Deliverable 2: **Create a Feature Class from XY data**

Assigned Date: **Wed/Thu, 2015-02-04/05** (week 04)

Due Date: **Wed, 2015-02-25, 08:30 AM** (week 07)

Grade Value: **30%** of Final Grade

# Important

1. *Plagiarism is a serious offense and will not be tolerated! Plagiarized assignments will receive a grade of 0 (zero) as per college policy. In addition, professionalism/participation grades will be reduced to 50%. Please refer to the official college policy on academic misconduct which can be found here:* [*http://www.niagaracollege.ca/Content/LinkClick.aspx?fileticket=i-eJpagoNDQ%3d&tabid=1001*](http://www.niagaracollege.ca/Content/LinkClick.aspx?fileticket=i-eJpagoNDQ%3d&tabid=1001)*.*
2. *Save your work periodically.*
3. *The assignment will be evaluated in terms of both functionality and industrial standard coding style.*

# Background:

In addition to shapefiles and geodatabase feature classes, GIS data source may come from a plain tabular data that contain geographic locations in the form of (x, y) coordinates. The tabular data can also be added as 3D content if the table contains z-coordinate (such as elevation values).

(x, y, [z]) coordinates describe point locations of the geographic features on the earth's surface. The coordinate data could be in either Geographic Coordinate System (GCS) or a Projected Coordinate System (PCS) and collected using survey equipment, such as total station and GPS units.

(x, y, [z]) table must contain two numerical fields: one for the x-coordinate and one for the y-coordinate. The values in the fields may be represented as any coordinate system and units, such as GCS's latitude and longitude in decimal degrees or UTM's northing and easting in meters. A field for the z-coordinates that enables 3D geometry is optional.

# Purpose:

Write a piece of Python script to create a point feature class from a (x, y) text file and assign the appropriate spatial reference system.

# Procedures:

1. Copy the \assignments\d2CreatePointFeatureclass\d2RawData subfolder into C:\temp (create C:\temp subfolder if necessary).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table 1. Example Data of** farm.txt | | | | |
| **FarmId** | **Name** | **FarmType** | **EastingM** | **NorthingM** |
| 1 | Bilski Farm | Roadside Market | 613321.675 | 4784393.313 |
| 2 | Blueberry Knoll Berry Farm | Pick-Your-Own Farm | 621509.927 | 4753230.424 |
| 3 | Bohonos Farms | Roadside Market | 623095.369 | 4781952.345 |
| …… | | | | |

1. There are two text files, farm.txt and winery.txt, within \d2RawData subfolder. Both text files have the same data structure: 5 columns: FarmId, Name, FarmType, EastingM, and NorthingM. The EastingM and NorthingM columns are the coordinates based on the spatial reference system of **NAD\_1983\_UTM\_Zone\_17N**. Load them into Excel workbook to examine the provided data.
   1. The farm.txt file contains the location data of 45 farms in the Niagara Peninsula. The first three records of the farms are shown in Table 1.
   2. The winery.txt file contains the location data of 46 wineries in the Niagara Peninsula. The first three records of the wineries are shown in Table 2.
2. Use the contents of the textbook (through Chapters 1 to 8) to write a piece of Python script, named as LastNameInitialGisc9307D2.py in your own H:\ drive, to:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table 2. Example Data of** winery.txt | | | | |
| **FarmId** | **Name** | **FarmType** | **EastingM** | **NorthingM** |
| 1001 | 20 Bees Winery | Winery | 650838.901 | 4785333.131 |
| 1002 | Angels Gate Winery | Winery | 622162.316 | 4780289.768 |
| 1003 | Cave Spring Cellars | Winery | 632539.904 | 4778749.123 |
| …… | | | | |

* 1. Create a new \d2ProcDataLastNameInitial subfolder within c:\temp subfolder. Delete the new subfolder if it exists before create it. The d2ProcDataLastNameInitial subfolder is used to save your own processed results.
  2. Create both farm.shp and winery.shp shapefile based on the provided text files with the desired spatial reference system of **NAD\_1983\_UTM\_Zone\_17N**.
  3. Create a single shapefile npFarm.shp (with the same spatial reference system) that contains all farms (the combination of farm.txt, winery.txt, and other potentially existing text files) located in the Niagara Peninsula.
  4. Save all created shapefiles in \d2ProcDataLastNameInitial subfolder.
  5. Given a rectangle extent, that is defined by XMin = 610000, YMin = 4760000, XMax = 660000, YMax = 4780000, print out the number of farms are located within the rectangle.

# Special Notes:

* 1. Make sure your codes, LastNameInitialGisc9307D2.py, will be working properly on your instructor’s machine, where the c:\temp\d2RawData subfolder, containing the required farm.txt and winery.txt, already exists on the instructor’s machine.
  2. Hard coding should be avoided. Assume that there are several dozens of text files of the same structure within \d2RawData subfolder; the script snippet is able to create all shapefiles based on the base names of the corresponding text files. For example, if there is a farm.txt, then a shapefile named as farm.shp will be created.
  3. All five columns (FarmId, Name, FramType, EastingM, and NorthingM) should appear in the corresponding attribute tables.
  4. [Challenging] The codes for determining the number of farms are located within the given rectangle could be modified to answer more specific questions, such as ‘how many farms are there for each farm type within the given feature extent?’

# Deliverables:

Send only the LastNameInitialGisc9307D2.py to your instructor by email at [jjiang@niagaracollege.ca](mailto:jjiang@niagaracollege.ca). If your mail server fails to send the .py file, then compress the .py file using 7-Zip application instead (do not use WinZip, WinRar, or other compression applications) and resend.

# Marking Scheme:

* 1. 10% Email etiquette and formal transmittal letter;
  2. 80% Functionality and professional coding style;
  3. 10% File and folder structure.