Intermediate GIS Work Book

For ArcMap 10.2

Geodatabase

- XML
- Domains
- Subtypes
- Linear Referencing
- Feature Linked Annotation
- Attachments
- Representations
- Metadata
- Projections

Editing

- Points Lines and Polygons
- Tables
- Georeferencing
- COGO

Geoprocessing

- Tools
- Model Builder
- Python

Topology

GPS

- X,Y data
- Excel tables
- Garmin

Bathymetry/Lidar

Data Driven Pages

ArcOnline

Tips & Tricks

- Customizing ArcMap

Topic: Installing Class Data
Problem:

Solution:

Data used: USB Stick , CD or Zip file

License: Basic

All the data you need for this class are in these folders. Transfer the Folders **GISClass** to **C:\Student\.** If the student folder doesn't exist then create on first. Optional: Transfer the folder Kachemak_Bay_DEM_1239 for one exercise with spatial analyst , and the **LAS folder** for the LiDar with 3_D Analyst extension lesson. These last two folders a very large.

Topic: Setting up a geodatabase for this class
Problem: First, we need to bring the data that we will be using into a geodatabase

Solution: We will create a geodatabase, then import an XML Document that contains some the data for the class

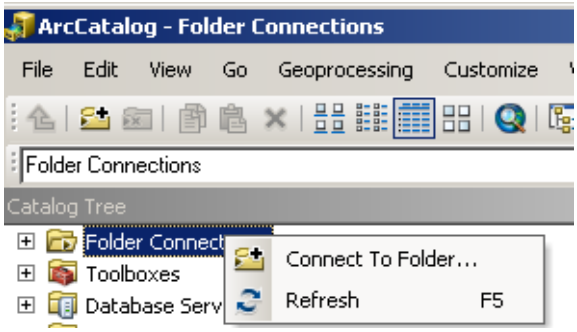
Data used: Kachemak.xml

License: Standard (Not available in basic)

Open ArcCatalog

1 in the catalog tree, under **Folder Connections,** navigate to your C:\ drive.

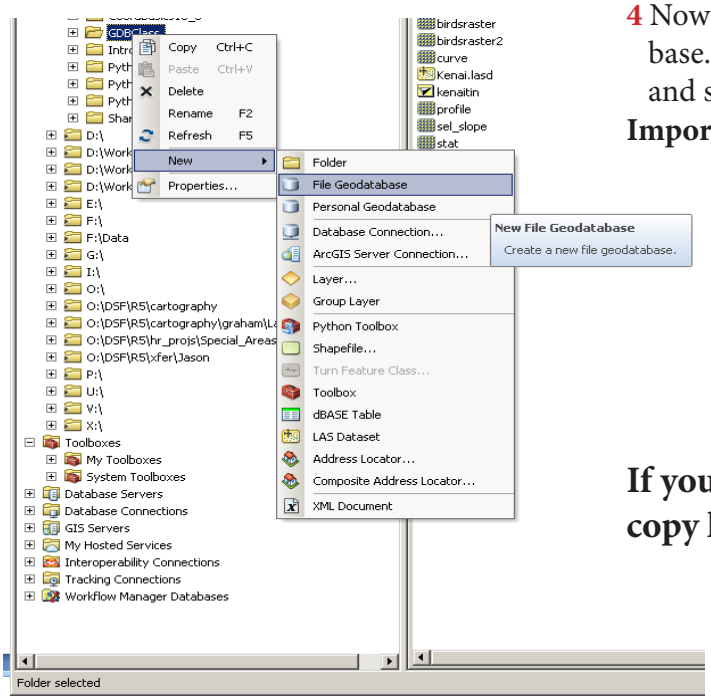
If you do not see it right click on the Folder and



select **Connect to Folder** and select C:\

2 Create a folder under
C:\Student\GDBClass

3 Right click on GDBClass and select **New > File Geodatabase**, name it KBayWildlife



4 Now we will import the class data into this new geodata base. Begin by right-clicking on the **KBayWildlife.gdb** and select:

Import > XML Workplace Document

Select the .xml file from the Student data folder and make sure the **Data** button is selected
Click **Next** and **Finish.**

This process brings all the data and schema that was created in another geodatabase.

