**Coursework 3:**

Label of window

**GUI design**

**Update data**

Input box, user enters data here

Label text box

**Update Score**

**Team 1:**

**Team 1 score**

**Team 2:**

**Team 2 score**

**Exit**

**Update Score**

Update score button

Exit button

**Source Code**

import csv

def menu(): #Prints menu on screen with choices to different functions

print(""" 0.Quit

1.Display menu

2.get team information

3.get winner

4.update score""")

def createLeague(csvFile):

'''

reads the .csv file and convert the information into the list

The .csv file must be the same format with the same heading for this program to work

'''

leagueInfo = [] #creates list to store team data

num = 0

with open(csvFile, 'r') as fp: #opens .csv file

reader = csv.reader(fp,delimiter = ',') #saves data from .csv file and opens it as read mode,sperarted by comas

for team in reader:

if num > 1: # used so headings are not saved

teamName = team[0] #saves each data into relevant variable data into the list

matchPlayed = int(team[1])

wins = int(team[2])

draws=int(team[3])

loss=int(team[4])

points = (int(wins)\*3)+(int(draws))

header = [teamName,[wins,draws,loss,points,matchPlayed]]

leagueInfo.append(header) #adds team data into list

num +=1

return leagueInfo #returns list

leagueList = createLeague("europeanLeagues.csv") #creates list

def teamValidTest(leagueList,teamName):

'''

Checks if team written down is on the list

comapres name to all team on list, if not found user is asked to enter a correct name

'''

totalNumTeam = len(leagueList)

teamName=teamName

count = 0

found = False

finish = False

while not finish:

for teams in leagueList:

if teams[0].lower() == teamName.lower():

teamName = teams[0]

finish = True

count += 1

if count == totalNumTeam and not finish:

message = "Error team ", teamName, " not found(check grammer),enter a correct team?"

teamName = input(message)

count = 0

return teamName

def getTeam(leagueList,teamName):

'''

Searches through list until the team is found and returns information relevant to team

'''

found = False

for teams in leagueList:

if teams[0] == teamName:

teamInfo = teams

return teamInfo

def getWinner(leagueList):

'''

compares all teams score in list, then saves the highest score

'''

winner = ''

highestPoints = -1 #-1 i set since that is a imposible value and guanrantee a winner

for teams in leagueList:

teamPoints = int(teams[1][3])

if teamPoints == highestPoints:

if int(winner[1][2]) < int(teams[1][2]):

winner = teams

highestPoints = teamPoints

elif teamPoints > highestPoints:

winner = teams

highestPoints = teamPoints

return winner[0]

def updateScore(leagueList,team1,team1Score,team2,team2Score):

'''

updates score based on scores entered by user

'''

if team1Score > team2Score:

winner = 1 #team1 wins

elif team2Score > team1Score:

winner = 2 #team2 wins

else:

winner = 3 #draw

for teams in leagueList:

if teams[0] == team1:

teams[1][4] += 1 #updates match played

if winner == 1:

teams[1][0] +=1 #Adds 1 to wins and give 3 points to score

teams[1][3] +=3

elif winner == 2:

teams[1][2] +=1 #adds 1 to loss

else:

teams[1][1] += 1 #adds 1 to draw and adds 1 to score

teams[1][3] +=1

elif teams[0] == team2:

teams[1][4] += 1 #updates match played

if winner == 2:

teams[1][0] +=1 #Adds 1 to wins and give 3 points to score

teams[1][3] +=3

elif winner == 1:

teams[1][2] +=1 #adds 1 to loss

else:

teams[1][1] += 1 #adds 1 to draw and adds 1 to score

teams[1][3] +=1

print("Data updated")

def getValidNum(message):

valid = False

while not valid:

try:

points = int(input(message))

if points < 0:

print("enter a valid score")

else:

valid = True

except ValueError:

print("Enter a valid number")

return points

def clickUpdateScore(): #button for updating score

updateScore(leagueList,teamValidTest(leagueList,team1.get()),team1Score.get(),teamValidTest(leagueList,team2.get()),team2Score.get())

menu() #runs menu function to display menu

choice = "3"

from tkinter import \*

root = Tk() #creates the root window

while choice != "0":

choice = str(input('Enter your choice, matching the corresponding option?'))

