

Course:	INFO6143
Professor:	Jim Cooper
Project:	Project #2 – MLB Data Collection – Version 1.0
Due Date:	Monday, April 15, 2024
Submitting:	Please see the last page for instructions

How will my project be marked?

- This project counts for 15% of your final grade and will be evaluated using the following grid:

Marks Available	What are the Marks Awarded For?	Mark Assigned
2	Good coding style including proper indentation and use of variable and object naming conventions and suitable comments.	
1	Read and successfully process the URLs.txt file	
3	Successfully retrieve both JSON strings and convert each type to a valid Python dictionary.	
3	Extract the six required data fields as specified in the PDF description.	
3	Use the extracted data to create a nested list that can be used to build a corresponding CSV file.	
2	Iterate through the three test URLs provided and generate the correct output CSV file.	
1	Proper zipped submission.	
15	Total	

Project Description

This project has some overlap with the first project from the JavaScript course (INFO-6144) however in this case, we're creating an application that has virtually no user interface and handles all input and output using files.

Despite the lack of a UI, we'll still be able to retrieve JSON data from a distant web site and convert the JSON to a Python dictionary that allows the same kind of access that we had with JavaScript.

The main goal of this project will be to navigate the dictionary and extract a subset of the data to be saved to an output file. What makes this difficult is that the dictionary for this project, is substantially more complex than any other dictionary application we've seen in the course.

When you have your data, you'll generate your output file using CSV formatting as this will be the most useful. Since you acquired some knowledge of the major league baseball site working on the JavaScript project, you can leverage that knowledge to make this project slightly easier.

Specific Requirements:

Because we're doing a batch job, we can choose to put several dates in a file and read that file as input when our Python code starts up. This will allow us to handle multiple arbitrary dates in succession and end up creating a single output file that's a collection of the data we want for all the dates selected. While there are various ways to store the date information, in this case, we'll just create several identical URLs that have different date values:

<https://statsapi.mlb.com/api/v1/schedule/games/?sportId=1&startDate=2023-05-07&endDate=2023-05-07>
<https://statsapi.mlb.com/api/v1/schedule/games/?sportId=1&startDate=2023-06-07&endDate=2023-06-07>
<https://statsapi.mlb.com/api/v1/schedule/games/?sportId=1&startDate=2023-07-07&endDate=2023-07-07>

These URLs are stored in a file called `urls.txt` and located in the same folder as your Python code.

Once you're able to retrieve the JSON data from the MLB site and subsequently convert the JSON into a Python dictionary, you'll need to extract a subset of the data for output to the CSV file. This file should be named `"MLBData.csv"`

Regarding your CSV data, you should start with the following header information:

```
header = ["Date", "Home Team", "Away Team", "Game PK", "Headline", "MP4"]
```

See the following screen captures from the two JSON files (included) for help locating the properties needed to match with the header columns:

JSON #1:



```
1 {
2   "copyright": "Copyright 2023 MLB Advanced Media, L.P. Use of any content
3   "totalItems": 15,
4   "totalEvents": 0,
5   "totalGames": 15,
6   "totalGamesInProgress": 0,
7   "dates": [
8     {
9       "date": "2023-05-07",
10      "totalItems": 15,
11      "totalEvents": 0,
12      "totalGames": 15,
13      "totalGamesInProgress": 0,
14      "games": [
15        {
16          "gamePk": 718277,
17          "gameGuid": "0d6bc969-342a-423b-aecb-8b145764f524",
18          "link": "/api/v1.1/game/718277/feed/live",
19          "gameType": "R",
20          "season": "2023",
21          "gameDate": "2023-05-07T15:35:00Z",
```

```

25     "codedGameState": "F",
26     "detailedState": "Final",
27     "statusCode": "F",
28     "startTimeTBD": false,
29     "abstractGameCode": "F"
30 },
31 "teams": {
32     "away": {
33         "leagueRecord": {
34             "wins": 22,
35             "losses": 12,
36             "pct": ".647"
37         },
38         "score": 2,
39         "team": {
40             "id": 110,
41             "name": "Baltimore Orioles",
42             "link": "/api/v1/teams/110"
43         },
44         "isWinner": false,
45         "splitSquad": false,
46         "seriesNumber": 11
47     },
48     "home": {
49         "leagueRecord": {
50             "wins": 24,
51             "losses": 11,
52             "pct": ".686"
53         },
54         "score": 3,
55         "team": {
56             "id": 144,
57             "name": "Atlanta Braves",
58             "link": "/api/v1/teams/144"
59         },
60         "isWinner": true,
61         "splitSquad": false,
62         "seriesNumber": 11
63     }
64 },
65 "venue": {
66     "id": 4705,
67     "name": "Truist Park",
68     "link": "/api/v1/venues/4705"
69 },
70 "content": {
71     "link": "/api/v1/game/718277/content"
72 },
73 "isTie": false,
74 "gameNumber": 1,
75 "publicFacing": true