If you dont have Standard or Advanced, there is a copy here:

\GISClass\Data\BU

Topic: Domains and Subtypes of Feature Classes
Problem: Typing is tedious, errors can be made in spelling or syntax rendering your data useless. By utilizing **Domains** and **Subtype**s you can eliminate user input errors.

Solution: In this task we will create a new feature class that has both **Subtypes** and **Domain**s aiding tagged bird documentation. Subtypes are unique to a feature class, Domains are used across a geodatabase

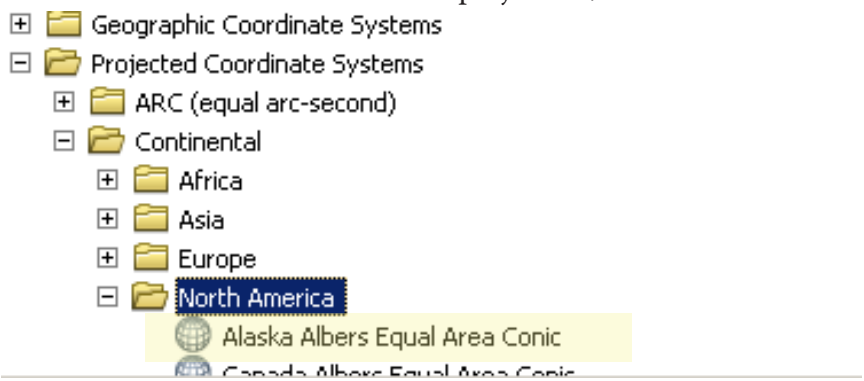
Data used: KayBayWildlife.gdb/Infrastructure/Alaska

License: Basic

Open ArcCatalog 

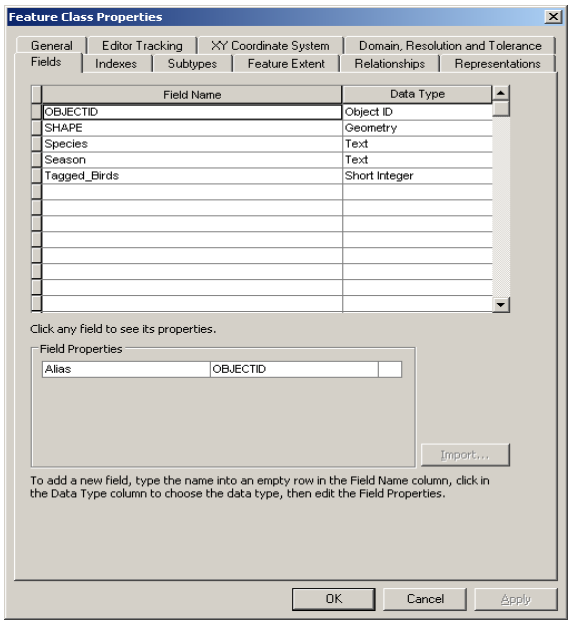
1 Right click on **KBayWildLife.gdb** select **New> Feature Dataset**, name it : **"Waterfowl"**

