019(search by single player):-



Sold Players

Player	Team	Type	Price
Mohit Sharma	Chennai Super Kings	Bowler	Rs 5,00,00,000
Keemo Paul	Delhi Capitals	All-Rounder	Rs 50,00,000
Mohammad Shami	Kings XI Punjab	Bowler	Rs 4,80,00,000
	Kings XI Punjab	All-Rounder	Rs 1,00,00,000
	Mumbai Indians	Batsman	Rs 80,00,000

tk

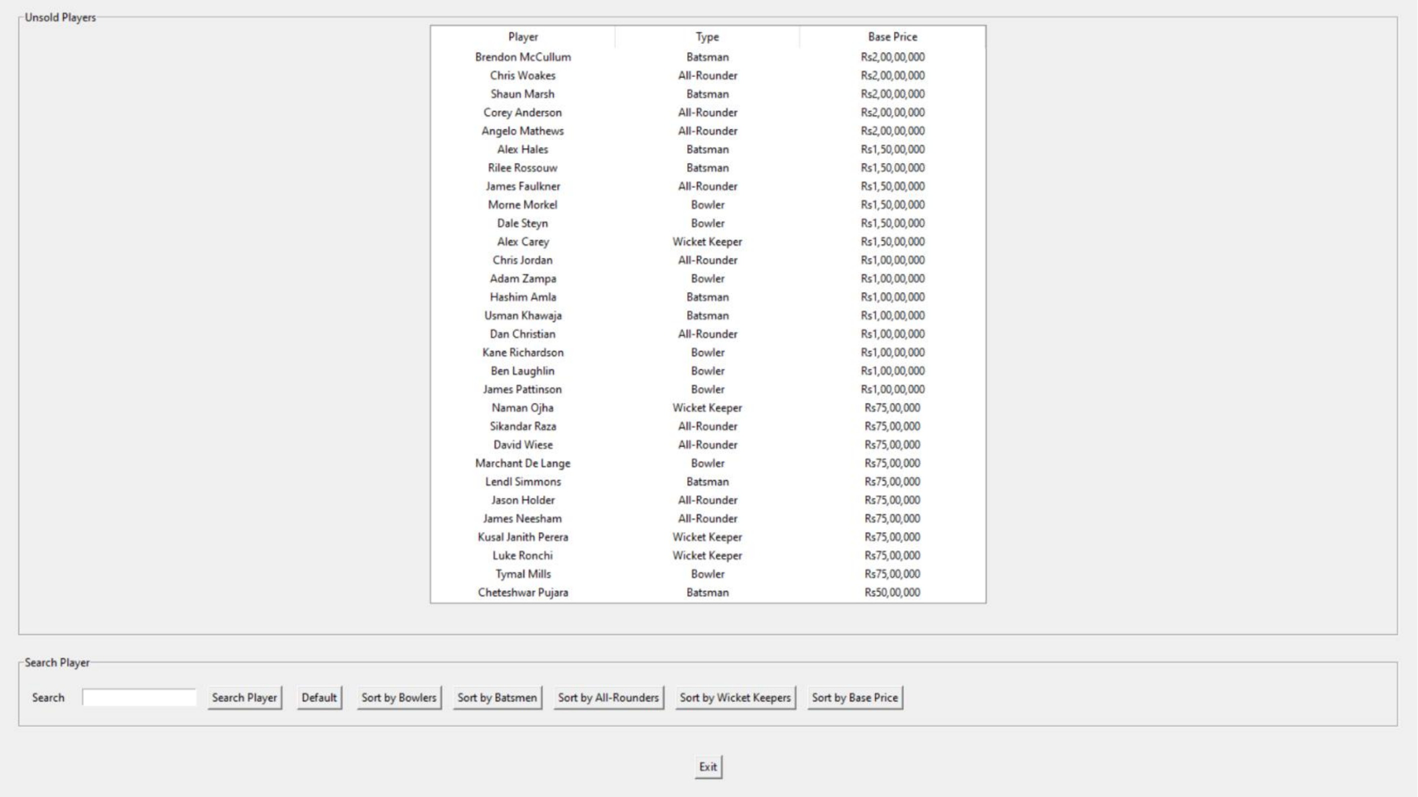
Single player stats

Attributes	Stats
Player	Mohit Sharma
Team	Chennai Super Kings
Type	Bowler
Price	Rs 5,00,00,000

