IPL FANTASY TEAM BUILDER

Bharatiya Vidya Bhavan's

RAJAJI VIDYASHRAM

"Bhavani Campus" No. 6, Kilpauk Garden Road, Kilpauk, Chennai – 600 010

CERTIFICATE

Certified to be a bonafide record of Project work in COMPUTER SCIENCE done by

_____

__Of class XII during the year _____.

Date:

Place:

Internal Examiner

External Examiner

CONTENTS

ACKNOWLEDGEMENT

We are elated in presenting our Computer Science Project.

We express our sincere gratitude to our Principal **Shri P.G.Subramanian** and the institution for providing us with the excellent laboratory facilities for the successful completion of this project. We are extremely grateful to our computer science teachers **Mrs. K Indumathi** and **Mrs. Kalaivani G** for their guidance and valuable suggestions. We also thank the lab assistant **Mrs. Ananthi** and the software engineer **Mr. Dinesh** for their timely help. We thank our fellow classmates for the support and suggestions they gave during the project.

AIM

To create a program that allows the users to create a team of cricket players from a list of players and play a game with their respective team.

INTRODUCTION

This project deals with one of the fascinating games in the world, Cricket. This program gives the users an opportunity to create a team of players of their respective choice with a specified amount of in-game currency and enables them to play a game of “Hand Cricket” with their team to earn more points to add more players.

MODULES INCLUDED

getpass- this allows users to enter password with echo.

pyfiglet- creates ascii text art.

pickle- dump and load data files.

os- provides functions for interacting with the operating system.

pandas- Python library that is used for faster data analysis.

sys- provides functions and variables used to manipulate different parts of the Python runtime environment.

time- This module provides various time-related functions.

random- module that you can use to make random numbers

FUNCTIONS INCLUDED

showByPriceRange- sorts players by price

addToUserTeam- adds player to user's team

showUserTeam- displays user's team

showUserTeam2- displays user's team (for handcricket)

Handcricket- handcricket game

showBowlers- sorts players by role (Bowler)

showBatsmen- sorts players by role (Batsmen)

showAllRounders- sorts players by role (All Rounders)

showWicketKeepers- sorts players by role (Wicket Keepers)

showAllPlayers- displays all the players

selectPlayer- allows user to select a player to add to their team

menu2- Third menu

menu1- Second menu

mainMenu- Main Menu (Starting Menu)

printcooltext – create ascii text art

createNewUser- create a user

login- login to user account

FILES USED

- **playedDb.csv:**

  Contains all the players who played in the previous IPL tournament along with their role and price.

- **credentials.dat:**
  - Username
  - Password
  - Player's Money

- **username.txt:**

  program generates a new file with the username to store the players added to their team along with the players role and price.