2 Select projection, use Alaskan Albers then click **Next, Next, Finish**



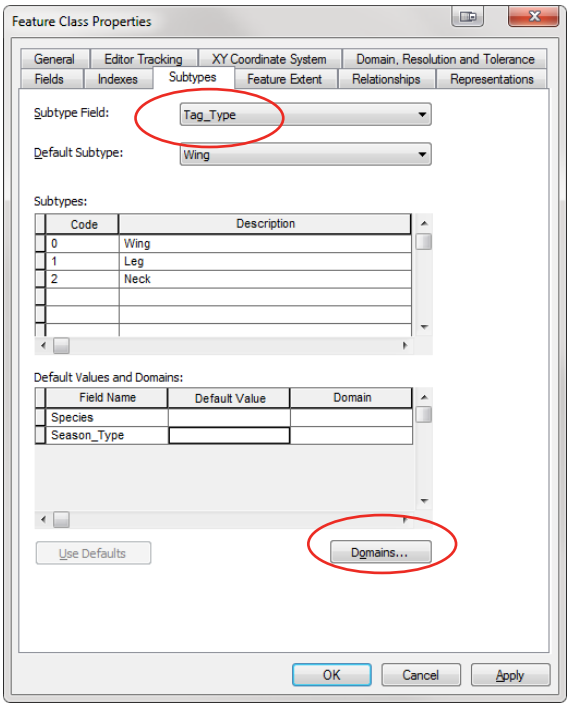
Do not use spaces when naming anything in ArcGIS. Use an underscore. You can use spaces in the Alias

3 Right click the **Waterfowl** Feature data set select **New > Feature Class**, name it: **'Tagged_Birds'** and in the type drop down box select Point Features, click **Next, Next.**



4 In the **Field Name** column add '**Species**' with **Data Type = Text, 'Tag_Type' = Long Integer, and 'Season_type' = Text.** Click **Finish**

5 Right click on **Tagged_Birds** and open its properties



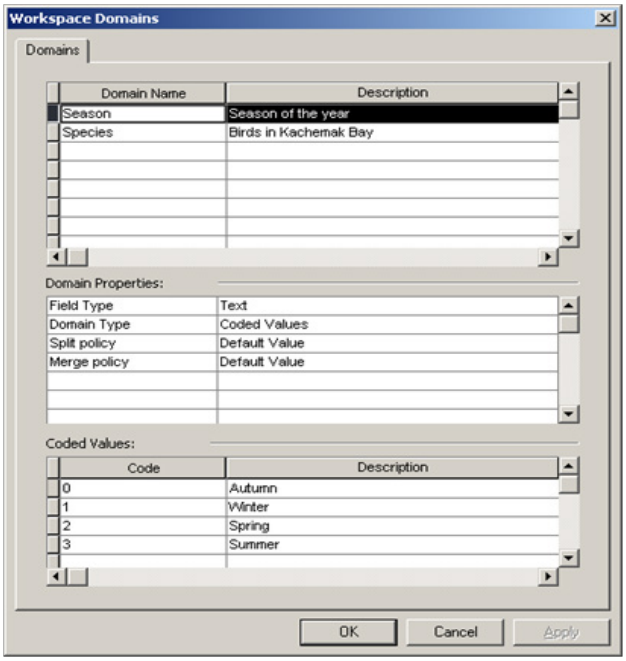
6 Open the **Subtypes** tab and add Tag_Type to the Subtype Field from the drop down list. Add the codes as follows:

0 = wing
1 = leg
2 = neck

7 Now click the **Domains** button on the bottom add two more domains called **Season_Type** and **Species**, Field type = text Domain type = coded values add these coded values to:

Season_Type	to **Species** add
0 Autumn	0 Kittlitz Murrelet
1 Winter	1 Tufted Puffin
2 Spring	2 Steller's Jay
3 Summer	3 Black Oystercatcher

Click **OK**



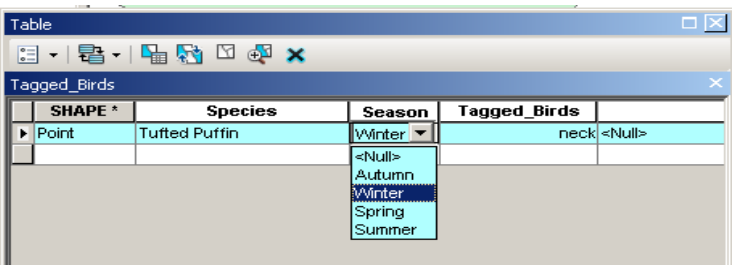
8 Back in the **Subtypes** tab, set the domains by clicking on the Domain field and selecting the corresponding domain. You must set each subtype
Then go to the **Fields** tab. In the **Fields** tab set the domains of **Species**, **Season_Type** in the field properties.