Search Player

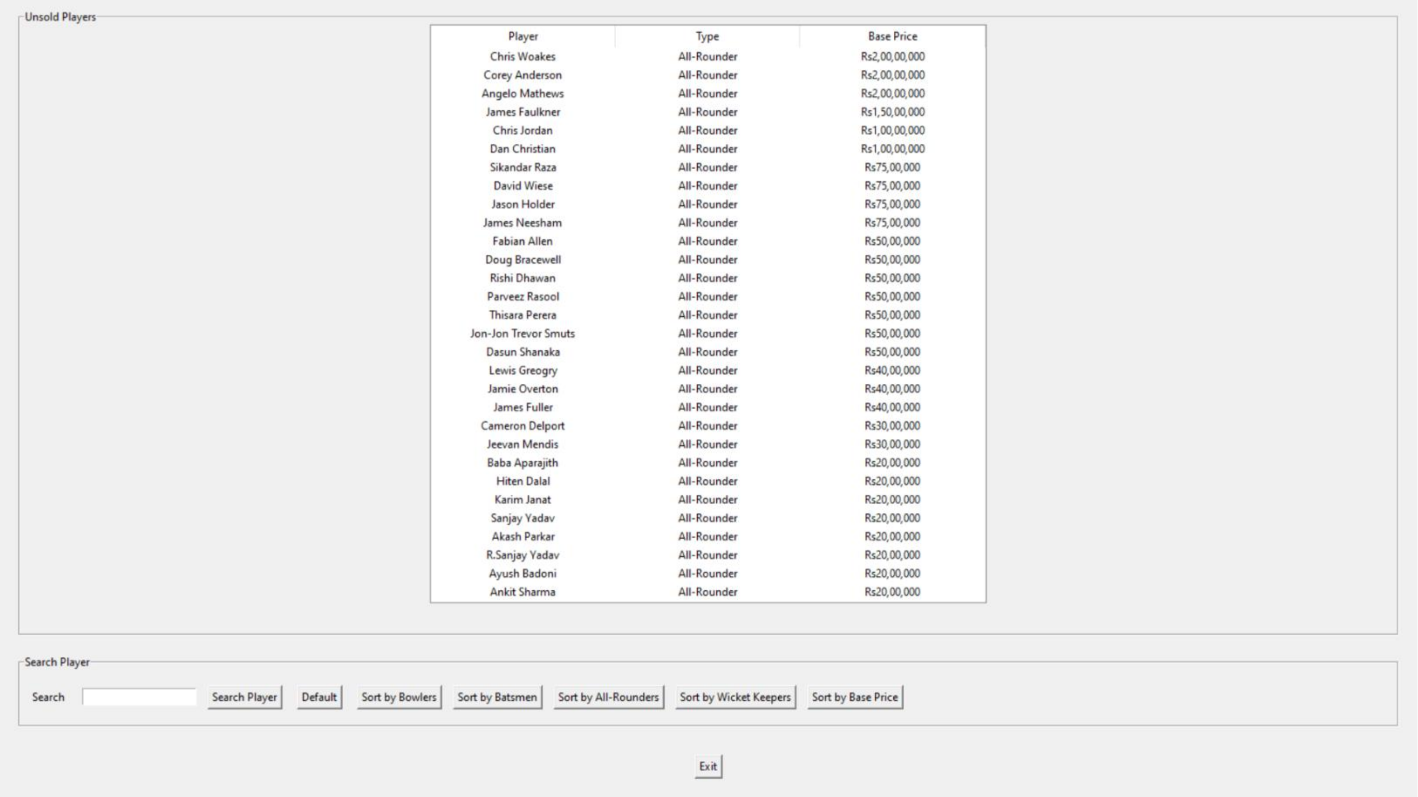
Search: mohit

viii) Unsold players 2019(Auction Data):-



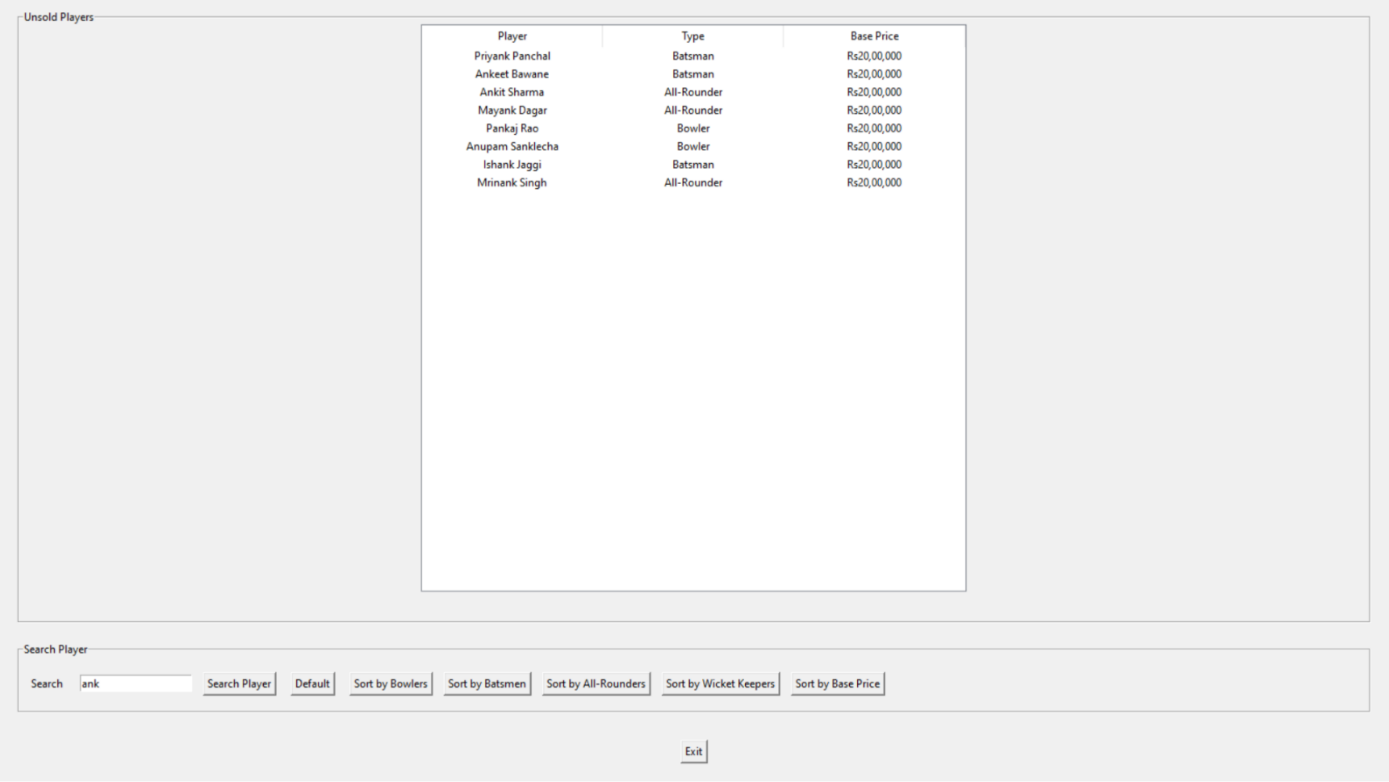
Player	Type	Base Price
Brendon McCullum	Batsman	Rs2,00,00,000
Chris Woakes	All-Rounder	Rs2,00,00,000
Shaun Marsh	Batsman	Rs2,00,00,000
Corey Anderson	All-Rounder	Rs2,00,00,000
Angelo Mathews	All-Rounder	Rs2,00,00,000
Alex Hales	Batsman	Rs1,50,00,000
Rilee Rossouw	Batsman	Rs1,50,00,000
James Faulkner	All-Rounder	Rs1,50,00,000
Morne Morkel	Bowler	Rs1,50,00,000
Dale Steyn	Bowler	Rs1,50,00,000
Alex Carey	Wicket Keeper	Rs1,50,00,000
Chris Jordan	All-Rounder	Rs1,00,00,000
Adam Zampa	Bowler	Rs1,00,00,000
Hashim Amla	Batsman	Rs1,00,00,000
Usman Khawaja	Batsman	Rs1,00,00,000
Dan Christian	All-Rounder	Rs1,00,00,000
Kane Richardson	Bowler	Rs1,00,00,000
Ben Laughlin	Bowler	Rs1,00,00,000
James Pattinson	Bowler	Rs1,00,00,000
Naman Ojha	Wicket Keeper	Rs75,00,000
Sikandar Raza	All-Rounder	Rs75,00,000
David Wiese	All-Rounder	Rs75,00,000
Marchant De Lange	Bowler	Rs75,00,000
Lendl Simmons	Batsman	Rs75,00,000
Jason Holder	All-Rounder	Rs75,00,000
James Neesham	All-Rounder	Rs75,00,000
Kusal Janith Perera	Wicket Keeper	Rs75,00,000
Luke Ronchi	Wicket Keeper	Rs75,00,000
Tymal Mills	Bowler	Rs75,00,000
Cheteshwar Pujara	Batsman	Rs50,00,000

