

HW7: CS 110X C 2014

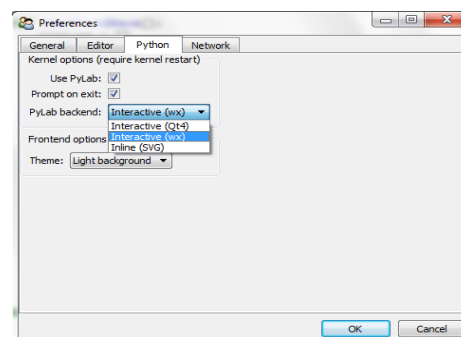
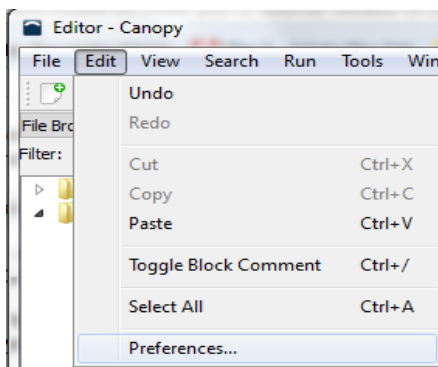
Note: This final homework is a **partner homework** and must be completed by each partner pair. When you complete this assignment, you must not share your answers with any other student. Only one person from a partner pair needs to submit the assignment, but make sure that you submit before the deadline!

For this assignment, every function that you write must have a suitable documentation string as we present in class. Check the rubric to see the point values assigned for each question so you can maximize the points you get on this assignment.

Please make sure that when you submit your assignment, you submit a single “HW7.py” file that contains your entire assignment.

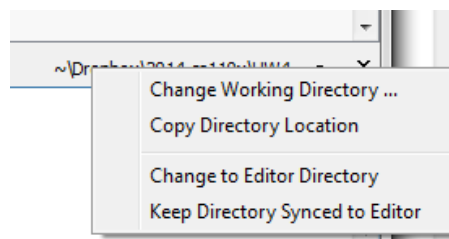
Canopy Issues

If you are running Canopy then you have to make a small configuration change for this homework to work properly. From within the Canopy Editor, select menu item **Edit | Preferences...**



Then in the Preferences window, select the **Python** tab and be sure that **PyLab backend** is set to “Interactive (wx)”. I have tested this on Windows and on a MacBook.

In Canopy, you need to set the working directory within the code editor. Near the right edge about one inch from the bottom you will see a small triangle that you can click on to change the working directory. Select the “Change to Editor Directory” so you will find the files that you need for this assignment.



Homework Themes

This homework will introduce students to working with online repositories of information. You will use a third party coding library to access the **forecast.io** API which returns weather-related data (both current and historical) for a given latitude, longitude. You will use your skills from working with CSV files.

First, you need to get a free account at **forecast.io**. Visit <http://forecast.io> and register for an account. Once you have done so, you can visit <https://developer.forecast.io> at any time to view the status of your account. Specifically, you have 1,000 free API calls per day, which should be sufficient for the homework assignment you are to complete. If you exceed this daily limit, you could open a second free account with your partner's email address, or a different email address that you have access to.

Note: UNDER NO CIRCUMSTANCE SHOULD YOU ENTER BILLING INFORMATION! OTHERWISE YOU MAY INCUR CHARGES THAT ARE NOT NEEDED FOR THIS ASSIGNMENT AND YOU WILL BE RESPONSIBLE FOR PAYING THOSE BILLS!

Once you are able to log into <https://developer.forecast.io>, you will see at the bottom information about your API Secret Key:



You will need this unique key for this assignment. Do not give this key information to anyone outside your homework partnership since you are limited to 1,000 API calls per day. Once you have this key, you can request information simply by typing the following URL into a browser:

<https://api.forecast.io/forecast/SECRETKEY/42.2667,-71.8000>

where you replace SECRETKEY with the exact contents of your API key. In doing so, you will see the full raw data that you can discover. Fortunately, you can write a program to process this information.

```
{"latitude":42.2667,"longitude":-71.8,"timezone":"America/New_York","offset":-5,"currently":{"time":1393363186,"summary":"Overcast","icon":"cloudy","nearestStormDistance":15,"nearestStormBearing":190,"precipIntensity":0,"precipProbability":0,"temperature":24.45,"apparentTemperature":15.98,"dewPoint":11.39,"humidity":0.57,"windSpeed":7.54,"windBearing":261,"visibility":9.97,"cloudCover":0.94,"pressure":1013.34,"ozone":442.38},"minutely":{"summary":"Overcast for the hour","icon":"cloudy","data":[{"time":1393363140,"precipIntensity":0,"precipProbability":0}, {"time":1393363200,"precipIntensity":0,"precipProbability":0}, {"time":1393363260,"precipIntensity":0,"precipProbability":0}, {"time":1393363320,"precipIntensity":0,"precipProbability":0}, {"time":1393363380,"precipIntensity":0,"precipProbability":0}, {"time":1393363440,"precipIntensity":0,"precipProbability":0}, {"time":1393363500,"precipIntensity":0,"precipProbability":0}, {"time":1393363560,"precipIntensity":0,"precipProbability":0}, {"time":1393363620,"precipIntensity":0,"precipProbability":0}, {"time":1393363680,"precipIntensity":0,"precipProbability":0}, {"time":1393363740,"precipIntensity":0,"precipProbability":0}, {"time":1393363800,"precipIntensity":0,"precipProbability":0}, {"time":1393363860,"precipIntensity":0,"precipProbability":0}, {"time":1393363920,"precipIntensity":0,"precipProbability":0}, {"time":1393363980,"precipIntensity":0,"precipProbability":0}],.....
```

