In [9]:

```
# File: 1_data_collection.ipynb

# Import necessary libraries
import requests  # For fetching HTML content
from bs4 import BeautifulSoup  # For parsing HTML
import pandas as pd  # For structuring data
import os  # For managing file paths
import time  # For adding delays between requests

# Define seasons and generate URLs
base_url = "https://fbref.com/en/comps/9/{season}/schedule/{season}-Premie
seasons = ["2018-2019", "2019-2020", "2020-2021", "2021-2022", "2022-2023"
urls = [base_url.format(season=season) for season in seasons]

# Print URLs to verify
print("Generated URLs:")
for url in urls:
    print(url)
```

```
Generated URLs:
https://fbref.com/en/comps/9/2018-2019/schedule/2018-2019-Premier-League-
Scores-and-Fixtures (https://fbref.com/en/comps/9/2018-2019/schedule/2018
-2019-Premier-League-Scores-and-Fixtures)
https://fbref.com/en/comps/9/2019-2020/schedule/2019-2020-Premier-League-
Scores-and-Fixtures (https://fbref.com/en/comps/9/2019-2020/schedule/2019
-2020-Premier-League-Scores-and-Fixtures)
https://fbref.com/en/comps/9/2020-2021/schedule/2020-2021-Premier-League-
Scores-and-Fixtures (https://fbref.com/en/comps/9/2020-2021/schedule/2020
-2021-Premier-League-Scores-and-Fixtures)
https://fbref.com/en/comps/9/2021-2022/schedule/2021-2022-Premier-League-
Scores-and-Fixtures (https://fbref.com/en/comps/9/2021-2022/schedule/2021
-2022-Premier-League-Scores-and-Fixtures)
https://fbref.com/en/comps/9/2022-2023/schedule/2022-2023-Premier-League-
Scores-and-Fixtures (https://fbref.com/en/comps/9/2022-2023/schedule/2022
-2023-Premier-League-Scores-and-Fixtures)
https://fbref.com/en/comps/9/2023-2024/schedule/2023-2024-Premier-League-
Scores-and-Fixtures (https://fbref.com/en/comps/9/2023-2024/schedule/2023
-2024-Premier-League-Scores-and-Fixtures)
```

In [4]:

```
# Test fetching HTML content for one season
test_url = urls[0]  # Use the first season's URL for testing
response = requests.get(test_url)

# Check if the request was successful
if response.status_code == 200:
    print("Successfully fetched HTML content!")
else:
    print(f"Failed to fetch HTML content. Status code: {response.status_co

# Print the first 500 characters of the HTML to verify
print("First 500 characters of the HTML:")
print(response.text[:500])
```

```
Successfully fetched HTML content!
First 500 characters of the HTML:


<!DOCTYPE html>
<html data-version="klecko-" data-root="/home/fb/deploy/www/base" lang="e
n" class="no-js" >
<head>
    <meta charset="utf-8">
    <meta http-equiv="x-ua-compatible" content="ie=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0,
maximum-scale=2.0" />
    <link rel="dns-prefetch" href="https://cdn.ssref.net/req/202504030" /
>
<script>
/* https://docs.osano.com/hc/en-us/articles/22469433444372-Google-Consent
-Mode-v2 (https://docs.osano.com/hc/en-us/articles/22469433444372-Google-
Consent-Mode-v2)  */
  window.dataLayer = w
```