Click **OK**

9 Open ArcMap, add feature classes: Alaska_Clipped (GISCLass\Data\Editing\Editing.gdb) and Tagged_ Birds, Zoom in to Homer, Alaska.

10 Open the editor tool bar, select Editor/Start Editing and add a point. Open the Attribute table or the Attribute Window. You can access the attribute window by clicking on the **Editor** button on the Editor toolbar

>Editing Windows > Attributes



Add ten more points and randomly attribute your data as as you go. Save and stop editing

Topic: Linear Referencing
Problem: We need to designate the mile posts along the Sterling Highway

Solution: Calculate the M(measurement) values of the vertices of an arc line

Data used: KayBayWildlife.gdb/Infrastructure/Alaska, roads

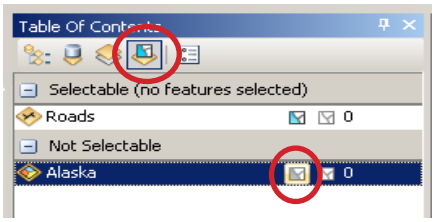
License: Basic

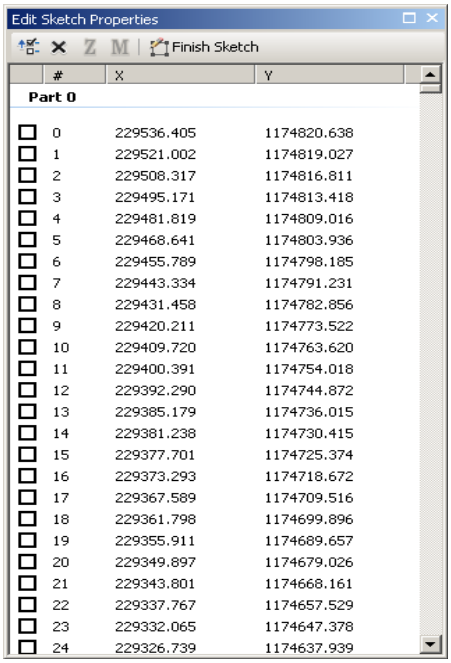
---

1 Open ArcMap and add data: Alaska and Roads. Right click on the Roads layer in the TOC and select zoom to layer.

2 If it is not all ready open, open the **Editor** tool bar by right clicking on the empty gray area on the top part of ArcMap