Homework Instructions

This Homework has **Four** questions, and you are advised to complete them in order. You first must unzip the [following zip file](#) and place the contents on your computer exactly as they are stored in the file. Note that [this zip file](#) already contains the HW7.py file that you are to modify for this assignment. Do not move it, or the other files, because this will prevent the code from working. Note that a suitable **helper.py** is already included in [this zip file](#), as you had used for HW6.

```
import forecastio
import datetime

# You will have to insert your own API key here, which you can
# get after you have created a free account at forecast.io
# DO NOT CHANGE THE NAME OF THIS VARIABLE ONLY ITS VALUE
api_key = "<<SECRETKEY>>"
wlat = 42.2667
wlong = -71.8000

def main():

    # Retrieve current information for given (latitude, longitude)
    forecast = forecastio.load_forecast(api_key, wlat, wlong)
    summarize(forecast)

    # To create a specific date in history, enter (YYYY,MM,DD). Note
    # the 12 reflects 12 noon on the given day (March-31-1997)
    date = datetime.datetime(1997,3,31,12,0,0)

    forecast = forecastio.load_forecast(api_key, wlat, wlong, date)
    summarize(forecast)

def summarize(forecast):
    """Given a forecast object, print useful statistics"""

    dailyForecast = forecast.daily()
    print "Daily Summary: ", dailyForecast.summary

    # Get information for just the first day in forecast (or history)
    today = dailyForecast.data[0]
    print "Today: ", datetime.datetime.fromtimestamp(today.utime)

    print "Max temperature: ", today.temperatureMax
    print "Min temperature: ", today.temperatureMin
    print "Precipitation: ", today.precipType
    print "Precipitation Acc. ", today.precipAccumulation
```

There are Python files present in the ZIP file that you need to review. Specifically, **sample.py** shown above which demonstrates how you can use the forecast API to retrieve information about weather-related information for any given latitude, longitude coordinates.

Version: 2/26/2014

Try running the **sample** module once you have properly entered your SECRETKEY into the value for `api_key`. If you do not exactly type in the value, you will get the error "No JSON object could be decoded". However, if you properly enter a valid SECRET KEY, then you will see the following output when you execute **main()**.

```
Daily Summary: Light snow off-and-on throughout the week; temperatures peaking at 30° on Saturday.
```

```
Today: 2014-02-26 00:00:00
```

```
Max temperature: 25.93
```

```
Min temperature: 7.69
```

```
Precipitation: snow
```

```
Precipitation Acc. 0.058
```

```
Daily Summary: None
```

```
Today: 1997-03-31 01:00:00
```

```
Max temperature: 41.05
```

```
Min temperature: 27.53
```

```
Precipitation: snow
```

```
Precipitation Acc. 8.964
```

If you were in Massachusetts on **April 1 1997**, you may remember that was the day we had 30 inches of snow suddenly dumped down on us.

This homework will give you the opportunity to write Python code that accesses online information repositories. For this project the sample code demonstrates all the functionality that you need.

In the Homework 7 zip file there is a [USCapitals.csv](#) file that contains information from the 2010 Census about the 50 US capitals. Here are the first few rows:

abbrev	state	capital	latitude	longitude	population
AL	Alabama	Montgomery	32.38012	-86.3006	205764
AK	Alaska	Juneau	58.29974	-134.407	31275
AZ	Arizona	Phoenix	33.44826	-112.076	1445632
AR	Arkansas	Little Rock	34.74866	-92.2745	193524
CA	California	Sacramento	38.57907	-121.491	466488

You will need to process this data in this homework assignment. Once again, the **helper** module (included in the zip file) will prove useful.

For each question be sure you understand exactly the format of the output that is requested. You will lose points if you do not exactly follow the format of the output for the individual questions. Should you have any questions, be sure to review the HW7 rubric and post questions on the HW7 discussion forum.

