



# EVENTS IN THE US

*ticketmaster*®

Sebastian Alonso, Aimee Galindo, Mark Helotie,  
Monila Khadka, Marshal Rockafellow





# Introduction

We have created an interactive map that allows users to explore a list of entertainment events ( based on selected cities/markets in the United States. )





## Features

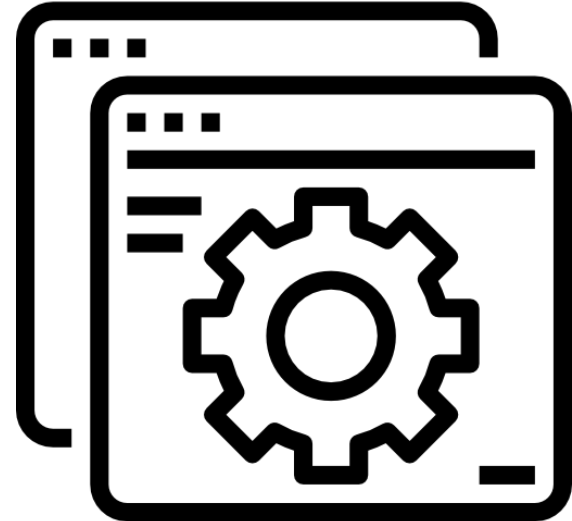
City groups in an Array for selected markets placed as markers on the map.

Hover over a marker to see the city/market name.

Click the market to zoom in and view the events in that area.

Click on the event to see the data fetched on that event.

Events fetched live via the TicketMaster API by market id



## Built With



Leaflet.js - An open-source JavaScript library for interactive maps.



TicketMaster API - To fetch event data based on city/market selection.



OpenStreetMap - Open-source map data used as a base layer for the map.



Javascript - A scripting language that is used to create dynamic and interactive content, control multimedia, animate elements, and perform a wide range of tasks on web pages



HTML - The markup language that is used to structure, layout, and give meaning to the webpage.



CSS - A stylesheet language that defines how HTML or XML documents should be visually presented on different media.



Jupyterlab- Web-based interactive development environment.

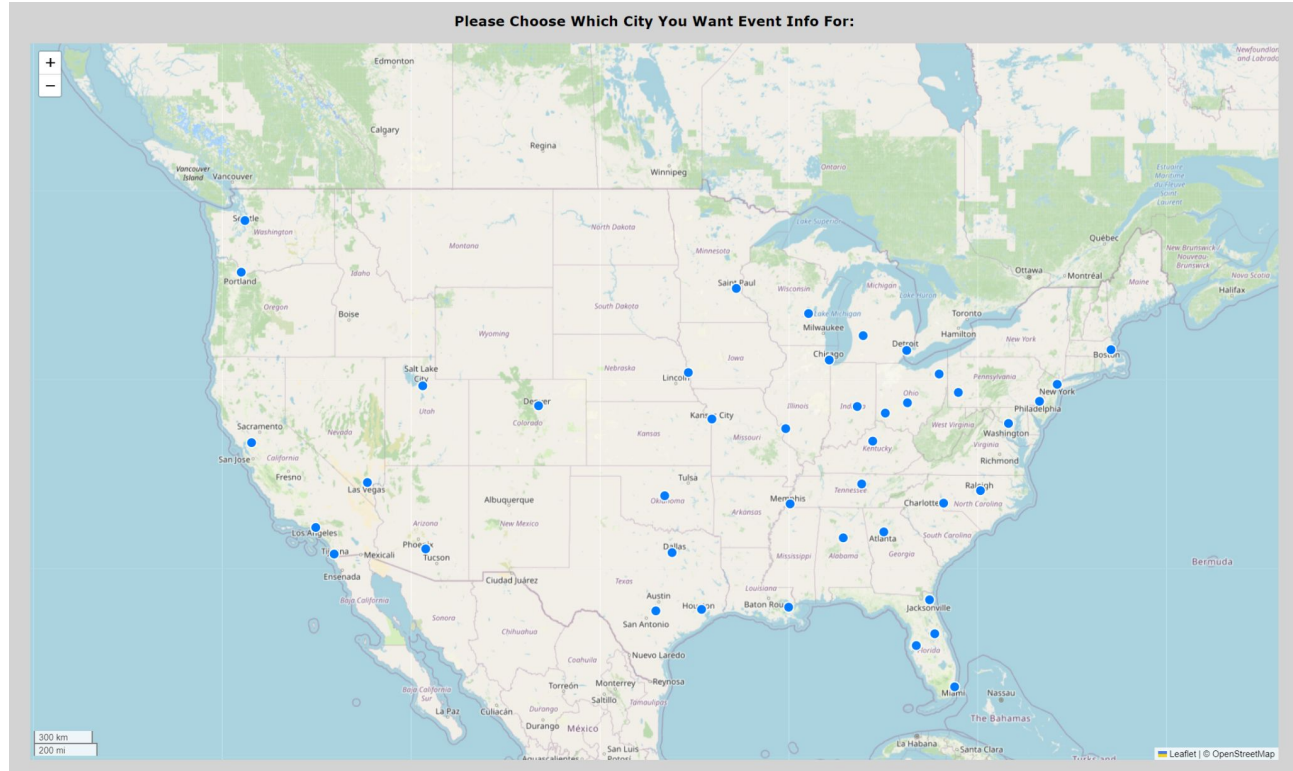


# Demo

[Click here to view!](#)

# Markers

Gathering of TM metro areas and  
putting together the lat/long  
array to create the initial map



# HTML & CSS

```
<!DOCTYPE html>
<html lang="en-us">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ticketmaster City Markers</title>

  <!-- Leaflet CSS -->
  <link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.4/dist/leaflet.css"
    integrity="sha256-p4NxAoJBhIIN+hmNHrzRCf9tD/miZyoHS0obTRR9BMY="
    crossorigin="" />

  <!-- Leaflet JavaScript code -->
  <script src="https://unpkg.com/leaflet@1.9.4/dist/leaflet.js"
    integrity="sha256-20nQcchB9coBqijjZRGuk2/Z9VM+kNiyxNV1lvTLzBo="
    crossorigin=""></script>

  <!-- Our CSS -->
  <link rel="stylesheet" type="text/css" href="static/css/style.css">
</head>
<body>
  <p id="location"><center>Please Click on one of the Markers below</center></p>
  <div id="map"></div>
  <div id="events"></div>

  <!-- Our JavaScript -->
  <script type="text/javascript" src="static/js/logic.js"></script>
</body>
</html>
```

```
body {
  padding: 0;
  margin: 0;
  background-color: #lightgray;
  font-family: Verdana, Arial, sans-serif;
  margin-left: 2%;
  margin-right: 2%;
  margin-bottom: 2%;
}

#map,
body,
html {
  height: 97%;
}

.custom-div-icon {
  background-color: #007bff; /* Marker backgr
  border: 2px solid #fff; /* Marker border co
  color: #fff; /* Marker text color */
  font-size: 14px;
  border-radius: 50%;
  width: 30px; /* Marker width */
  height: 30px; /* Marker height */
  line-height: 30px;
  text-align: center;
}

table, th, td {
  border: 1px solid #lightgray;
  border-collapse: collapse;
}
```

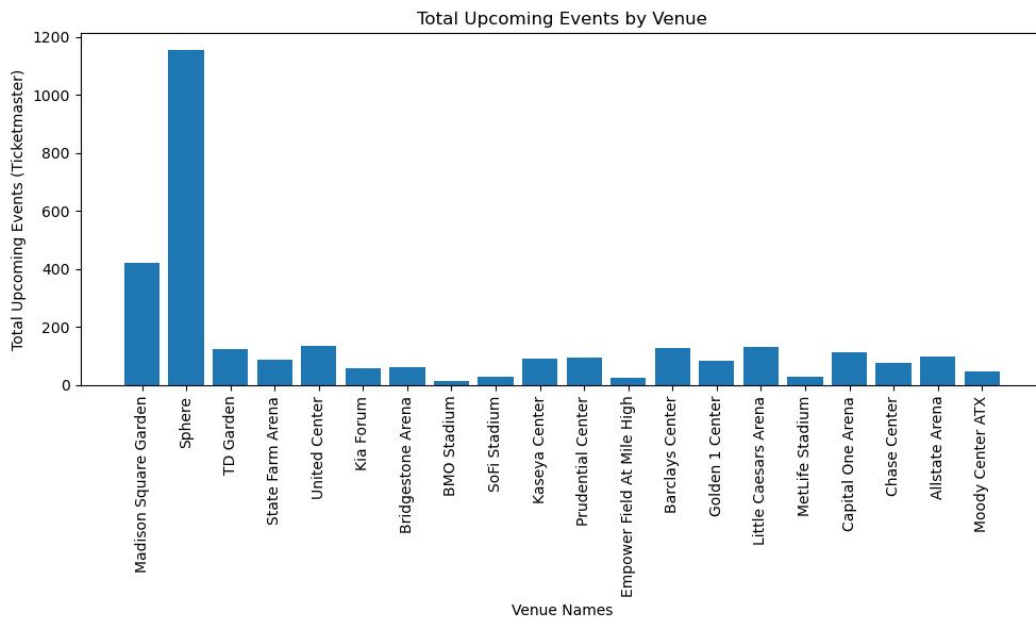
## HTML

- Leaflet CSS
- Leaflet JavaScript code

## style.css

- Marker

# Jupyterlab Analysis



```
#Venue
response_venue=requests.get('https://app.ticketmaster.com/discovery/v2/venues?apikey=KitAybi')
venue_data = response_venue.json()

venue = venue_data.get("_embedded", {}).get("venues", []) # Extract the events list
venue_list=[]
upcoming=[]

for venues in venue:
    venue_name = venues.get("name", "") # Extract 'venue name' list for each event
    venue_count=venues.get("upcomingEvents",{}).get("ticketmaster",0)
    venue_list.append(venue_name)
    upcoming.append(venue_count)

# Create a dictionary to associate venue names with their respective total upcoming events
venue_count = dict(zip(venue_list, upcoming))
print(venue_count)

# Extract venue names and counts
venue_names = list(venue_count.keys())
venue_counts = list(venue_count.values())

# Create a bar graph
plt.figure(figsize=(10, 6)) # Adjust the figure size as needed
plt.bar(venue_names, venue_counts)

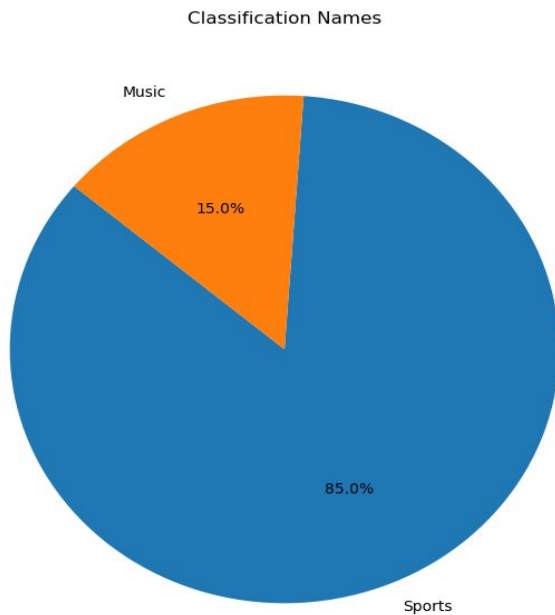
# Adding labels and title
plt.xlabel('Venue Names')
plt.ylabel('Total Upcoming Events (Ticketmaster)')
plt.title('Total Upcoming Events by Venue')

# Rotating x-axis labels for better readability
plt.xticks(rotation=90)

# Show the graph
plt.tight_layout()
plt.show()
```



# Today's (10/5/2023) Events



- The other categories that Ticketmaster identifies are: Film, Arts & Theater, Music, undefined & Miscellaneous.

```
response_event=requests.get('https://app.ticketmaster.com/discovery/v2/events?apikey=KiTAybgY7A9y1T65k2RupJBbJq
response_json = response_event.json()

# Event class for today
# Extract 'segment' names from the event data
events = response_json.get("_embedded", {}).get("events", []) # Extract the events list
segments = []

for event in events:
    classifications = event.get("classifications", []) # Extract 'classifications' list for each event
    #print(classifications)
    for classification in classifications:
        segment_name = classification.get("segment", {}).get("name")
        segments.append(segment_name)

# Count the occurrences of each 'segment' name
name_counts = Counter(segments)

# Creating a pie chart
plt.figure(figsize=(8, 8))
plt.pie(name_counts.values(), labels=name_counts.keys(), autopct='%1.1f%%', startangle=140)
plt.title('Classification Names')

# Displaying the pie chart
plt.show()
```

# Javascript

```
1 // Cities array data created
2 city_data = [
3   {
4     'id': 1,
5     'name': 'Birmingham',
6     'lat': 33.53894154,
7     'lng': -86.81664397
8   },
9   {
10    'id': 2,
11    'name': 'Charlotte',
12    'lat': 35.23466349,
13    'lng': -80.84030215
14  },
15  {
16    'id': 3,
17    'name': 'Chicago',
18    'lat': 41.90491852,
19    'lng': -87.66652212
20  },
21  {
22    'id': 4,
23    'name': 'Cincinnati & Dayton',
24    'lat': 39.48953605,
25    'lng': -84.32655339
26  },
27  {
28    'id': 5,
29    'name': 'Dallas - Fort Worth',
```

Array

```
1 // Create a map object.
2 let myMap = L.map("map", {
3   center: [40, -98],
4   zoom: 5,
5   fullscreenControl: true
6 });
7
8 // Add a tile layer.
9 L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
10   attribution: '&copy; <a href="https://www.openstreetmap.org/copyright">OpenStreetMap</a>'
11 }).addTo(myMap);
12
13 // Add the scale to the map
14 L.control.scale().addTo(myMap);
15
16 // Create a layer group for city markers.
17 const cityMarkerGroup = L.layerGroup().addTo(myMap);
18
19 // Function to fetch events in a city using Ticketmaster API
20 ✓ function getEventsInCity(id) {
21   const apiKey = 'S2v9Md44UbLI7UVMA583AjbIZ4dPB5tu';
22
23   // Define the Ticketmaster API URL to fetch events in the city
24   const apiUrl = `https://app.ticketmaster.com/discovery/v2/events.json?marketId=${id}&apiKey=${apiKey}&size=10`;
25
26   fetch(apiUrl)
```

Logic

# Ticketmaster API

Make live API calls without writing a single line of code! The API Explorer currently works with the [Discovery API](#)

Discovery API v2

S2v9Md44UblI7UVMA583AjbIz4dPB5tu

Methods

☐ All ☒ Get

Event Search

Event Search

Find events and filter your sea location, date, availability, and more.

Parameters

id

keyword

attractionId

marketId

Filter by market id

venueId

postalCode

latlong

unit

source

radius

none

none

locale

marketId

startDateTime

includeTBA

includeTBD

endDateTime

none






none

includeTest

size

page

Page		
size: 20	totalElements: 619	totalPages: 31
number: 0		
Events 20		
Eagles - The Long Goodbye	2023-10-05 Ball Arena	>
Denver Broncos vs. New York Jets	2023-10-08 Empower Field ...	>
PINK: TRUSTFALL TOUR	2023-10-25 Ball Arena	>
Denver Broncos vs. Green Bay Packers	2023-10-22 Empower Field ...	>
Eagles - The Long Goodbye	2023-10-06 Ball Arena	>
Denver Broncos vs. Kansas City Chiefs	2023-10-29 Empower Field ...	>
Denver Broncos vs. Minnesota Vikings	2023-11-19 Empower Field ...	>
Denver Broncos v New York Jets -...	2023-10-08 Empower Field ...	>
Denver Broncos vs. Cleveland Browns	2023-11-26 Empower Field ...	>
Denver Broncos V Kansas City Chiefs ...	2023-10-29 Empower Field ...	>

Object		
name: Denver Broncos vs. New York Jets	type: event	
id: G5vzZ9tK3mVw3	test: false	
url: https://www.ticketmaster.com/denver-broncos-vs-new-york-jets-denver-colorado-10-08-2023/event/1E005E94D2BB4C92	locale: en-us	
Images		
	Ratio: 4_3 (225 x 305)	>
	Ratio: 3_2 (427 x 640)	>
	Ratio: 3_2 (203 x 305)	>
	Ratio: 16_9 (1152 x 2048)	>
	Ratio: 16_9 (115 x 205)	>



## Challenges

- Postgres database hosted on Elephant SQL / flask
- Clear events layer per city click

## Next Steps

- Selection of events within the same venue
- Filters to better fit your own preferences



# Thank you



Happy to answer any questions!