Unsold players 2019(Sorted by All-Rounders):-



Player	Type	Base Price
Chris Woakes	All-Rounder	Rs2,00,00,000
Corey Anderson	All-Rounder	Rs2,00,00,000
Angelo Mathews	All-Rounder	Rs2,00,00,000
James Faulkner	All-Rounder	Rs1,50,00,000
Chris Jordan	All-Rounder	Rs1,00,00,000
Dan Christian	All-Rounder	Rs1,00,00,000
Sikandar Raza	All-Rounder	Rs75,00,000
David Wiese	All-Rounder	Rs75,00,000
Jason Holder	All-Rounder	Rs75,00,000
James Neesham	All-Rounder	Rs75,00,000
Fabian Allen	All-Rounder	Rs50,00,000
Doug Bracewell	All-Rounder	Rs50,00,000
Rishi Dhawan	All-Rounder	Rs50,00,000
Parveez Rasool	All-Rounder	Rs50,00,000
Thisara Perera	All-Rounder	Rs50,00,000
Jon-Jon Trevor Smuts	All-Rounder	Rs50,00,000
Dasun Shanaka	All-Rounder	Rs50,00,000
Lewis Greogry	All-Rounder	Rs40,00,000
Jamie Overton	All-Rounder	Rs40,00,000
James Fuller	All-Rounder	Rs40,00,000
Cameron Delport	All-Rounder	Rs30,00,000
Jeevan Mendis	All-Rounder	Rs30,00,000
Baba Aparajith	All-Rounder	Rs20,00,000
Hiten Dalal	All-Rounder	Rs20,00,000
Karim Janat	All-Rounder	Rs20,00,000
Sanjay Yadav	All-Rounder	Rs20,00,000
Akash Parkar	All-Rounder	Rs20,00,000
R.Sanjay Yadav	All-Rounder	Rs20,00,000
Ayush Badoni	All-Rounder	Rs20,00,000
Ankit Sharma	All-Rounder	Rs20,00,000

