**SeatGeek API Documentation**

**API Endpoint**

<https://api.seatgeek.com/2>

**Resource Endpoints**

/events

/events/{EVENT\_ID}

/performers

/performers/{PERFORMER\_ID}

/venues

/venues/{VENUE\_ID}

**Authentication**

SeatGeek provides two options to authenticate yourself during calls.

* Client\_id
* Client\_secret

We are using only client\_id in our project. To generate query string, we must add client\_id to the end of any valid URL to authenticate our request.

**Responses**

The API provides two basic types of response documents:

* Single resource response document and
* Bulk resource response document.

The former returns a single document of the requested resource. This type of document is only returned when attempting to access a document with a: https://api.seatgeek.com/2/events/{EVENT\_ID} style request.

A bulk resource response document will include an array of single resource documents along with some additional meta information about the response. If no resources are found, meta information will be returned with an empty resources array.

**Response Document Formats**

Default is JSON. Acceptable argument is JSON and XML. We are not using XML, so we don’t have to provide any argument in query string.

**Meta**

Each bulk response is appended with meta information

**took:**

Response time in milliseconds

**geolocation:**

Details about the results of geolocation. null if geolocation not requested, or unsuccessful.

**per\_page:**

Number of documents per page

**total:**

Total number of matched documents

**page:**

Current paginated page number

**Resource Endpoints Description**

1. **Events**

To find an event matching specific search criteria you must send a request to the **/events** endpoint. Sending a request to **/events** will return a paginated list of all events.

**Example Requests:**

[**https://api.seatgeek.com/2/events/801255**](https://api.seatgeek.com/2/events/801255)

* 1. **performers Argument**

The performers argument is used to scope the result set to specific performers. The performers argument may be used several times in the same query.

The simplest query to the performers argument looks like below examples. This will return all events where the New York Mets are participating.

**Example Requests:**

[**https://api.seatgeek.com/2/events?performers.slug=new-york-mets**](https://api.seatgeek.com/2/events?performers.slug=new-york-mets)

[**https://api.seatgeek.com/2/events?performers.slug=new-york-**](https://api.seatgeek.com/2/events?performers.slug=new-york-)**mets&performers=new-york-yankees**

* 1. **venue Argument**

The venue argument is used to scope the result set to a specific venue or to venues with specific properties. Any of the fields for searching for venues may be used with the venue argument (e.g. venue.city, venue.id, or venue.state). To make an OR query, a list of venue ids separated by commas can be passed to venue.id

**Example Requests:**

**Events in NY state**

**https://api.seatgeek.com/2/events?venue.state=NY**

**Events at CitiField or Yankee Stadium**

[**https://api.seatgeek.com/2/events?venue.id=3,8**](https://api.seatgeek.com/2/events?venue.id=3,8)

* 1. **query Argument**

The q argument is used to search across all possible parts of a response document.

q: optional, default is null

This is a very broad search, and will possibly return unexpected results. For example, a query for 'madison', will return all events at any venue in Madison, WI, venues with Madison in their name (Madison Square Garden), and any performer with 'Madison' in their title.

The q argument also allows for some natural language queries such as yankees march, or skrillex on june 4th

**Example Requests**

**Matches 'boston celtics'**

**https://api.seatgeek.com/2/events?q=boston+celtics**

**Matches 'Coachella'**

**https://api.seatgeek.com/2/events?q=coachella**

* 1. **taxonomies Argument**

The taxonomies argument is used to scope the result set to events matching a specific type. You can find the list of currently available taxonomies at the taxonomies endpoint. You can search across multiple taxonomies by adding multiple taxonomies arguments to your query

**Example Requests**

**Sporting Events**

**https://api.seatgeek.com/2/events?taxonomies.name=sports**

**Monster Truck Events**

**https://api.seatgeek.com/2/events?taxonomies.name=monster\_truck**

**Sports or Concerts**

**https://api.seatgeek.com/2/events?taxonomies.name=sports&taxonomies.name=concert**

**Monster Truck Events (by taxonomy id)**

**https://api.seatgeek.com/2/events?taxonomies.id=1060400**

**==============================================================================**

1. **Performers**

To find a performer matching your search criteria you will send a request to the **/performers** endpoint. Sending a request to **/performers** will return a paginated list of all performers.