```

JSON #2:

```
1643     "id": "orioles-vs-braves-highlights-x7656"
1644   },
1645   "freeGame": false,
1646   "enhancedGame": false
1647 },
1648 "highlights": {
1649   "scoreboard": null,
1650   "gameCenter": null,
1651   "milestone": null,
1652   "highlights": {
1653     "items": [
1654       {
1655         "type": "video",
1656         "state": "A",
1657         "date": "2023-05-07T20:12:12.23Z",
1658         "id": "orioles-vs-braves-highlights-x7656",
1659         "headline": "Orioles vs. Braves Highlights",
1660         "seoTitle": "",
1661         "slug": "orioles-vs-braves-highlights-x7656",
1662         "blurb": "Harris II's walk-off double lifts Braves in the",
1663         "keywordsAll": [
1664           {
1665             "type": "game",
1666             "value": "gamepk-718277",
1667             "displayName": "2023/05/07 bal@atl"
1668           },
1669           {
1670             "type": "game_pk",
1671             "value": "718277",
1672             "displayName": "2023/05/07 bal@atl"
1673           },
1674           {
1675             "type": "team",
1676             "value": "teamid-110",
1677             "displayName": "Baltimore Orioles"
1678           }
1679         ]
1680       }
1681     ]
1682   }
1683 }
```

```

1946         "width": 1920,
1947         "height": 810,
1948         "src": "https://img.mlbstatic.com/mlb-images/i
1949         "at2x": "https://img.mlbstatic.com/mlb-images/
1950         "at3x": "https://img.mlbstatic.com/mlb-images/
1951     }
1952 }
1953 ],
1954 "noIndex": false,
1955 "mediaPlaybackId": "f9c808d4-6a371217-a34af655-csvm-diamon
1956 "title": "Orioles vs. Braves Highlights",
1957 "description": "Michael Harris II hit a walk-off RBI doubl
1958 "duration": "00:03:49",
1959 "mediaPlaybackUrl": "",
1960 "playbacks": [
1961     {
1962         "name": "mp4Avc",
1963         "url": "https://mlb-cuts-diamond.mlb.com/FORGE/202
1964         "width": "",
1965         "height": ""
1966     },
1967     {
1968         "name": "hlsCloud",
1969         "url": "https://mlb-cuts-diamond.mlb.com/FORGE/202
1970         "width": "",
1971         "height": ""
1972     },
1973     {
1974         "name": "HTTP_CLOUD_WIRED",
1975         "url": "https://mlb-cuts-diamond.mlb.com/FORGE/202
1976         "width": "",
1977         "height": ""
1978     },
1979     {
1980         "name": "HTTP_CLOUD_WIRED_60",
1981         "url": "https://mlb-cuts-diamond.mlb.com/FORGE/202
1982         "width": "",

```

As often happens in the real world, projects are initiated without a detailed step by step process for reaching completion. In this case, we're providing you with a means to acquire the data and a sample of the output you need to produce it but some of the strategies for achieving the correct result are being left for you to discover.

*** End of Requirements ***

Notes:

Here's some sample Python code for retrieving MLB JSON data for the current day:

```

import json
import requests

response = requests.get("https://statsapi.mlb.com/api/v1/schedule/games/?sportId=1")
mlb_json = json.loads(response.text)
json_str = json.dumps(mlb_json)
pyObject = json.loads(json_str)

print(json_str)