Q1	List manipulation
<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Skills</div> <div style="border: 1px solid black; padding: 2px;">TBA</div>	<p>Write a function <code>decompose(values)</code> that returns a list of lists. If the length of values is not a perfect square then an empty list <code>[]</code> is returned. If the length = k^2 then this will return a list of k lists, each of which contains k elements.</p> <p>The elements will retain their same ordering but will be subdivided as shown below.</p> <p>Hint: Work this problem out manually and then try to write the logic.</p> <p>Note: <code>math.sqrt(n)</code> will be useful</p>
Sample Output in IDLE	<pre>>>> decompose([1,2,3,4]) [[1, 2], [3, 4]] >>> decompose([1,2,3]) [] >>> decompose([1,2,3,4,5,6,7,8,9]) [[1, 2, 3], [4, 5, 6], [7, 8, 9]]</pre>

Q2	Access Forecast API
<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Skills</div> <div style="border: 1px solid black; padding: 2px;">TBA</div>	<p>Write a function <code>currentDailySummary(latitude, longitude)</code> that returns a textual summary of the current day's forecast for the given latitude and longitude. Specifically, you need to print the [High Low] temperatures and Summary for today.</p> <p>Hint: review the sample.py file showing how to access the forecast API. Note that your <code>HW7.py</code> file must have your API key so both you and the TAs can execute it.</p> <p>The second latitude/longitude is Honolulu. Where would you rather be today?</p> <p>Note that the output changes daily, so your output may be different!</p>
Sample Output in IDLE	<pre>>>> print currentDailySummary(42.279167, -71.416667) High=28.86 Low=9.91 Mostly cloudy until evening. >>> print currentDailySummary(21.3, -157.816667) High=74.36 Low=68.65 Breezy in the evening.</pre>

Q3	List manipulation and API access
<div style="border: 1px solid black; padding: 2px; margin-bottom: 2px;">Skills</div> <div style="border: 1px solid black; padding: 2px;">TBA</div>	<p>Write a function <code>weatherForPeriod(latitude, longitude, year, month, startDay, endDay)</code> that returns a tuple containing two lists (LOW, HIGH). LOW contains the minimum temperatures for a range of days in the given month and year while HIGH contains the maximum temperatures for a range of days in the given month and year. The desired (latitude, longitude) coordinates are the first two parameters for this function.</p> <p>Note: the return value is a tuple. These values won't change because they represent historical data.</p> <p>Be aware that the values [startDay, endDay) are inclusive/exclusive as we have talked about in class. So the output below shows the low and high temperatures for the 10th, 11th, and 12th of February 2014 for the given lat/long coordinates.</p>
Sample Output in IDLE	<pre>>>> weatherForPeriod(42.2667, -71.8, 2014, 02, 10, 13) ([9.48, 3.35, -0.77], [23.39, 20.55, 23.32]) >>> weatherForPeriod(21.3, -157.816667, 2014, 02, 10, 13) Out[146]: ([71.69, 70.02, 71.83], [80.85, 81.03, 79.33])</pre>

Q4	Process CSV files
<div style="border: 1px solid black; padding: 2px; margin-bottom: 2px;">Skills</div> <div style="border: 1px solid black; padding: 2px;">TBA</div>	<p>Write a function <code>capitalSummary(listOfCapitals)</code> that prints information about the US capitals.</p> <p>Specifically, it produces a tabular report of the populations of each state's capital and the percentage of people in that state living in the capital.</p> <p>Upon completion summary information shows the least populous and most populous state capitals. The output below shows just the first five states and the last two states, together with the summary.</p>
Sample Output in IDLE	<pre>>>> caps = helper.extractAllRecords('USCapitals.csv') >>> capitalSummary(caps) 205764 Montgomery,AL 0.0428408851001 31275 Juneau,AK 0.043345811568 1445632 Phoenix,AZ 0.225432657071 193524 Little Rock,AR 0.0661342635863 466488 Sacramento,CA 0.0124923179641 ... 233209 Madison,WI 0.0409265684256 59466 Cheyenne,WY 0.104638395214 Total population in state capitals:12234921 Least Populous capital: Montpelier,VT Most Populous capital: Phoenix,AZ</pre>

How To Get Started On This Assignment

A ZIP file is provided with the template and initial data sets.

Note: You only have to submit your modified HW7.py file using the web-based turnin system. As we have mentioned in class, only one of the team members needs to submit the assignment. But just make sure that something gets submitted!

What happens to your Forecast.IO account?

You can read the [terms and conditions](#) for this account. To discontinue the licensing agreement with Forecast, simply discontinue using the API. To be safe, you should reset the API key in Forecast.IO once you have received a grade for HW7. You can do this from the main login window:



Change Log

1. Note that the format of the output for Question 4 now matches the description. Be sure to compute the percentage of people in a state that live in its capital.
2. I fixed the defect in the Forecast API that wasn't working today in class. Be sure you download the [latest ZIP file](#).
3. NOTE: Updated last question to explain where **caps** came from...