**Example Requests:**

[**https://api.seatgeek.com/2/performers/801255**](https://api.seatgeek.com/2/events/801255)

* 1. **slug Argument**

The slug argument is used to scope the result to a performer matching a slug.

**Example Requests**

**New York Mets**

**https://api.seatgeek.com/2/performers?slug=new-york-mets**

* 1. **query Argument**

The q argument is used to search across all possible parts of a response document.

**Example Requests**

**Matches 'red hot'**

**https://api.seatgeek.com/2/performers?q=red+hot**

**Matches 'skrillex'**

**https://api.seatgeek.com/2/performers?q=skrillex**

* 1. **taxonomies Argument**

The taxonomies argument is used to scope the result set to performers matching a specific type. You can find the list of currently available taxonomies at the taxonomies endpoint.

**Example Requests**

**Sporting Events**

**https://api.seatgeek.com/2/performers?taxonomies.name=sports**

**Monster Truck Events**

**https://api.seatgeek.com/2/performers?taxonomies.name=monster\_truck**

**Monster Truck Events (by taxonomy id)**

**https://api.seatgeek.com/2/performers?taxonomies.id=1060400**

* 1. **genres Argument**

The genres argument is used to scope the result set to performers matching a specific genre. You can find the list of currently available genres at the /genres endpoint.

**Example Requests**

**Rock Bands**

**https://api.seatgeek.com/2/performers?genres.slug=rock**

1. **Venues**

To find a venue matching your search criteria you will send a request to the /venues endpoint. Sending a request to /venues will return a paginated list of all performers.

* 1. **city Argument**
     1. https://api.seatgeek.com/2/venues?city=rockford
  2. **state Argument**
     1. https://api.seatgeek.com/2/venues?state=il
  3. **country Argument**
     1. https://api.seatgeek.com/2/venues?country=US
  4. **postal\_code Argument**
     1. https://api.seatgeek.com/2/venues?postal\_code=90210
  5. **query Argument**
     1. https://api.seatgeek.com/2/venues?q=madison

**API Features**

The SeatGeek API supports sorting, pagination, and geolocation. Geolocation is only available for resources with a location attribute (in other words, not /performers).

**Geolocation**

The API can geolocate requests based on a user's IP address, postal code (currently, only US and Canadian postal codes are supported), or latitude and longitude. In order to geolocate a request, either geoip or both lat and lon must be provided. Details about geolocation will be returned in the meta document.

**GET https://api.seatgeek.com/2/events?geoip=true**

**Pagination**

The SeatGeek Platform API has built in pagination support.

**GET https://api.seatgeek.com/2/venues?per\_page=25&page=3**

**Sorting**

The SeatGeek Platform API has basic support for sorting the response result set.

GET https://api.seatgeek.com/2/events?sort=score.desc

**Filtering**

The SeatGeek Platform API has basic support for filtering the response result set.

**GET https://api.seatgeek.com/2/events?listing\_count.gt=0**

**In Short,**

**API Endpoint:** [**https://api.seatgeek.com/2**](https://api.seatgeek.com/2)

**Resource Endpoints:**

* + - **/events**
    - **/performers**
    - **/venues**

**Events:** [**https://api.seatgeek.com/2**](https://api.seatgeek.com/2)**/events?...**

performers Argument

id

slug

venue Argument

query Argument

id Argument

taxonomies Argument

taxonomies.name

taxonomies.id

**Performers:** [**https://api.seatgeek.com/2**](https://api.seatgeek.com/2)**/performers?...**

slug Argument

query Argument

id Argument

taxonomies Argument

taxonomies.name

taxonomies.id

genres Argument

**Venues:** [**https://api.seatgeek.com/2**](https://api.seatgeek.com/2)**/venues?...**

city Argument

state Argument

country Argument

postal\_code Argument

query Argument

id Argument