Explanations (both technical and non) of what I’m doing for Char’s project using the Google Maps API:

Non-technical:

Google Maps is widely considered to be the standard for route-finding and navigation. Part of this project requires calculating, in miles, the *driving* distance between an origin and one or more destinations, which requires (a) identifying the fastest route between those points and (b) calculating that route’s distance. To do this by hand, you could imagine simply going to <https://maps.google.com> and typing in the location pairs by hand, and copy-pasting the resulting distance measurements to a spreadsheet. What I’m doing is essentially just automating that – I read origin / destination pairs from a spreadsheet, send those pairs to Google Maps, and automatically parse the results and append them to the spreadsheet.

Technical

The part of the Google Maps API I’m using is documented here: <https://developers.google.com/maps/documentation/distance-matrix/intro>

It’s super straightforward. Input looks like this:

[https://maps.googleapis.com/maps/api/distancematrix/json?units=imperial&origins=Washington,DC&destinations=New+York+City,NY|Austin,TX&key={MY\_API\_KEY}](https://maps.googleapis.com/maps/api/distancematrix/json?units=imperial&origins=Washington,DC&destinations=New+York+City,NY|Austin,TX&key=%7BMY_API_KEY%7D)

As you can see, we’re passing in a single origin (Washington, DC) and multiple destinations (New York City, NY & Austin, TX), and the API returns JSON that looks like this:

{

"destination\_addresses" : [ "New York, NY, USA", "Austin, TX, USA" ],

"origin\_addresses" : [ "Washington, DC, USA" ],

"rows" : [

{

"elements" : [

{

"distance" : {

"text" : "225 mi",

"value" : 361722

},

"duration" : {

"text" : "3 hours 48 mins",

"value" : 13668

},

"status" : "OK"

},

{

"distance" : {

"text" : "1,524 mi",

"value" : 2451975

},

"duration" : {

"text" : "22 hours 12 mins",

"value" : 79934

},

"status" : "OK"

}

]

}

],

"status" : "OK"

}

As you can see, there are two distance fields, corresponding to the two origin / destination pairs we passed in.

I used Python 3.6 to parse the input spreadsheet, grab the location pairs, and make the HTTP request to the API. The same script parses the Google Maps output, appends it to the relevant row(s) in the spreadsheet, and writes the resulting spreadsheet to disk.

Here are some assumptions I’m making:

* Excel is bad about treating numerical-looking text as numbers. This is a problem because the ZIP codes we’re using tend to all start with 0, which means that Excel truncates them to 4 digits instead of 5. To counteract that, if the ZIP code that I read out of the spreadsheet is less than 5 digits, I prepend it with a 0.
* I also prepend ‘USA’ to all of the ZIP code strings, because otherwise Google Maps thinks that some of them are in Brazil (or whatever)
* I’m assuming an input format identical to the one that Charlotte gave me as an example, but if that changes, I can change the script correspondingly without too much effort.