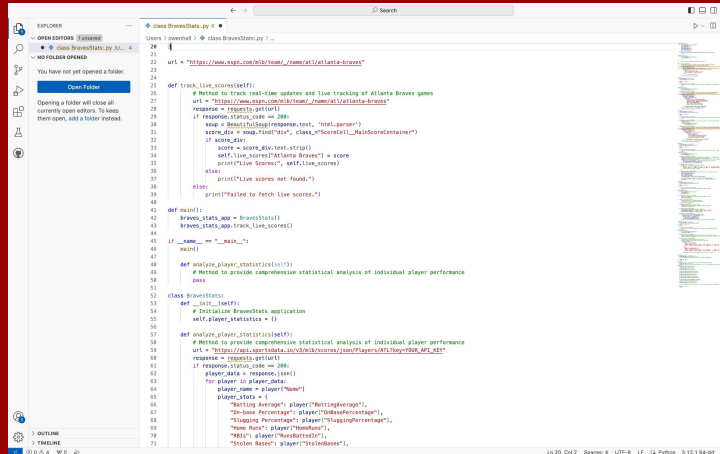


# Braves Stat Tracker by Owen Hall

Braves Stat Tracker is an application that is meant to track team and players statistics live. The goal was to create an application to track my personal favorite baseball team.



```
1 # BravesStatTracker.py
2
3 import requests
4 from bs4 import BeautifulSoup
5
6 # BravesStatTracker.py
7
8 # Method to track live team and player statistics
9 def track_live_scores(self):
10     # Method to track live team and player statistics
11     url = "https://www.espn.com/mlb/scoreboard/_/teams/atlanta-braves"
12     response = requests.get(url)
13     if response.status_code == 200:
14         soup = BeautifulSoup(response.text, "html.parser")
15         score_div = soup.find("div", class_="scoreboard__main-container")
16         score = score_div.text.strip()
17         self.live_scores["Atlanta Braves"] = score
18         print("Live Scores", self.live_scores)
19     else:
20         print("Live scores not found.")
21
22 # Method to provide comprehensive statistical analysis of individual player performance
23 def analyze_player_statistics(self):
24     # Method to provide comprehensive statistical analysis of individual player performance
25     url = "https://api.sportsdata.io/v3/mlb/scores/json/Players/ATL/braves_2024_2025"
26     response = requests.get(url)
27     if response.status_code == 200:
28         player_data = response.json()
29         for player in player_data:
30             player_name = player["Name"]
31             player_stats = {
32                 "Batting Average": player["BattingAverage"],
33                 "On-base Percentage": player["OnBasePercentage"],
34                 "Slugging Percentage": player["SluggingPercentage"],
35                 "Home Runs": player["HomeRuns"],
36                 "RBIs": player["RBIs"],
37                 "Stolen Bases": player["StolenBases"],
38             }
39
40 # Main method
41 def main():
42     braves_stat_tracker = BravesStatTracker()
43     braves_stat_tracker.track_live_scores()
44
45 if __name__ == "__main__":
46     main()
47
48 # Class BravesStatTracker
49 class BravesStatTracker:
50     def __init__(self):
51         # Initialize BravesStatTracker application
52         self.player_statistics = {}
53
54     def analyze_player_statistics(self):
55         # Method to provide comprehensive statistical analysis of individual player performance
56         url = "https://api.sportsdata.io/v3/mlb/scores/json/Players/ATL/braves_2024_2025"
57         response = requests.get(url)
58         if response.status_code == 200:
59             player_data = response.json()
60             for player in player_data:
61                 player_name = player["Name"]
62                 player_stats = {
63                     "Batting Average": player["BattingAverage"],
64                     "On-base Percentage": player["OnBasePercentage"],
65                     "Slugging Percentage": player["SluggingPercentage"],
66                     "Home Runs": player["HomeRuns"],
67                     "RBIs": player["RBIs"],
68                     "Stolen Bases": player["StolenBases"],
69                 }
```