if choice == "1": #opens menu if user choces 1

menu()

elif choice == "2": #runs getTeam function if 2 is chosen

teamName = input('enter a teamname you want to search?')

print("[Team name,[wins,draws,loses,Points,Total Match played]]")

print(getTeam(leagueList,teamValidTest(leagueList,teamName)))

elif choice =="3": #runs get winner function and displays in GUI

message ='The winning team is ',getWinner(leagueList)

root.title("Most points") #title for window

label = Label(root,text = message).grid(row = 0,column = 0) #creates lable variable

Button(root, text="Quit",command=root.quit).grid(row = 1,column = 0) #used to exit window

root.mainloop() # used to keep window visable

elif choice =="4":

#section of code no longer in use due to Gui

#team1 = teamValidTest(leagueList,input("Enter the first team?"))

#team1Score = getValidNum("enter the first teams score?")

#team2 = teamValidTest(leagueList,input("Enter the secound team?"))

#team2Score = getValidNum("enter the secound teams score?")

#updateScore(leagueList,team1,team1Score,team2,team2Score)

root.title("update Score") #titlle for window

Label(root,text="Team 1").grid(row=0,column = 0)#creates the lable

team1 = StringVar()

Entry(root,textvariable=team1).grid(row=0,column = 1) #creates the inputbox

Label(root,text="Team 1 Score").grid(row=1,column = 0)

team1Score = IntVar()

Entry(root,textvariable= team1Score).grid(row=1,column = 1)#creates the inputbox

Label(root,text="Team 2").grid(row=2,column = 0)

team2 = StringVar()

Entry(root,textvariable=team2).grid(row=2,column = 1) #creates the inputbox

Label(root,text="Team 2 Score").grid(row=3,column = 0)

team2Score = IntVar()

Entry(root,textvariable= team2Score).grid(row=3,column = 1)#creates the inputbox

button = Button(root, text ="Update Score",command=clickUpdateScore) #button to run update score function

button.grid(row=4,column=0)

Button(root, text="Quit",command=root.quit).grid(row=4,column=1) #button to exit window

root.mainloop()

elif choice == "0": #exits code

print("thank you for using the program,goodbye")

else: #if choice is invalid error mesage is displayed

print("error invalid option")

**europeanleagues.csv**

**opened in Excel:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| France | Ligue 1 |  |  |  |
| Team | Match Played | Wins | Draw | loss |
| Paris-SG | 13 | 11 | 2 | 0 |
| Lyon | 13 | 7 | 4 | 2 |
| Caen | 13 | 8 | 0 | 5 |
| Angers | 13 | 6 | 4 | 3 |
| Saint-Etienne | 13 | 7 | 1 | 5 |
| Nice | 13 | 6 | 3 | 4 |
| Lorient | 13 | 5 | 5 | 3 |
| Rennes | 13 | 5 | 5 | 3 |
| Monaco | 13 | 5 | 5 | 3 |
| Nantes | 13 | 6 | 1 | 6 |
| Bordeaux | 13 | 4 | 5 | 4 |
| Guingamp | 13 | 4 | 4 | 5 |
| Marseille | 13 | 4 | 3 | 6 |
| Reims | 13 | 4 | 3 | 6 |
| Bastia | 13 | 4 | 2 | 7 |
| Lille | 13 | 2 | 7 | 4 |
| Montpellier | 13 | 3 | 3 | 7 |
| G.F.C. Ajaccio | 13 | 3 | 3 | 7 |
| Toulouse | 13 | 1 | 6 | 6 |
| Troyes | 13 | 0 | 4 | 9 |