PROJECT CODE

```
from getpass import getpass #Getpass Module
import pyfiglet #used to create ascii text art
import pickle
import os
import pandas as pd # Pandas Library
import sys
from time import sleep # Time module
import random

credentialsFile = "credentials.dat" #Stores user details
dbFile = "playerDb.csv" #CSV file with all player info
newUserMoney = 40  # Starting Users money
globalUserDetails = list

def printcooltext(text):
    cooltext= pyfiglet.figlet_format(text, font='doom')
    print(cooltext)

def createNewUser():
    credentialsList = []
    username = input("Username: ")
    try:
        with open(credentialsFile, 'rb+') as f:
            credentialsList = pickle.load(f)
    except EOFError:
        pass

    for userDetails in credentialsList:
        if username == userDetails[0]:
            print("Username already exists")
            return False
    password = getpass("Password: ")
    credentialsList.append([username, password, newUserMoney
])
    with open(credentialsFile, 'wb+') as f:
        pickle.dump(credentialsList, f)
        f.flush()
    global globalUserDetails
    globalUserDetails = [username, password, newUserMoney]


def login():
    credentialsList = []
    username = input("Username: ")
    try:
        with open(credentialsFile, 'rb+') as f:
            credentialsList = pickle.load(f)
    except EOFError:
        pass
    foundUser = False

    for userDetails in credentialsList:
        if username == userDetails[0]:
```

```
            foundUser = True
    if foundUser == False:
        print("Username does not exist")
        return False
    password = getpass("Password: ")
    moneyLeft = 0
    for userDetails in credentialsList:
        if username == userDetails[0]:
            if password != userDetails[1]:
                print("Incorrect password, Try again")
                return False
            elif password == userDetails[1]:
                moneyLeft = userDetails[2]
    global globalUserDetails
    globalUserDetails = [username, password, moneyLeft]
    return True

def showByPriceRange(minValue, maxValue):
    os.system('cls') # cls is used to clear screen
    playerDb = pd.read_csv(dbFile)
    playerDb['Price'] = playerDb['Price'].astype(float)
    print(playerDb[(playerDb.Price >= minValue) & (playerDb.
Price <= maxValue)])

def addToUserTeam(playerNumber):
    os.system('cls')
    playerDb = pd.read_csv(dbFile)
    playerName = ""
    pd.set_option('expand_frame_repr', False)
    with open(globalUserDetails[0] + ".txt", 'a+') as f:
        for index, row in playerDb.iterrows():
            if index == playerNumber:
                print("User has balance " + str(globalUserDe
tails[2]) + ' cr')

                if globalUserDetails[2] < row["Price"]:
                    print("You cannot buy this player, Balan
ce = " + str(globalUserDetails[2]))
                    break
                globalUserDetails[2] = globalUserDetails[2]
- row["Price"]
                print("The User's balance will now be " +  s
tr(globalUserDetails[2]) + "cr")

                playerName = str(row["Player"])
                f.write(str(row["Player"]))
                f.write("\t")
                f.write(str(row["Role"]))
                f.write("\t")
                f.write(str(row["Price"]))
                f.write("\t")
        f.write("\n")
        f.close()
```

```
    credentialsList = []
    try:
        with open(credentialsFile, 'rb+') as f:
            credentialsList = pickle.load(f)
    except EOFError:
        pass

    for i, userDetails in enumerate(credentialsList):
        if globalUserDetails[0] == userDetails[0]:
            credentialsList[i][2] = globalUserDetails[2]

    with open(credentialsFile, 'wb+') as f:
        # Using wb+ to clear and overwrite whole file
        pickle.dump(credentialsList, f)
        f.flush()

    print(playerName + " added to team")


def showUserTeam():
    os.system('cls')
    print("Username: ", globalUserDetails[0])
    print("Balance: ", globalUserDetails[2])
    with open(globalUserDetails[0] + ".txt", 'r') as f:
        lines = f.readlines()
        for line in lines:
            print(line)
        f.close()
def showUserTeam2():
    os.system('cls')
    print("Username: ", globalUserDetails[0])
    print("Choose your player:\n")
    with open(globalUserDetails[0] + ".txt", 'r') as f:
        lines = f.readlines()
        for line in lines:
            print(line)
        f.close()

def Handcricket():
    score=0
    target=0
    cheatcount=0
    status1 = 0
    status2 = 1
    ballcount=0
    uname=globalUserDetails[0]
    tosschoice=""
    random2=""
    reward=5

    userselect= input("Choose heads or tails:\n")
    random1= random.choice(['heads','tails'])
    if userselect== random1:
```

```
        os.system('cls')
        print("You won the toss!")
        sleep(2)
        os.system('cls')