3 Start an editing session by clicking on the Editor button and selecting **Start editing**



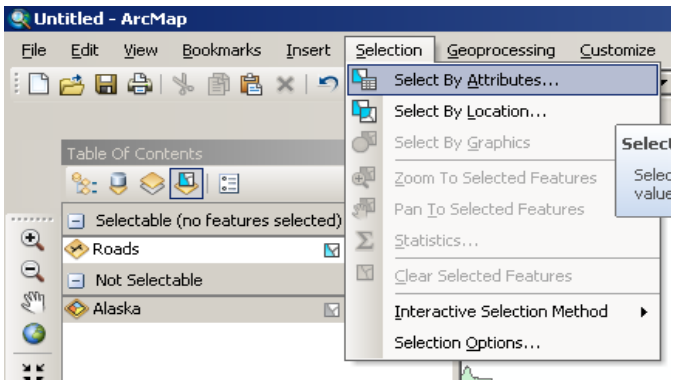


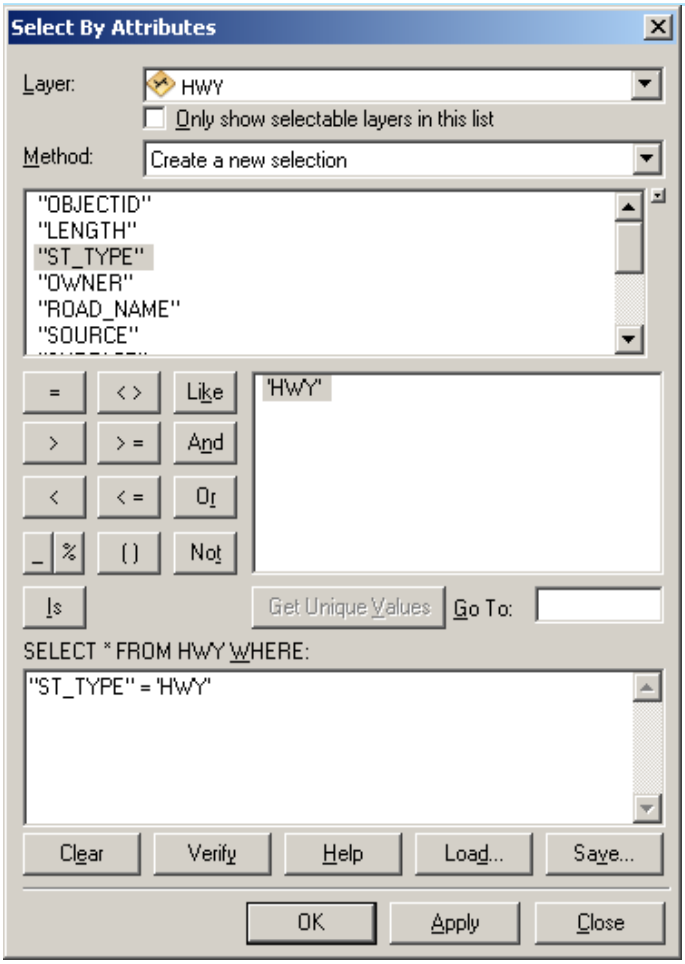
4 in the **TOC**, select the **List by Selection** button

5 De-select **Alaska** by clicking on the blue button, doing this means that only **Roads** can be selected

6 Select a Sterling Hwy, right click on it and select **Edit Vertices icon,** open the **Edit Sketch Properties** window  This displays the values of each vertex on a line, note that it only shows the x and y values.

7 Open the **Select by Attribute** window





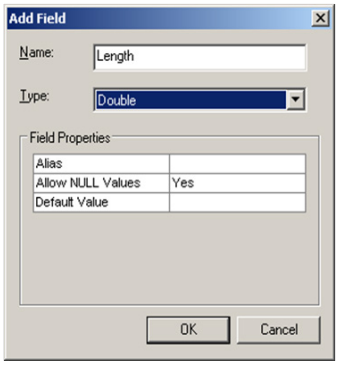
8 Set the **Select By Attributes** as shown here

9 In the table of contents, click on the **List by Drawing Order Button** then, right click on **Roads** and select **Data > Export Data** we are going to export just the highway to its own feature class. Make sure the top drop down says Export: Selected features and set the out put to go to the **Infrastructure dataset** and name it **HWY**

10 Open **ArcToolbox**  and find **Linear Referenc ing Tools > Create Routes.**

Input Line Features = 'HWY'
Route Identifier Field = 'Road_Name'
Set the output to the same dataset and name it 'HWY_ Route'. When it is done processing it should automati cally come into **ArcMap**

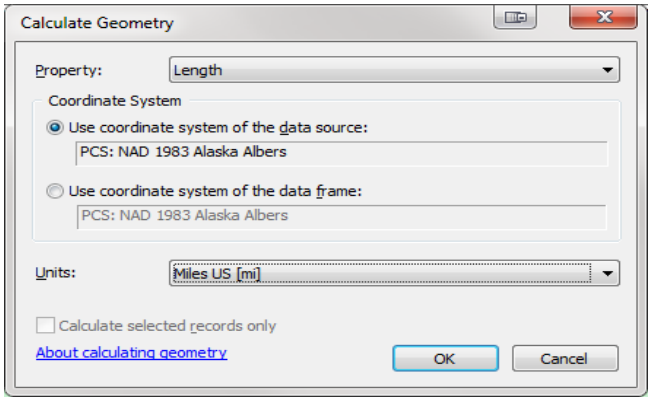
11 Now start an editing session on the new feature class and examine its sketch properties. Notice that now it has an **M** value. Stop Editing