In [5]:

```
# Parse HTML and locate the table
soup = BeautifulSoup(response.content, 'html.parser')
table = soup.find('table', {'class': 'stats_table'})

# Print the first few rows of the table to verify
if table:
    print("Table found!")
    print(table.find_all('tr')[:3])  # Print the first 3 rows
else:
    print("No table found.")
```

```
Table found!
[<tr> <th aria-label="Matchweek Number" class="poptip sort_default_asc ce
nter" data-stat="gameweek" data-tip="&lt;strong&gt;Matchweek Number&lt;/s
trong&gt;&lt;br&gt;Matchweek Number" scope="col">Wk</th> <th aria-label
="Day" class="poptip sort_default_asc center" data-stat="dayofweek" data-
tip="Day of week" scope="col">Day</th> <th aria-label="Date" class="popti
p sort_default_asc center" data-stat="date" data-tip="Date listed is loca
l to the match" scope="col">Date</th> <th aria-label="Time" class="poptip
sort_default_asc center" data-stat="start_time" data-tip="Time listed is
local to the match venue&lt;br&gt;Time is written in the 24-hour notation
&lt;br&gt;Your local time is in (·) " scope="col">Time</th> <th aria-labe
l="Home" class="poptip sort_default_asc center" data-stat="home_team" sco
pe="col">Home</th> <th aria-label="xG: Expected Goals" class="poptip cent
er" data-filter="1" data-name="xG: Expected Goals" data-stat="home_xg" da
ta-tip="&lt;strong&gt;xG: Expected Goals&lt;/strong&gt;&lt;br&gt;Expected
Goals&lt;br&gt;xG totals include penalty kicks, but do not include penalt
y shootouts (unless otherwise noted).&lt;br&gt;Provided by Opta.&lt;br&g
t;An underline indicates there is a match that is missing data, but will
be updated when available." scope="col">xG</th> <th aria-label="Score" cl
ass="poptip center" data-stat="score" data-tip="Numbers in parentheses in
dicate goals scored in penalty shootout" scope="col">Score</th> <th aria-
label="xG: Expected Goals" class="poptip center" data-filter="1" data-nam
e="xG: Expected Goals" data-stat="away_xg" data-tip="&lt;strong&gt;xG: Ex
pected Goals&lt;/strong&gt;&lt;br&gt;Expected Goals&lt;br&gt;xG totals in
clude penalty kicks, but do not include penalty shootouts (unless otherwi
se noted).&lt;br&gt;Provided by Opta.&lt;br&gt;An underline indicates the
re is a match that is missing data, but will be updated when available."
scope="col">xG</th> <th aria-label="Away" class="poptip sort_default_asc
center" data-stat="away_team" scope="col">Away</th> <th aria-label="Atten
dance" class="poptip center" data-stat="attendance" scope="col">Attendanc
e</th> <th aria-label="Venue" class="poptip sort_default_asc center" data
-stat="venue" scope="col">Venue</th> <th aria-label="Referee" class="popt
ip sort_default_asc center" data-stat="referee" scope="col">Referee</th>
<th aria-label="Match Report" class="poptip center" data-stat="match_repo
rt" scope="col">Match Report</th> <th aria-label="Notes" class="poptip ce
nter" data-stat="notes" scope="col">Notes</th> </tr>, <tr><th class="righ
t" data-stat="gameweek" scope="row">1</th><td class="left" csk="6" data-s
tat="dayofweek">Fri</td><td class="left" csk="20180810" data-stat="date">
<a href="/en/matches/2018-08-10">2018-08-10</a></td><td class="right" csk
="20:00:00" data-stat="start_time"><span class="venuetime" data-venue-epo
ch="1533927600" data-venue-time="20:00" data-venue-time-only="1">20:00</s
pan> <span class="localtime" data-label="your time"></span></td><td class
="right" data-stat="home_team" style="font-weight: bold;"><a href="/en/sq
uads/19538871/2018-2019/Manchester-United-Stats">Manchester Utd</a></td><
td class="right" data-stat="home_xg">1.5</td><td class="center" data-stat
="score"><a href="/en/matches/3ae83896/Manchester-United-Leicester-City-A
ugust-10-2018-Premier-League">2–1</a></td><td class="right" data-stat="aw
ay_xg">1.8</td><td class="left" data-stat="away_team"><a href="/en/squad
s/a2d435b3/2018-2019/Leicester-City-Stats">Leicester City</a></td><td cla
ss="right" csk="74439" data-stat="attendance">74,439</td><td class="left"
data-stat="venue">Old Trafford</td><td class="left" csk="Andre Marriner20
18-08-10" data-stat="referee">Andre Marriner</td><td class="left" data-st
at="match_report"><a href="/en/matches/3ae83896/Manchester-United-Leicest
er-City-August-10-2018-Premier-League">Match Report</a></td><td class="le
ft iz" data-stat="notes"></td></tr>, <tr><th class="right sort_show" data
-stat="gameweek" scope="row">1</th><td class="left" csk="7" data-stat="da
yofweek">Sat</td><td class="left" csk="20180811" data-stat="date"><a href
="/en/matches/2018-08-11">2018-08-11</a></td><td class="right" csk="12:3
0:00" data-stat="start_time"><span class="venuetime" data-venue-epoch="15
33987000" data-venue-time="12:30" data-venue-time-only="1">12:30</span> <
span class="localtime" data-label="your time"></span></td><td class="righ
```

```
t" data-stat="home_team"><a href="/en/squads/b2b47a98/2018-2019/Newcastle
-United-Stats">Newcastle Utd</a></td><td class="right" data-stat="home_x
g">1.0</td><td class="center" data-stat="score"><a href="/en/matches/dce1
5e01/Newcastle-United-Tottenham-Hotspur-August-11-2018-Premier-League">1–
2</a></td><td class="right" data-stat="away_xg">2.0</td><td class="left"
data-stat="away_team" style="font-weight: bold;"><a href="/en/squads/361c
a564/2018-2019/Tottenham-Hotspur-Stats">Tottenham</a></td><td class="righ
t" csk="51749" data-stat="attendance">51,749</td><td class="left" data-st
at="venue">St. James' Park</td><td class="left" csk="Martin Atkinson2018-
08-11" data-stat="referee">Martin Atkinson</td><td class="left" data-stat
="match_report"><a href="/en/matches/dce15e01/Newcastle-United-Tottenham-
Hotspur-August-11-2018-Premier-League">Match Report</a></td><td class="le
ft iz" data-stat="notes"></td></tr>]
```