        tosschoice=input("Choose to bat or bowl?\n")
    else:
        print("You lost the toss :(")
        sleep(2)
        random2 = random.choice(['bat', 'bowl'])
        os.system('cls')
        print("Opponent chose to "+ random2)
        sleep(2)
    showUserTeam2()

    userinput=input("Enter Player Name:\n")
    os.system('cls')
    if tosschoice=="bat"or random2 == "bowl":
        print("Current Batsman:", userinput)
        while status1==0:
            userchoice = int(input("Enter a no. from 1-
10:\n"))
            randomchoice = random.randint(1, 10)
            print("CPU:", randomchoice)
            if userchoice>10:
                os.system('cls')
                print("Current Score:", score)
                print("Haha, don't try to cheat\n")
                score-=userchoice
                cheatcount+=1
                if cheatcount==3:
                    actualbal-=5
                    os.system('cls')
                    sleep(2)
                    print("WOW! Why do you want to cheat so
bad?\n")
                    sleep(2)
                    print("Your balance is now reduced to:",
actualbal )
                    sleep(2)
                    menu1()
                print("If you cheat " + str(3 - cheatcount)
+ " more time(s), you will lose 5 credits!")
            ballcount+=1
            score += userchoice
            if userchoice == randomchoice:
                print("You got out :(")
                if ballcount==1:
                    score=1
                status1=1
                sleep(2)
                print("You scored:",score)
                print("Try to get the opponent out before th
ey score "+str(score)+" runs!")
```

```
        status1=0
        while status1==0:
            userchoice = int(input("Enter a no. from 1-
10:\n"))
            ballcount+=1
            randomchoice = random.randint(1, 10)
            print("CPU:", randomchoice)
            if userchoice>10:
                print("You can't enter values above 10!\n")
                print("Opponent scores "+str(randomchoice)+"
 runs this ball!")
                target+=randomchoice
            else:
                target += randomchoice
                if userchoice == randomchoice:
                    print("You got a wicket!")
                    sleep(2)
                    os.system('cls')
                    if ballcount==1:
                        target=1
                        print("You won, good job!\n")
                        sleep(2)
                        if score >= 200:
                            reward += 5

                    globalUserDetails[2]+=reward
                    print("Here's your reward:\n")
                    print("You have been rewarded " +
str(reward) + " credits!")
                    print("CurrentBalance:",
globalUserDetails[2])
                    input()
                    menu1()                         status1=1
                if target>score:
                    print("Ah... The opponent won!\n")
                    print("Better luck next time!")
                    input()
                    menu1()

    elif tosschoice=="bowl"or random2 == "bat":
        print("Current Bowler:", userinput)
        while status2==1:
            userchoice = int(input("Enter a no. from 1-
10:\n"))
            ballcount+=1
            randomchoice = random.randint(1, 10)
            print("CPU:", randomchoice)
            if userchoice>10:
                print("You can't enter values above 10!\n")
                print("Opponent scores "+str(randomchoice)+"
 runs this ball!")
                target+=randomchoice
```

```
            else:
                target += randomchoice
                if userchoice == randomchoice:
                    print("You got a wicket!")
                    sleep(2)
                    os.system('cls')
                    if ballcount==1:
                        target=1
                    status2=0
                    print("Can you chase the target?\n")
                    sleep(2)
                    print("Lets find out!\n")
                    sleep(1)
                    print("Opponent scored:",target,"\n")

        status2=1
        while status2==1:

            userchoice = int(input("Enter a no. from 1-
10\n"))
            randomchoice = random.randint(1, 10)
            print("CPU:", randomchoice)
            if userchoice > 10:
                os.system('cls')
                print("Current Score:", score)
                print("Haha, don't try to cheat\n")
                score-=userchoice
                cheatcount += 1
                if cheatcount == 3:
                    globalUserDetails[2]) -= 5
                    os.system('cls')
                    sleep(2)
                    print("WOW! Why do you want to cheat so
bad?\n")
                    sleep(2)
                    print("Your balance is now reduced to:",
 globalUserDetails[2])
