Homework 2 - Areas on a Sphere

When preparing your homework submissions, don't forget to `git fetch, git status, git pull` *before you issue any other commands in Git – this is to guard against you changing a document that someone else is working on in the same directory*[1].

Don't forget to `git add` *and* `git commit` *(with* `-m` *comments) frequently as you work. This allows other users to see how your work progressed and it automatically backs up your work as you produce. Thus, you're less likely to lose any of your work and/or so you can revert to earlier versions of your work as needed.*

Remember to comment your code carefully with your initials before every comment (as in # JCR `I just wrote a Python comment to document a change.` *Remember to provide an informative header for* **every** *function that you write. Also provide a README file to inform people how to run your code.*

Homework

1. Astronomers often survey "square" fields in the sky that have corners (vertices) in right ascension and declination $(\alpha_{min}, \delta_{min})$, $(\alpha_{max}, \delta_{min})$, $(\alpha_{max}, \delta_{max})$, $(\alpha_{min}, \delta_{max})$. Write a function to determine the area of such a general field in square degrees. Plot 4 areas that are progressively higher in declination for the same α_{min} and α_{max} and label the regions with their areas. Confirm that your function returns the correct area for a spherical cap as well as for a "square" field by inputting $(\alpha_{min}, \alpha_{max}, \delta_{min}, \delta_{max}) = (0°, 360°, 0°, 90°)$.

2. Write a function to randomly populate a "square" field drawn on the surface of a sphere. The procedure should take $\alpha_{min}, \alpha_{max}, \delta_{min}, \delta_{max}$ and return a set of (α, δ) coordinates that correctly populate that "square" field randomly in area. Use your function from the first item above to confirm that the areas you are populating contain the correct number of random points relatively to populating the entire sphere (*hint: remember that the area of the entire sphere is 4*np.pi*180.*180./np.pi/np.pi and refer to my equations for randomly populating the sphere from lectures).*

In my `week3` directory in `Git`, there is a list of quasars called *HW1quasarfile.dat*[2]. This is a list of 1,111 $g = 18$ ("18th magnitude") quasars that I've drawn from the Sloan Digital Sky Survey. Provided in the file are coordinates of the quasars in base-60 (*hms.ss* $\circ\ '\ ''$) format. This is the same file we used for HW1.

3. Write a procedure that determines the pixel number at $Nside = 4, Nside = 8$, and $Nside = 16$ of the HEALpix (ring) hierarchy for these quasars[3]. Create a *recarray* with tags *ra*, *dec*, and *pixnum* to store this information, where *pixnum* is a 3-array. Write your structure out to a fits file.

[1] This shouldn't be a big deal unless we're working collaboratively, but you should get into the habit *now*.

[2] In general, it is *not* a good idea to store large data files in Git as it slows down updates for all users, but this particular data file is very small.

[3] At $Nside = 4$ there are 192 total pixesl in the sky and each pixel has an area of 214.86 deg^2, at $Nside = 8$ there are 768 total pixels in the sky and each pixel has an area of 53.71 deg^2,and at $Nside = 16$ there are 3072 total pixels in the sky and each pixel has an area of 13.41 deg^2.

4. Write a procedure that reads the recarray from the *fits* file created in the previous step, plots the location of all the quasars in (α, δ), and over plots those quasars that lie in the 5 most over-dense[4] pixels at $Nside = 4$ of the HEALpix (ring) hierarchy in a different color using a different symbol.

[4]i.e. the 5 pixels that have the largest number of quasars per square degree