12 Open the attribute table for **HWY_Route** and add a new field called 'Length' and set type to double

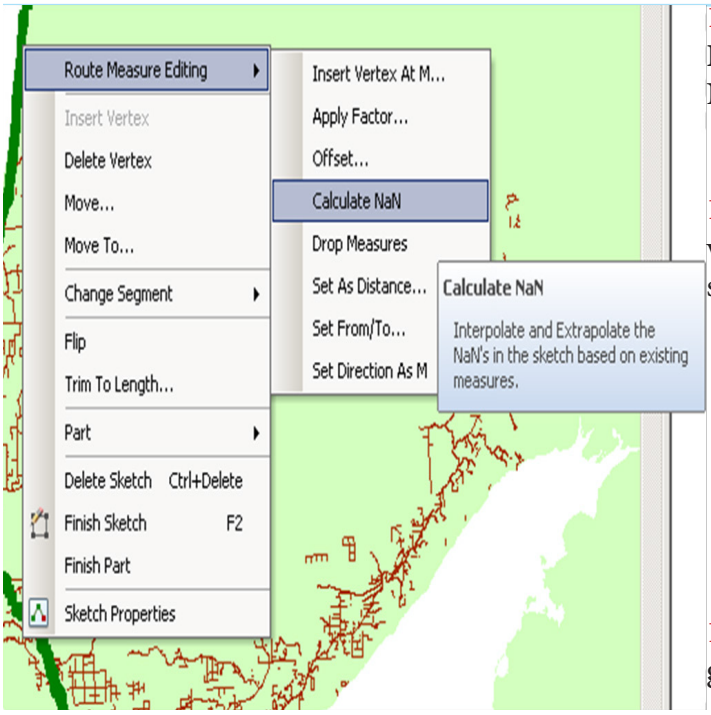
13 Right click the field name and select 'Calculate Geom- etry', Change units to miles and click ok and write down the length _____ then close the table.

14 Start an editing session to edit **HWY_Route**. Make sure that only **HWY_Route** is selectable. Double click on the Highway line with the arrow from the Editor toolbar. The Edit sketch properties window should open on the right side. Notice that now there is an M value assigned to each vertex.



15 Now, right click on the Highway arc and select **Route Measure Editing > Drop Measures** , this will clear the values from the M column so that can calculate our own values.

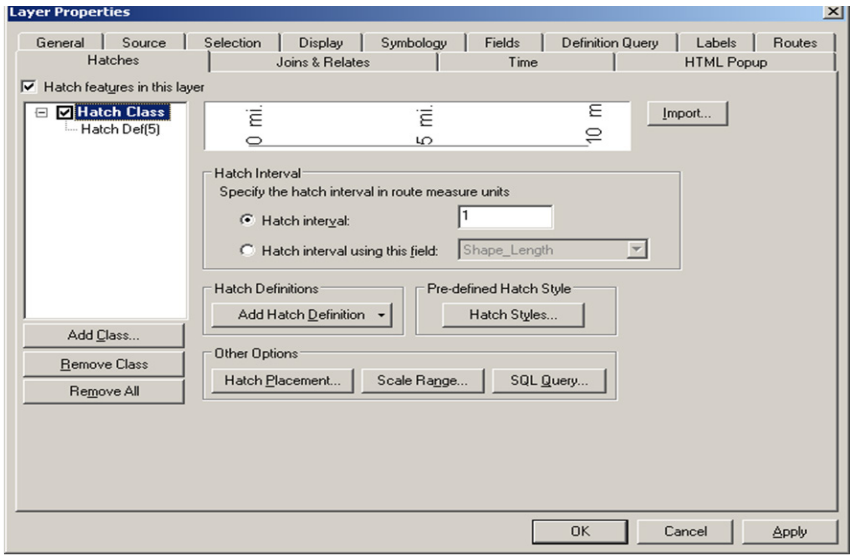
16 In the Edit sketch Properties window with the arc se- lected, scroll to the vary bottom and enter the number we recorded earlier into the '**M**' value, and at the very top put a 0 in the '**M**' value.



