

## Assignment 4 – Due 0

### 1. API-Driven Prototype – Comédie-Française Data

Working with your group from class, design and construct a basic prototype of an API-driven data presentation centered around a specific research question. This prototype should consist of 3 main components:

- a. A webpage using D3 or a similar library to generate a graphical representation from data retrieved through the API (this can be a very basic rendering, perhaps covering a small subset of the data your whole application would eventually present).
- b. Design elements, both incorporated into your visualization and also demonstrating the larger look and feel of your application through sketches and mockups.
- c. Written documentation which connects your group's use of the API to your research question and your group's design to a specific audience. This can also be a space to address remaining obstacles and to explain your group's overall design. (Please post this under Project Updates section of the course site).

See the more detailed instructions for getting started with API on the next page.

### 2. Read “The Potential of Spatial Humanities”

In his article, David Bodenhamer argues for a much broader definition of space beyond the mere geographical. Especially the “spatial turn” redefined space as a social and cultural concept. Now, data-driven GIS technology expands this notion, again redefining previous conceptions of space.

Read David Bodenhamer's text and use the text in Annotation Studio to comment on the value of GIS for the humanities (five annotations). Create meaningful tags that not only represent your comment but might also offer a better understanding of the text. Build on other student's tags or create your own.

Getting started with the CFRP API:

1. Install Postman at <https://www.getpostman.com> (requires Chrome).
2. In Postman, choose “Import”, select “Download from link”, paste in the URL “<https://www.getpostman.com/collections/cbae17cc22612d7477d6>”, and press “Import”.
3. View each of the four sample “Get” queries listed under the “api.cfrregisters.org” collection. Try making changes to the Headers and URL parameters, and hit “Send” to see how your results change.
4. View the different kinds of queries available by using the “Options” query. Try building your own query from one of the other endpoints listed there.
5. **Exercise** (optional): track attendance in the Parterre across the first 200 comedy performances after January 1, 1775.
  - a. Open the sample query titled “Performances with more than 100 parterre seats sold”
  - b. Look at the URL itself. Notice that it starts with “[https://cfrp-api.herokuapp.com/play\\_ticket\\_sales?](https://cfrp-api.herokuapp.com/play_ticket_sales?)”. That string, “play\_ticket\_sales”, is the **endpoint** the query is using (the part before that is the **root**). The parts after it are the **URL Parameters**.
  - c. Hit “Send” to see the results for this query. The results are displayed in **JSON** format. Each **item** appears as a series of lines nested in curly braces. Each of these lines contains a **key** (first, in bold) and a **value**. You’ll see the same set of keys in each item. These are the **column** that the data would have if viewed as a table. They are also the parameters you can include in the URL to narrow down your query.
  - d. Click “URL Params” to the right of the URL field, which will lay out the URL parameters below. You’ll see the first one has the key “total\_sold” and the value “gte.100” (*greater than or equal to 100*). For this exercise, we don’t care about the total number sold, so remove this parameter by clicking the X that appears to the right when you mouse over its row.
  - e. We only want comedy performances in our data, so we’ll add a new parameter. In the data, you’ll see one of the columns is “genre”, so you know you can use this as a parameter key. Click the grayed-out “URL Parameter Key” and type in “genre”. In the corresponding “Value” field, type in “eq.comédie”. This is how you tell the API to only return items whose genre value is *equal to* comédie. You’ll see your new parameter tacked on to the URL above.
  - f. Add another parameter to restrict your results to performances after January 1, 1775. You’ll see from the data that “date” is the key to use. For the value, enter “gte.1775-01-01” (we can tell this is the date format to use by looking at the data values).
  - g. Finally, we just want the first 200 items matching this query. Instead of a URL parameter, this is set by a **header** (a part of the request that gets sent to the API before the URL itself). Click on “Headers” to the right of

the URL to show them. You'll one header has the key "Range" and the value "1-100". Simply change this value to "1-200".

- h. Now, send your request by hitting "Send" again. You should see your results in JSON repopulate the lower part of the window, where you can copy or download them.

MIT OpenCourseWare  
<http://ocw.mit.edu>

CMS.633 / CMS.833 Digital Humanities  
Spring 2015

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.