                    sleep(2)
                    menu1()
                print("If you cheat " + str(3 - cheatcount)
+ " more time(s), you will lose 5 credits!")
            ballcount += 1

            score += userchoice
            if userchoice == randomchoice:
                print("You got out :(")
                status2 = 0
            if score>target:
                print("You won, good job!\n")
                if score >= 200:
                    reward += 5
                sleep(2)
                globalUserDetails[2]+=reward
```

```
                    print("Here's your reward:\n")
                    print("You have been rewarded " +
str(reward) + " credits!")
                    print("CurrentBalance:",
globalUserDetails[2])
                menu1()
            if score<target and status2==0:
                print("Oh no! You lost...\n Better luck next
 time!")
                menu1()
                if ballcount == 1:
                    score = 1
                sleep(2)
                print("You scored:", score)

def showBowlers():
    playerDb = pd.read_csv(dbFile)
    print(playerDb.loc[playerDb['Role'] == "Bowler"])

def showBatsmen():
    playerDb = pd.read_csv(dbFile)
    print(playerDb.loc[playerDb['Role'] == "Batsman"])

def showAllRounders():
    playerDb = pd.read_csv(dbFile)
    print(playerDb.loc[playerDb['Role'] == "All-Rounder"])

def showWicketKeepers():
    playerDb = pd.read_csv(dbFile)
    print(playerDb.loc[playerDb['Role'] == "Wicket Keeper"])

def showAllPlayers():
    playerDb = pd.read_csv(dbFile)
    print(playerDb)

def selectPlayer():
    playerNumber = int(input("Enter player number:\n"))
    addToUserTeam(playerNumber)
    sleep(3)

# After menu1 you get sent to menu2
def menu2():
    os.system('cls')
    option = int(input("Options\n1: Sort by value\n2: Sort b
y role\n3: Show all players\n4: Exit\n"))
    if option == 1:
        minVal = float(input("Enter min val( 0.1-
17 cr):\n"))
        maxVal = float(input("Enter max val ( 0.1-
17 cr):\n"))
        showByPriceRange(minVal, maxVal)
        selectPlayer()
    elif option == 2:
```

```
        playerRole = int(input("1: Bowler\n2: Batsman\n3: Al
l-Rounder\n4: Wicket Keeper\n"))
        if playerRole == 1:
            showBowlers()
            selectPlayer()
        elif playerRole == 2:
            showBatsmen()
            selectPlayer()
        elif playerRole == 3:
            showAllRounders()
            selectPlayer()
        elif playerRole == 4:
            showWicketKeepers()
            selectPlayer()
        else:
            print("Invalid entry, try again")
            sleep(2)
            menu2()
    elif option == 3:
        showAllPlayers()
        selectPlayer()
    elif option == 4:
        menu1()
    else:
        print("Invalid entry, try again")
        sleep(2)
        menu2()
    menu1()

# After main menu you get sent to menu1
def menu1():
    os.system('cls')
    option = int(input("Options\n1: Add to team\n2: View tea
m\n3: Play a game\n4: Exit\n"))
    if option == 1:
        menu2()
    elif option == 2:
        showUserTeam()
        input()
        menu1()
    elif option == 3:
        Handcricket()

    elif option == 4:
        os.system("cls")
        sys.exit()

    else:
        print("Invalid entry, try again")
        sleep(2)
        menu1()

# This is the starting Menu
```

```
def mainMenu():
    os.system('cls')
    option = int(input("Options:\n1: Login\n2: Create New Ac
count\n3: Exit\n"))
    if option == 1:
        if login() == False:
            print("Login failed, Try again or create new use
r")
            input()
            mainMenu()
        else:
            print("Login success, Press Enter to continue")
            input()

            menu1()
    elif option == 2:
        if createNewUser() == False:
            print("User creation failed, try again or login
to continue")
            input()
            mainMenu()
        else:
            print("User created successfully, login to conti
nue")
            input()
            mainMenu()
    elif option == 3:
        os.system('cls')
        sys.exit()
    else:
        print("Invalid entry, try again")
        sleep(2)
        mainMenu()

if __name__ == "__main__":
    pd.set_option('display.max_rows', None)
    pd.set_option('expand_frame_repr', False)

    # Create file if it does not exist
    with open(credentialsFile, 'ab+') as fp:
        pass
    os.system('cls')
    printcooltext("Welcome To IPL Fantasy Team Builder !")
    sleep(4)
    os.system('cls')
    mainMenu()
```

OUTPUTS

SUGGESTED IMPROVEMENTS

- Add option to sell players.
- Add option to reset password.
- Do checking of player in team when choosing the player for Hand Cricket.

BIBLIOGRAPHY

1. COMPUTER SCIENCE WITH PYTHON CLASS 11
BY SUMITA ARORA

2. COMPUTER SCIENCE WITH PYTHON CLASS 12
BY SUMITA ARORA

3. stackoverflow.com