

## OS Python week 3: Filters and simple analysis of vector data; Functions and modules

### Assignment 3a

Use filters and buffers to find the points in sites.shp that are within 1500 meters of the Nibley town boundaries.

1. Import the needed modules and set the working directory
2. Get the shapefile driver
3. Open sites.shp and get the layer
4. Open cache\_towns.shp and get the layer
5. Use an attribute filter to restrict cache\_towns.shp to rows where NAME = 'Nibley'
6. Get the Nibley geometry and buffer it by 1500 (Hint: There should only be one feature visible in the layer at this point because all non-Nibley ones have been filtered out)
  - a. Get the feature
  - b. Get the geometry
  - c. Buffer the geometry
7. Use the geometry you got in step 6c as a spatial filter on sites.shp
8. Now the only features visible in sites.shp are ones that fall within the 1500m buffer around Nibley town limits. Loop through these and print the value of the ID field for each of them
9. Close the two shapefiles

Turn in your code and screenshot of what got printed.

## Assignment 3b

Write a function to reproject a shapefile and then use it to reproject all of the shapefiles with this week's data.

1. Modify your solution to assignment 2b so that all of the work is inside of a function.
  - a. Make sure that the filenames and EPSG codes are not hard-coded but are passed in as function parameters instead
  - b. No need to set the working directory
  - c. Don't import the modules inside the function
  - d. Use the copyFields function that I provided to create the fields in the output shapefile instead of doing it for just the name and id fields like you did for homework 2b
  - e. Use the copyAttributes function that I provided to copy all of the attributes from an input feature to an output feature instead of doing it for just the name and id fields like you did for homework 2b
  - f. Make sure that your function creates the correct filename for the .prj file
2. Put your function in a module
  1. Import other modules at the top of your module, but not inside any of the functions
  2. Include the functions that I provided in functions.py
  3. Make sure you include your function from step 1!
3. Write another script that uses your new module to reproject the shapefiles
  - a. Import the needed modules and set the working directory
  - b. Use glob to get a list of all of the .shp files in the working directory
  - c. Loop through this list of shapefiles
    - i. Create a name for the new shapefile
    - ii. Call your function that's inside your module, using an EPSG code of 26912 (UTM 12N NAD83) for the input shapefiles and an EPSG code of 4269 (unprojected NAD83) for the output shapefiles

Turn in your module, your script that uses your module, and a screenshot of the projected shapefiles being displayed (make sure that they're not projected on-the-fly so that I can see that they're in the correct projection).