Data: NYC MTA turnstile data.  (source: <http://web.mta.info/developers/turnstile.html>)

**Goal: Calculate the average of weekday sum of morning exits and evening entries for each station for a particular week's worth of data.**

Download data for only one week of data: **Saturday, June 13, 2015**

**Procedures to follow:**

* Open up a file, use csv reader to read it, make a python dict where there is a key for each (C/A, UNIT, SCP, STATION). These are the first four columns. The value for this key should be a list of lists. Each list in the list is the rest of the columns in a row. For example, one key-value pair should look like

{ ('A002','R051','02-00-00','LEXINGTON AVE'): [ ['NQR456', 'BMT', '01/03/2015', '03:00:00', 'REGULAR', '0004945474', '0001675324'], ['NQR456', 'BMT', '01/03/2015', '07:00:00', 'REGULAR', '0004945478', '0001675333'], ['NQR456', 'BMT', '01/03/2015', '11:00:00', 'REGULAR', '0004945515', '0001675364'], ... ] }

The date is the third item in each list within the value list.

The time is the fourth item in each list within the value list.

The cumulative entries is the sixth item in each list within the value list.

The cumulative exits is the seventh item in each list within the value list.

* Turn this into a time series. For each key (basically the control area, unit, device address and station of a specific turnstile), have a list again, but let the list be comprised of just the point in time and the count of entries and exits.

This basically means keeping only the date, time, and entries and exits fields in each list. You can convert the date and time into datetime objects -- That is a python class that represents a point in time. You can combine the date and time fields into a string and use the [dateutil](https://labix.org/python-dateutil) module to convert it into a datetime object.

Your new dict should look something like

{ ('A002','R051','02-00-00','LEXINGTON AVE'): [ [datetime.datetime(2013, 3, 2, 3, 0), 3788, 4543], [datetime.datetime(2013, 3, 2, 7, 0), 3874, 4695], [datetime.datetime(2013, 3, 2, 12, 0), 3944, 5633], [datetime.datetime(2013, 3, 2, 17, 0), 4124, 6873], [datetime.datetime(2013, 3, 2, 23, 0), 4234, 7403], [datetime.datetime(2013, 3, 3, 3, 0), 4452,8439], [datetime.datetime(2013, 3, 3, 7, 0), 4651,9372], ... ], .... }

Now make it that we again have the same keys, but now we have a single value for entries and a single value for exits for a single datetime, which is the total number of passengers that entered and exited through this turnstile on that particular 4 hour window.

To get the number of entries for a particular starting datetime, you have to take the difference between cumulative entries for the starting datetime and the cumulative entries for the next later datetime.

To get the number of exits for a particular starting datetime, you have to take the difference between cumulative exits for the starting datetime and the cumulative exits for the next later datetime.

**By doing this, the last datetime will need to be dropped since there is no next datetime from which to calculate the difference for that last datetime.**

* So far we've been operating on a single turnstile level (SCP), let's combine turnstiles in the same ControlArea/Unit/Station combo (same station). There are some ControlArea/Unit/Station groups that have a single turnstile, but most have multiple turnstiles-- same value for the C/A, UNIT and STATION columns, different values for the SCP column.

We want to combine the numbers together -- for each ControlArea/UNIT/STATION combo, for each datetime, add the counts from each turnstile belonging to that combo.

Similarly, combine everything in each station, and come up with a time series of [(datetime1, exitcount1, entriescount1),(datetime2, exitcount2, entriescount2),...] type of time series for each STATION, by adding up all the turnstiles in a station.

Ultimately, we want to sum the **morning exits** and **evening entries** for each station for each day in the week’s worth of data. We define morning to be starting datetimes between the hours of 8am and 12pm and evenings to be starting datetimes between 4pm and 8pm. Then we sum the weekday sum of morning exits and evening entries and divide by 5 to get the average weekday sum of morning exits and evening entries for each station.

Finally sort the average by highest to lowest and plot the top 20 stations with the top 20 highest average weekday sum of morning exits and evening entries for each station in a bar chart with x axis as location (Station) and y axis as the average weekday sum of morning exits and evening entries.