```

Sample Output:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Date	Home Team	Away Team	Game PK	Headline	MP4							
2	07-05-23	Atlanta Braves	Baltimore Orioles	718277	Orioles vs. Braves Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/f9c808d4-6a371217-a34af65							
3	07-05-23	Pittsburgh Pirates	Toronto Blue Jays	718278	Blue Jays vs. Pirates Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/4d442fd8-10933dba-d34a35f							
4	07-05-23	Philadelphia Phillies	Boston Red Sox	718270	Red Sox vs. Phillies Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/eea5e074-163508e1-ef9c381							
5	07-05-23	New York Mets	Colorado Rockies	718276	Rockies vs. Mets Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/8e046e42-a03281b7-734e13f							
6	07-05-23	Cleveland Guardians	Minnesota Twins	718273	Twins vs. Guardians Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/4f8e654b-c7b12771-634e7dc							
7	07-05-23	Tampa Bay Rays	New York Yankees	718271	Yankees vs. Rays Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/e3259150-f3e9015a-87306f5							
8	07-05-23	Kansas City Royals	Oakland Athletics	718268	Athletics vs. Royals Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/93254967-2d29c894-4e6d1a							
9	07-05-23	St. Louis Cardinals	Detroit Tigers	718272	Tigers vs. Cardinals Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/c7630797-7a103856-7195be1							
10	07-05-23	Chicago Cubs	Miami Marlins	718269	Marlins vs. Cubs Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/19a76967-2e83ec51-28ca97c							
11	07-05-23	San Francisco Giants	Milwaukee Brewers	718267	Brewers vs. Giants Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/e7d34e5c-568eca2a-9d64a74							
12	07-05-23	Los Angeles Angels	Texas Rangers	718264	Rangers vs. Angels Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/bc8a863b-33b76923-c79ec82							
13	07-05-23	Arizona Diamondbacks	Washington Nationals	718263	Nationals vs D-backs Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/fb8461bd-bb83bac7-e4cf3dfc							
14	07-05-23	Seattle Mariners	Houston Astros	718265	Astros vs. Mariners Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/62c6f4e3-2393df5b-9ed55ab							
15	07-05-23	Cincinnati Reds	Chicago White Sox	718262	White Sox vs. Reds Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/aa0d0258-ff6e4dee-e3bc3ea							
16	07-05-23	San Diego Padres	Los Angeles Dodgers	718266	Dodgers vs. Padres Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-05/07/a4e14589-6dfa0b5e-89f9601							
17	07-06-23	Pittsburgh Pirates	Oakland Athletics	717865	Athletics vs. Pirates Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/8c1362fe-34bd7c1e-164b3fa							
18	07-06-23	San Diego Padres	Seattle Mariners	717863	Mariners vs. Padres Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/a9f35acd-8f83ec5c-cf11f44b							
19	07-06-23	Miami Marlins	Kansas City Royals	717860	Royals vs. Marlins Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/ead0e52b-519e5c89-1f9dce6							
20	07-06-23	Tampa Bay Rays	Minnesota Twins	717859	Twins vs. Rays Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/8629760d-43f72fc6-27d0fbb							
21	07-06-23	Washington Nationals	Arizona Diamondbacks	717866	D-backs vs. Nationals Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/88c65435-f3253935-f8454b5							
22	07-06-23	Toronto Blue Jays	Houston Astros	717856	Astros vs. Blue Jays Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/f13def58-8cbf5b51-74b2c6fa							
23	07-06-23	Cleveland Guardians	Boston Red Sox	717858	Red Sox vs. Guardians Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/566e67b6-273984b0-c5568d							
24	07-06-23	Cincinnati Reds	Los Angeles Dodgers	717853	Dodgers vs. Reds Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/7def3108-31185c2f-4434540							
25	07-06-23	Atlanta Braves	New York Mets	717850	Mets vs. Braves Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/5abd4316-08e8de14-e0eb84							
26	07-06-23	Milwaukee Brewers	Baltimore Orioles	717851	Orioles vs. Brewers Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/3cab9aa8-711a9394-bcfe3ea							
27	07-06-23	Texas Rangers	St. Louis Cardinals	717854	Cardinals vs. Rangers Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/cb5166dd-cf85eb6a-5b961e8							
28	07-06-23	Colorado Rockies	San Francisco Giants	717852	Giants vs. Rockies Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/5aca4c82-9feab7fa-f28d3cda							
29	07-06-23	Los Angeles Angels	Chicago Cubs	717855	Cubs vs. Angels Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/07/80adfe10-efef036ac-1160df33							
30	07-06-23	Philadelphia Phillies	Detroit Tigers	717862	Tigers vs. Phillies Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/08/320b6058-c1bea012-08e443c							
31	07-06-23	New York Yankees	Chicago White Sox	717857	White Sox vs. Yankees Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-06/08/9f623c72-f30f7a44-390f2f5c							
32	07-07-23	Detroit Tigers	Toronto Blue Jays	717457	Blue Jays vs. Tigers Highlights	https://mlb-cuts-diamond.mlb.com/FORGE/2023/2023-07/07/ad1c887a-5b77caa4-b28b67e							

How should I submit my project?

Electronic Submission:

Submit your program files to the *Info6143 “Project 2”* electronic submission folder in *FanshaweOnline*. These files should be submitted as a single “zip” file containing your web application’s complete website.

I strongly recommend that you test your own submission to ensure that nothing has been missed.

Submit your project on time!

Project submissions must be made on time! Late projects will be subject to divisional policy on missed tests and late projects. In accordance with this policy, no late projects will be accepted without prior notification being received by the instructor from the student.

Submit your own work!

It is considered cheating to submit work done by another student or from another source. Helping another student cheat by sharing your work with them is also not tolerated. Students are encouraged to share ideas and to work together on practice exercises, but any code or documentation prepared for a project must be done by the individual student. Penalties for cheating or helping another student cheat may include being assigned zero on the project with even more severe penalties if you are caught cheating more than once. Just submit your own work and benefit from having made the effort on your own.