In [9]:

```
# Extract column headers and remove duplicates
headers = []
for header in table.find_all('th'):
    if 'data-stat' in header.attrs:
        header_name = header['data-stat']
        if header_name not in headers:  # Add only if it's not already in
            headers.append(header_name)

print("Column headers:", headers)
```

```
Column headers: ['gameweek', 'dayofweek', 'date', 'start_time', 'home_tea
m', 'home_xg', 'score', 'away_xg', 'away_team', 'attendance', 'venue', 'r
eferee', 'match_report', 'notes']
```

In [11]:

```
# Extract rows
rows = []
for row in table.find_all('tr')[1:]:  # Skip header row
    cells = [cell.text.strip() for cell in row.find_all(['td', 'th'])]
    rows.append(cells)

print("First few rows of data:")
print(rows[:3])
```

```
First few rows of data:
[['1', 'Fri', '2018-08-10', '20:00', 'Manchester Utd', '1.5', '2–1', '1.
8', 'Leicester City', '74,439', 'Old Trafford', 'Andre Marriner', 'Match
Report', ''], ['1', 'Sat', '2018-08-11', '12:30', 'Newcastle Utd', '1.0',
'1–2', '2.0', 'Tottenham', '51,749', "St. James' Park", 'Martin Atkinso
n', 'Match Report', ''], ['1', 'Sat', '2018-08-11', '15:00', 'Fulham',
'0.7', '0–2', '1.0', 'Crystal Palace', '24,821', 'Craven Cottage', 'Mike
Dean', 'Match Report', '']]
```

In [12]:

```
# Save data to CSV
df = pd.DataFrame(rows, columns=headers)
raw_data_dir = r"C:\Users\matth\OneDrive\Documents\data_science_project\pr
os.makedirs(raw_data_dir, exist_ok=True)

# Save the data for the test season
file_path = os.path.join(raw_data_dir, "2018-2019.csv")
df.to_csv(file_path, index=False)
print(f"Data saved to {file_path}")
```

```
Data saved to C:\Users\matth\OneDrive\Documents\data_science_project\prem
ier-league-home-advantage\data\raw_data\2018-2019.csv
```

In [13]:

```
# Scrape data for all seasons
for url in urls:
    # Fetch HTML content
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'html.parser')

    # Locate the table
    table = soup.find('table', {'class': 'stats_table'})

    # Extract headers
    headers = []
    for header in table.find_all('th'):
        if 'data-stat' in header.attrs:
            header_name = header['data-stat']
            if header_name not in headers:
                headers.append(header_name)

    # Extract rows
    rows = []
    for row in table.find_all('tr')[1:]:  # Skip header row
        cells = [cell.text.strip() for cell in row.find_all(['td', 'th'])]
        rows.append(cells)

    # Create DataFrame and save to CSV
    df = pd.DataFrame(rows, columns=headers)
    season = url.split('/')[-3]
    file_path = os.path.join(raw_data_dir, f"{season}.csv")
    df.to_csv(file_path, index=False)
    print(f"Successfully scraped {season} and saved to {file_path}")

    # Add delay to avoid overloading server
    time.sleep(3)
```

```
Successfully scraped 2018-2019 and saved to C:\Users\matth\OneDrive\Documents\data_science_project\premier-league-home-advantage\data\raw_data\2018-2019.csv
Successfully scraped 2019-2020 and saved to C:\Users\matth\OneDrive\Documents\data_science_project\premier-league-home-advantage\data\raw_data\2019-2020.csv
Successfully scraped 2020-2021 and saved to C:\Users\matth\OneDrive\Documents\data_science_project\premier-league-home-advantage\data\raw_data\2020-2021.csv
Successfully scraped 2021-2022 and saved to C:\Users\matth\OneDrive\Documents\data_science_project\premier-league-home-advantage\data\raw_data\2021-2022.csv
Successfully scraped 2022-2023 and saved to C:\Users\matth\OneDrive\Documents\data_science_project\premier-league-home-advantage\data\raw_data\2022-2023.csv
Successfully scraped 2023-2024 and saved to C:\Users\matth\OneDrive\Documents\data_science_project\premier-league-home-advantage\data\raw_data\2023-2024.csv
```

In [ ]: