Events.html
- keyword ← text box
- distance → 10, placeholder, miles ← text box
- category → default, ← drop down
- location ← text box
- check box → ipinfo.io API

default category ← all types provided in API

- Background image
- If auto location is selected then the location text box should disappear. when not selected location is a required field with * → input is string.

* **Distance**: current loc is center and should find from the radius of the curr loc just show that o/p. default is 10miles.

* **Buttons**

① Search : take valid input
JS to request python script with form i/ps.
GET to transfer data to web server
python using flask → extract inputs → invoke the API

If invalid input :
Show error called "inoHtip". If no values where default is avail. then should take default values.

Do not call ticketmaster API from JS.
Use XMLHTTP (or file fetch()) to call ipinfo. & google maps decoding directly from JS.

② Clear : clear the result area and the form fields. set it to default
— Do using JS

---

* Displaying event results.

→ client JS should use i/p address to get the geocoding via the google maps decoding API
Google Maps geocoding
→ address : format by national post service, avoid additional elements
→ key : applications API key
↳ will return lat & long

Ticketmaster uses geohash so convert lat/long to geohash on client or server side.

A Python library for geohash encoding is available online. The source code can be found here:

https://pypi.org/project/geolib/

Call the function geihash.encode() as in:

```
geohash = geohash.encode("10.2995", "-21.9993", 7)
```

A JavaScript library for the same lat/long → geohash conversion is available here:

https://www.movable-type.co.uk/scripts/geohash.html

and here:

https://github.com/chrisveness/latlon-geohash/blob/master/latlon-geohash.js

Call the function Geohash.encode() as in:

```
const geohash = Geohash.encode(52.205, 0.119, 7);
```

* Ticketmaster API (returns JSON)

Parameter : 1. apikey
2. geopoint
3. radius
4. segmentid
5. unit -(miles)
6. keyword

Use JS to parse the json object, extract needed fields, and display the results in tabular format

Needed fields : 1. Date, Icon, event name, genre, venue name.

If api returns an empty set, then display "No records found".

HTML Table Column	API service response
Date	The value of the "localDate" and "localTime" attributes that is part of "events" object
Icon	The value of the "images" attribute that is part of the "events" object.
Event	The value of the "name" attribute that is part of the "events" object.
Genre	The value of the "segment" attributes that is part of the "events" object.
Venue	The value of the "name" attribute that is part of the "venue" object inside "events" object.

---

* Displaying table of events

table view containing all events for a given distance filtered by category and keyword
→ sort the events based on event, genre or venue by clicking on the table header
fetch the results of 20 events (or less)

→ event details
→ on name click, a card should display detailed results of the event below the table of listed events and the page should automatically scroll all the way down to the cards loc.
(use JS)

Should display the detailed information using the event details API and venue search API
Two parameters :
→ ID : event id
→ apikey :

→ clicking on this should open a new tab with artist's upcoming events.

If the json stream does not contain a certain field then it will not appear in the details page.

* Ticket status is colored
→ on sale : green
→ off sale : red
→ canceled : black
→ postponed : orange
→ rescheduled : orange.

a venue details card should be displayed the text "Show venue details" and the arrow should disappear.
→ page should automatically scroll down
→ to retrive details: request venue search api
↳ keyword, api key

open in a new page.

If JSON file doesnot contain a certain field the value should be set to N/A