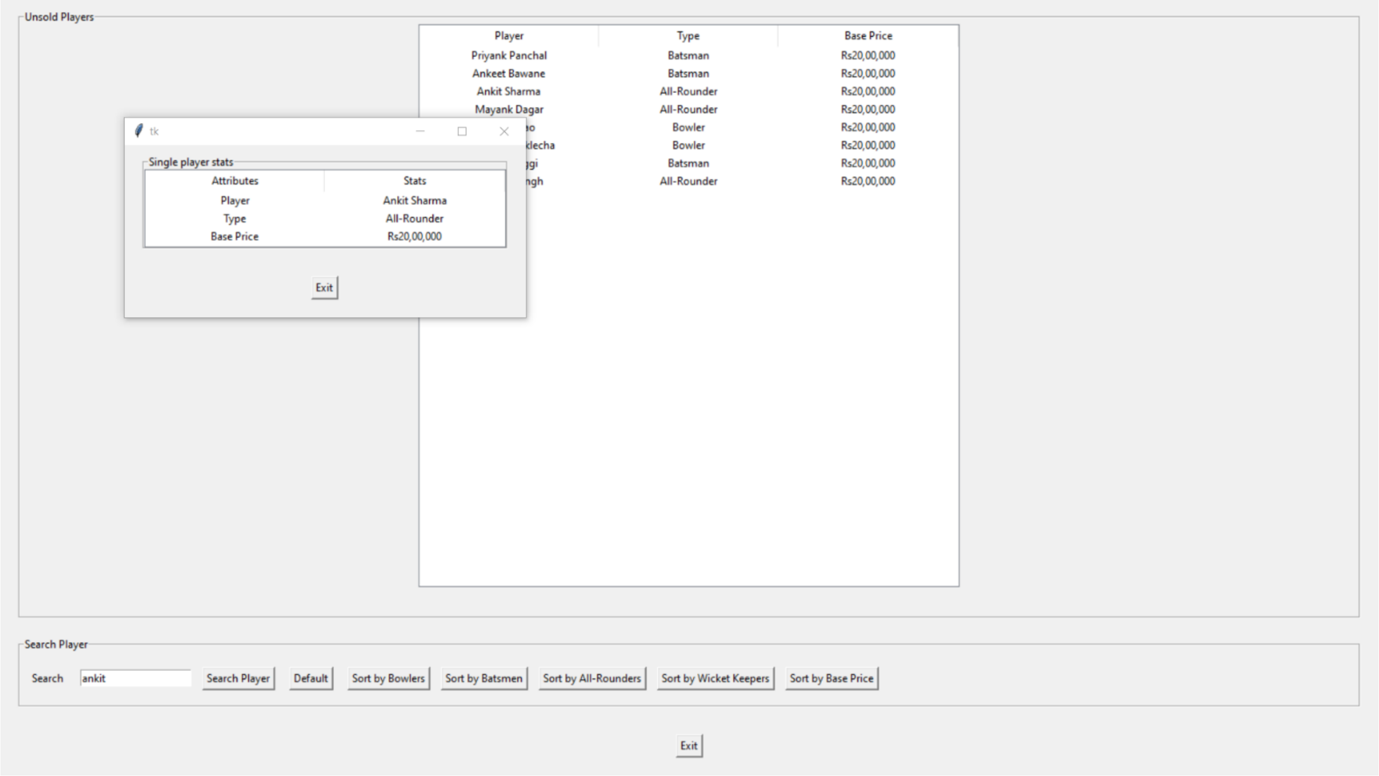
Unsold players 2019(search by keyword):-



Player	Type	Base Price
Priyank Panchal	Batsman	Rs20,00,000
Ankeet Bawane	Batsman	Rs20,00,000
Ankit Sharma	All-Rounder	Rs20,00,000
Mayank Dagar	All-Rounder	Rs20,00,000
Pankaj Rao	Bowler	Rs20,00,000
Anupam Sanklecha	Bowler	Rs20,00,000
Ishank Jaggi	Batsman	Rs20,00,000
Mrinank Singh	All-Rounder	Rs20,00,000

Search ank

Unsold players 2019(search by single player):-



Player	Type	Base Price
Priyank Panchal	Batsman	Rs20,00,000
Ankeet Bawane	Batsman	Rs20,00,000
Ankit Sharma	All-Rounder	Rs20,00,000
Mayank Dagar	All-Rounder	Rs20,00,000
ao	Bowler	Rs20,00,000
klecha	Bowler	Rs20,00,000
gi	Batsman	Rs20,00,000
ngh	All-Rounder	Rs20,00,000

Attributes	Stats
Player	Ankit Sharma
Type	All-Rounder
Base Price	Rs20,00,000

Search ankit

Bibliography

To Develop this project many references were used:-

- https://www.iplt20.com
- https://realpython.com/python-gui-tkinter/
- https://stackoverflow.com/
- https://www.geeksforgeeks.org/
- https://youtu.be/zK1H4etcb8M
- https://youtu.be/Jvul-wr-_Bg