**Opened in notepad:**

France,Ligue 1,,,

Team,Match Played,Wins,Draw,loss

Paris-SG,13,11,2,0

Lyon,13,7,4,2

Caen,13,8,0,5

Angers,13,6,4,3

Saint-Etienne,13,7,1,5

Nice,13,6,3,4

Lorient,13,5,5,3

Rennes,13,5,5,3

Monaco,13,5,5,3

Nantes,13,6,1,6

Bordeaux,13,4,5,4

Guingamp,13,4,4,5

Marseille,13,4,3,6

Reims,13,4,3,6

Bastia,13,4,2,7

Lille,13,2,7,4

Montpellier,13,3,3,7

G.F.C. Ajaccio,13,3,3,7

Toulouse,13,1,6,6

Troyes,13,0,4,9

**Testing:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Data Used** | **Expected Result** | **Actual Results** |
| 1. Text file is read correctly and data stored in memory | europeanleagues.csv | Data is stored in a list in this format [teamName,[wins,draws,loss,points,matchPlayed]] | List format:  [teamName,[wins,draws,loss,points,matchPlayed]]  Leaguelist = [['Paris-SG', [11, 2, 0, 35, 13]], ['Lyon', [7, 4, 2, 25, 13]], ['Caen', [8, 0, 5, 24, 13]], ['Angers', [6, 4, 3, 22, 13]], ['Saint-Etienne', [7, 1, 5, 22, 13]], ['Nice', [6, 3, 4, 21, 13]], ['Lorient', [5, 5, 3, 20, 13]], ['Rennes', [5, 5, 3, 20, 13]], ['Monaco', [5, 5, 3, 20, 13]], ['Nantes', [6, 1, 6, 19, 13]], ['Bordeaux', [4, 5, 4, 17, 13]], ['Guingamp', [4, 4, 5, 16, 13]], ['Marseille', [4, 3, 6, 15, 13]], ['Reims', [4, 3, 6, 15, 13]], ['Bastia', [4, 2, 7, 14, 13]], ['Lille', [2, 7, 4, 13, 13]], ['Montpellier', [3, 3, 7, 12, 13]], ['G.F.C. Ajaccio', [3, 3, 7, 12, 13]], ['Toulouse', [1, 6, 6, 9, 13]], ['Troyes', [0, 4, 9, 4, 13]]] |
| 2. Given a team information about the team is displayed correctly | europeanleagues.csv | **Team entered:**  Lyon  **Expected output:**  Name,wins,draw,loss,points,games  Lyon,7,4,2,13 | Enter your choice, matching the corresponding option?2  enter a teamname you want to search?Lyon  [Team name,[wins,draws,loses,Points,Total Match played]]  ['Lyon', [7, 4, 2, 25, 13]] |
| **Team entered**  Nice  **Expected output:**  Nice,6,3,4,13 | Enter your choice, matching the corresponding option?2  enter a teamname you want to search?Nice  [Team name,[wins,draws,loses,Points,Total Match played]]  ['Nice', [6, 3, 4, 21, 13]] |
| **Team entered:**  Paris-SG  **Expected output:**  Paris-SG,11,2,0,13 | enter a teamname you want to search?Paris-SG  [Team name,[wins,draws,loses,Points,Total Match played]]  ['Paris-SG', [11, 2, 0, 35, 13]] |
| **Team entered:**  G.F.C. Ajaccio  **Expected output:**  G.F.C. Ajaccio,3,3,7,13 | Enter your choice, matching the corresponding option?2  enter a teamname you want to search?G.F.C. Ajaccio  [Team name,[wins,draws,loses,Points,Total Match played]]  ['G.F.C. Ajaccio', [3, 3, 7, 12, 13]] |
|  |  | **Team entered**  bastia  **Expected output**  Bastia,4,2,7,13 | Enter your choice, matching the corresponding option?2  enter a teamname you want to search?bastia  [Team name,[wins,draws,loses,Points,Total Match played]]  ['Bastia', [4, 2, 7, 14, 13]] |
|  |  | **Team entered:**  saint-etienne  **Expected result**  Saint-Etienne,7,1,5,13 | Enter your choice, matching the corresponding option?2  enter a teamname you want to search?saint-etienne  [Team name,[wins,draws,loses,Points,Total Match played]]  ['Saint-Etienne', [7, 1, 5, 22, 13]] |
