**Python Codes**

!pip install requests

!pip install beautifulsoup4

!pip install csv

import requests

from bs4 import BeautifulSoup

import csv

url = "https://fbref.com/en/squads/822bd0ba/Liverpool-Stats"

response = requests.get(url)

content = response.content

soup = BeautifulSoup(content, 'html.parser')

table = soup.find('table', class\_='stats\_table')

if table is not None:

    headers = table.find('thead').find\_all('th')

    column\_names = [header.text.strip() for header in headers]

    rows = table.find('tbody').find\_all('tr')

    player\_data = []

    for row in rows:

        player\_row = [data.text.strip() for data in row.find\_all('td')]

        player\_data.append(player\_row)

    # Save data to CSV file

    filename = "liverpool\_stats.csv"

    with open(filename, 'w', newline='', encoding='utf-8') as csvfile:

        writer = csv.writer(csvfile)

        writer.writerow(column\_names)

        writer.writerows(player\_data)

    print("Data saved to", filename)

else:

    print("Table not found on the webpage.")

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

dt = pd.read\_csv("liverpool\_stats.csv")

dt

dt.head()

dt.info()

dt.describe()

dt.isnull()

dt.count

# Create a correlation heatmap

plt.figure(figsize=(10, 8))

sns.heatmap(dt.corr(), annot=True, cmap='coolwarm')

plt.title("Correlation Heatmap")

plt.show()

# Plotting analysis

plt.figure(figsize=(10, 6))

plt.scatter(dt['Age'], dt['Gls'], color='red')

plt.xlabel("Age")

plt.ylabel("Gls")

plt.title("Age vs Goals")

plt.show()

plt.figure(figsize=(10, 6))

sns.boxplot(x=dt['Nation'], y=dt['Ast'])

plt.xlabel("Nation")

plt.ylabel("Ast")

plt.title("Nation vs Assists")

plt.show()

# Calculate POSITIVE accuracy

accuracy = len(dt[dt['Player'] == 'DF']) / len(dt) \* 100

print("Accuracy (DF):", accuracy, "%")

accuracy = len(dt[dt['Player'] == 'MF']) / len(dt) \* 100

print("Accuracy (MF):", accuracy, "%")

import scipy.stats as stats

# Perform inferential statistics

nation\_counts = dt['Nation'].value\_counts()

print("Nation Counts:")

print(nation\_counts)

# Perform t-test between 'Goals' of English (ENG) and Non-English players

eng\_goals = dt[dt['Nation'] == 'ENG']['Gls']

non\_eng\_goals = dt[dt['Nation'] != 'ENG']['Gls']

t\_stat, p\_value = stats.ttest\_ind(eng\_goals, non\_eng\_goals)

print("T-Test Results:")

print("T-Statistic:", t\_stat)

print("P-Value:", p\_value)

# Summary graph

plt.figure(figsize=(10, 6))

sns.countplot(x='Pos', data=dt)

plt.xlabel("Pos")

plt.ylabel("Count")

plt.title("Distribution of Players by Position")

plt.show()