I took the data in from 3 sources:

https://www.baseball-reference.com/(baseball reference)
https://www.kaggle.com/datasets/paulrjohnson/mlb-ballparks(kaggle)
https://github.com/toddrob99/MLB-StatsAPI/wiki(API)

I tried to find an SQL database for MLB data but was unable to do so.

For mets_game_scoring_plays, I collected the count of the number of scoring plays in each game in 2022 that the Mets were involved in from the API, ie how many scoring plays there were for either team in every Mets game. I also numbered the games from 1-162 and made a 2 column table from this data.

For mets_schedule, I took in data from baseball reference that had a ton of useless or impractical columns. From the original csv of the mets schedule, I dropped a column saying "boxscore" in every row that is impractical in a real database, Tm, which always said "NYM" so it's useless, and a lot of other columns that I didn't think were needed, like cLI(championship leverage index) or GB(games back from 1st place).I ended up with the columns that I explain in Table Summary.pdf

I also modified a column in this table to be away_game and have a simple "Yes" or "No" value instead of having to parse through an @ symbol which was intuitively confusing. I also renamed a few columns.

For mlb_stadiums, from kaggle, some of the 3 letter initials were off. It had ARZ instead of ARI, KC instead of KCR, SD instead of SDP, etc. I manually adjusted these since there weren't many of them, and I didn't need to drop any columns, although I did rename them.

For mlb_standings, from baseball reference, I cut out excess columns I didn't want to include before reading the CSV file into python.
https://www.baseball-reference.com/leagues/majors/2022-standings.shtml. If you click on this link and scroll down to detailed standings you will see a lot more columns that I did not include. Since I did this before reading the file in, I didn't need to adjust the data in Python besides renaming the columns.

For mets_highlights, from the API, I did a similar process to what I did for mets_game_scoring plays. I processed every highlight from Mets games, and then separated them into the highlight titles, descriptions, and links. I used ::: as a delimiter to separate these 3 elements. Then I used a for loop and a while loop to get the proper set of game numbers in the game_number columns, since many of these highlights come from the same game.

Then I realized I needed to connect mlb_standings to the rest of the database, so I added team_data to connect with mlb_standings via its team name column and then connected

team_data to mets_schedule and mlb_stadiums via its team abbreviation column. I used mlb_standings ‘team_name’ and mlb_stadiums ‘team_abbreviation’ as the data for this table, so I didn’t actually take in any new data from any new sources, just reorganized already existing data into a new table for convenience.