17 Right click on a node of the selected arc and select **Route Measure Editing** then select **Calculate NaN**, Now every point has a measure in miles.

18 Next, you will learn how to label, using the new **M** values. Save and stop editing. Save your project to your student folder.

19 Right click on **HWY_Route** and select properties and go the **Hatches** tab



Set **Hatch Interval** to '1'
Click on the first 'Hatch Def' in the left window

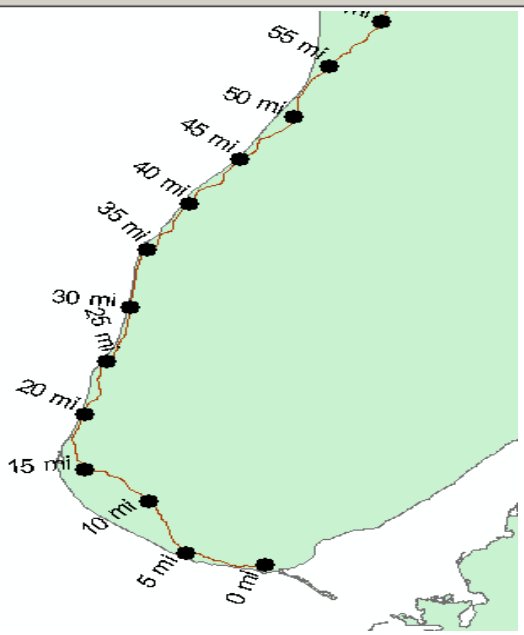
Set **Hatch Interval** to '5'

Set the **Hatch** to **Marker**

Check the **Label theses hatches** box and enter Label Settings

Add 'mi.' To the **suffix** window

Then click OK, OK



Be sure to check the box in the upper left corner to **Hatch Features in this layer**.

Now the layer should be layered every five miles. Try adjusting the hatch intervals for different results

Save your ArcMap project

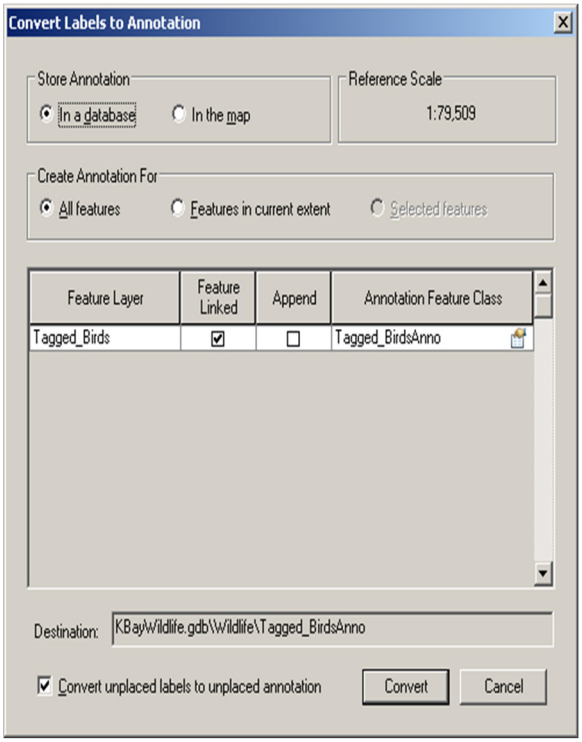
Topic: Feature Linked Annotation
Problem: Using labels gives you limited editing options. Just using text to label your map can be tedious and frustrating every time you need to make a change.

Solution: Feature Linked Annotation allows you to edit the placement of your text and whenever the data is changed in the attribute table, it will automatically be changed in the annotation

Data used: Alaska, Tagged_Birds

License: Standard. Feature-linked annotation is read-only in ArcGIS for Desktop Basic

---



1 Bring Tagged_Birds into your project and open the properties and go the **Labels** tab, check box at the top **Label Features in This Layer**

2 Open the expression panel

Enter: **[Species] & " " & [OBJECTID]** : In the lower window. Make sure the parser is set to **VBScript** and verify what you entered.

