Now that we've taken the means of all of the columns, we can plot out ell\_percent by district. Not only did we find the mean of ell\_percent, but we also took the means of the lon and latcolumns, which will give us the coordinates for the center of each district.

Instructions

* Use the code snippet from before that creates a map, then draws rivers, coastlines, and boundaries.
* Convert the lon column of districts to a list, and assign it to the longitudes variable.
* Convert the lat column of districts to a list, and assign it to the latitudes variable.
* Call the [scatter()](http://matplotlib.org/basemap/api/basemap_api.html#mpl_toolkits.basemap.Basemap.scatter) method on m, and pass in longitudes and latitudes as arguments.
  + Make sure to pass in longitudes and latitudes in the correct order.
  + Pass in the keyword argument s=50 to increase the size of the points in the scatterplot.
  + Pass in the keyword argument zorder=2 to plot the points on top of the rest of the map. Otherwise the method will draw the points underneath the land.
  + Pass in the keyword argument latlon=True to indicate that we're passing in latitude and longitude coordinates, rather than axis coordinates.
  + Pass in the keyword argument c with the value districts["ell\_percent"] to plot the ell\_percent.
  + Pass in the keyword argument cmap="summer" to get the right color scheme.
* Show the plot using the [show()](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.show) method.

Answer

import pandas as pd

import matplotlib.pyplot as plt

from mpl\_toolkits.basemap import Basemap

m = Basemap(

projection='merc',

llcrnrlat=40.496044,

urcrnrlat=40.915256,

llcrnrlon=-74.255735,

urcrnrlon=-73.700272,

resolution='i'

)

m.drawmapboundary(fill\_color='#85A6D9')

m.drawcoastlines(color='#6D5F47', linewidth=.4)

m.drawrivers(color='#6D5F47', linewidth=.4)

longitudes = districts["lon"].tolist()

latitudes = districts["lat"].tolist()

m.scatter(longitudes, latitudes, s = 50, zorder = 2, latlon = True, c = districts["ell\_percent"], cmap = "summer")

plt.show()

