Geography 378: Introduction to Geocomputing
Lab 6: OGR and Python

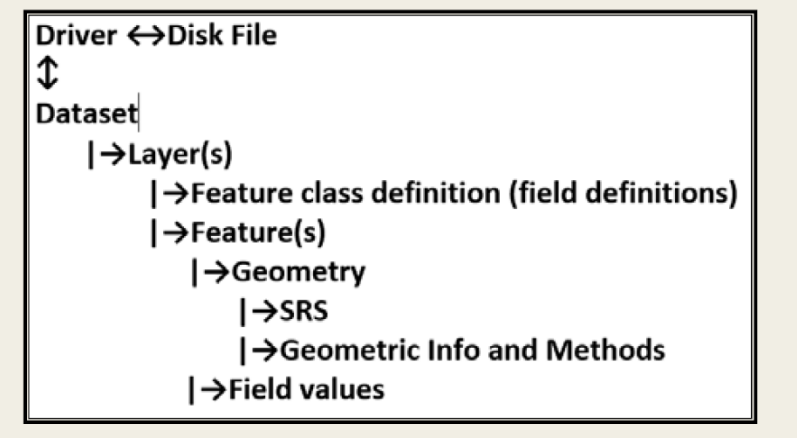
Assigned: 11/17
Due: 11/30
(Lab on Wednesday, Nov 24th will be an optional Q&A session).
15 points

Hand-in

- Please collect your answers in a single .py file called **lab6_yourname.py**.
- Submit the file to the assignment folder called "Lab 6".
- Include appropriate comments to explain what each line or block of code accomplishes. **You must comment your code for full credit.**

Background

Lab 5 provided experience using GDAL with Python. Now we turn to the OGR library, which exposes methods for vector (feature) processing. Once again you should carefully review the lecture notes and examples---the tasks below build heavily on them. Below is an OGR class.



In this exercise you will read two Shapefiles: one is a polygon layer containing land parcels in the city of Santa Monica, California, and another is a linestring layer containing the path of a hypothetical proposed power line (created for this exercise). Both files are in the same coordinate system having ground units of **feet**.

Lab Task

1. (5 pts) Using OGR, write a Python script to read PowerLine.shp and calculate the length of the power line **in miles**. You can use the functions below to get point coordinates in the line feature.

A LineString (Polyline) is an ordered set of Points. The position of a point within a string is given by 0-based index:

```
line  = feat.GetGeometryRef()
nPts = line.GetPointCount()

thirdPoint = line.GetPoint(2)
```

Directions:

i. The line length is the sum of the distance between consecutive points within that line feature.

ii. You are NOT allowed to directly use the geometry length() method, but you can test your result using this method. If your program works properly, the answer will be 2.17 miles.

2. (10 pts) Extend the program in task 1 so that it also opens Parcels.shp.
   a. Print out the **attribute names** and **attribute data types** of the layer. (Note that this question does not ask for attribute values of any parcel.)
   b. Print out the owner's address (field "SITUSADDR") and the area of every parcel that is **crossed by the power line**.

   Hints: use poly2.Crosses(line) can test whether poly2 crosses line.