| 3. If team is not in the league an error message is displayed | europeanleagues.csv | **Team entered:**  123  **Expected output:**  Error message,enter new team | Enter your choice, matching the corresponding option?2  enter a teamname you want to search?123  [Team name,[wins,draws,loses,Points,Total Match played]]  ('Error team ', '123', ' not found(check grammer),enter a correct team?') |
|  |  | **Team entered:**  wedfc  **Expected output:**  Error message | Enter your choice, matching the corresponding option?2  enter a teamname you want to search?wedfc  [Team name,[wins,draws,loses,Points,Total Match played]]  ('Error team ', 'wedfc', ' not found(check grammer),enter a correct team?') |
| 4. Invalid menu option displays an error message | europeanleagues.csv | **Entered option:**  -1  **Expected output:**  Error message | Enter your choice, matching the corresponding option?-1  error invalid option  Enter your choice, matching the corresponding option? |
|  |  | **Entered option:**  88  **Expected output:**  Error message | Enter your choice, matching the corresponding option?88  error invalid option  Enter your choice, matching the corresponding option? |
|  |  | **Entered option:**  5  **Expected output:**  Error message | Enter your choice, matching the corresponding option?5  error invalid option  Enter your choice, matching the corresponding option? |
|  |  | **Entered option:**  fev  **Expected output:**  Error message | Enter your choice, matching the corresponding option?fev  error invalid option  Enter your choice, matching the corresponding option? |
| 5. Program stops when it should | europeanleagues.csv | **Entered option:**  0  **Expected output:**  Goodbye message, exit program | Enter your choice, matching the corresponding option?0  thank you for using the program,goodbye  >>> |
| 6. Displaying winner correctly | europeanleagues.csv | **Calculation:**  Points=(wins\*3)+draw  Paris-SG=(11\*3)+2=35  **Result:**  Paris-SG has the most points   |  |  | | --- | --- | | Paris-SG | 35 | | Lyon | 25 | | Caen | 24 | | Angers | 22 | | Saint-Etienne | 22 | | Nice | 21 | | Lorient | 20 | | Rennes | 20 | | Monaco | 20 | | Nantes | 19 | | Bordeaux | 17 | | Guingamp | 16 | | Marseille | 15 | | Reims | 15 | | Bastia | 14 | | Lille | 13 | | Montpellier | 12 | | G.F.C. Ajaccio | 12 | | Toulouse | 9 | | Troyes | 4 | |  |
| 7. Data updated correctly when results of a match is introduced | europeanleagues.csv | **Entered Data**  Lyon 3 points  Caen 2 points  **Expected result**:  Lyon wins  Lyon =[8,4,2,28,14]  Caen=[8,0,6,24,14] | **before**  ['Lyon', [7, 4, 2, 25, 13]]  After  ['Lyon', [8, 4, 2, 28, 14]]  Before  ['Caen', [8, 0, 5, 24, 13]]  after  ['Caen', [8, 0, 6, 24, 14]] |
|  |  | **Entered data**  Nice 1 point  Nantes 3 points  **Expected result:**  Nantes wins  Nice = [6, 3, 5, 21, 14]  Nantes = [7, 1, 6, 22, 14] | before  ['Nice', [6, 3, 4, 21, 13]]  after  ['Nice', [6, 3, 5, 21, 14]]  before  ['Nantes', [6, 1, 6, 19, 13]]  after  ['Nantes', [7, 1, 6, 22, 14]] |
|  |  | **Entered data**  Monaco 3 points  Nantes 3 points  **Expected result:**  Draw  Monaco= [5, 6, 3, 21, 14]  Nantes= [6, 2, 6, 20, 14] | before  ['Monaco', [5, 5, 3, 20, 13]]  After  ['Monaco', [5, 6, 3, 21, 14]]  before  ['Nantes', [6, 1, 6, 19, 13]]  after  ['Nantes', [6, 2, 6, 20, 14]] |