Close that window and open the Placement Properties window

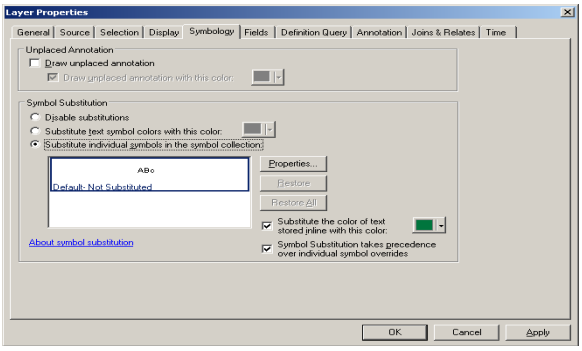
Change the angle to 45 and check **'Place overlapping labels'**

Close the windows

Right click **Tagged_Birds** in the table of contents and select **'Convert Labels to Annotation'**

3 If you click **In the Map** your Anno will not be feature linked, it must be stored in a geodatabase. Note the reference scale, make sure this is the scale you will want the annotation to be viewed at. Click **Convert** and your Anno will be added.

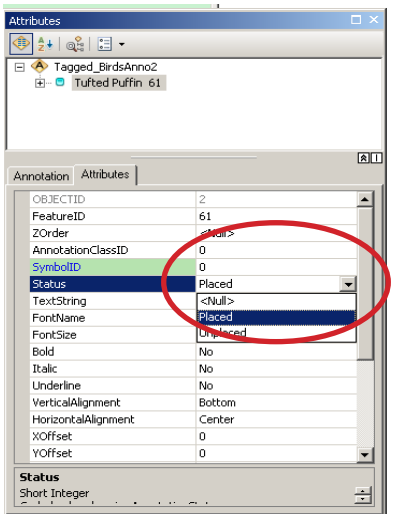
Next to the black arrow in the **Editor** tool bar
Is a black arrow with a blue A, use this to edit your Anno.



Try adding data and changing the attributes of other points.

4 Open the properties to the Anno layer, how can you change the appearance of your Anno?

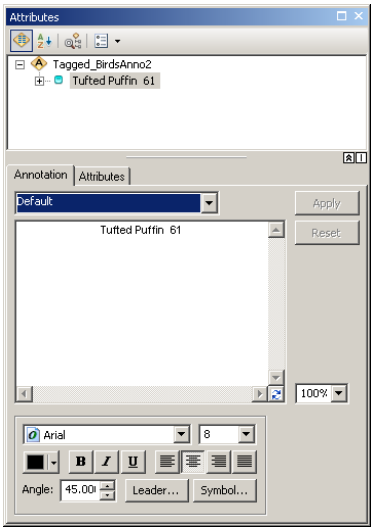
Select the **Substitute individual symbols in the symbol collection** button Click on Properties and change the symbology, This will change the symbology for all of the anno

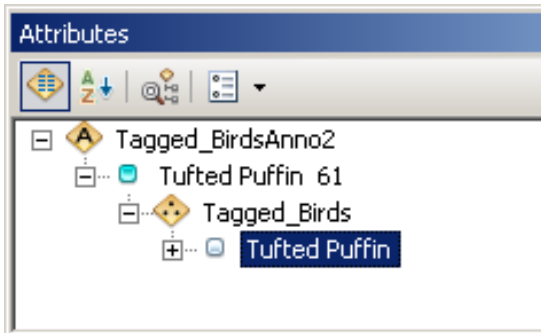


5 Open the attributes for the **Tagged_BirdsAnno** by selecting on one of the annotations and right clicking on it and selecting **Attributes**.

There is a field that will let you turn the anno on or off.

In the annotation window You can change the symbology individually





6 Sometimes an area might be very congested with points and annotation and you lose track of what goes where. With a selected Anno, in the attribute window, if you click on the + next to the name it will open up to show which point it is connected to. Click on the points name and it will flash the point in the data view.

Save your project and close it

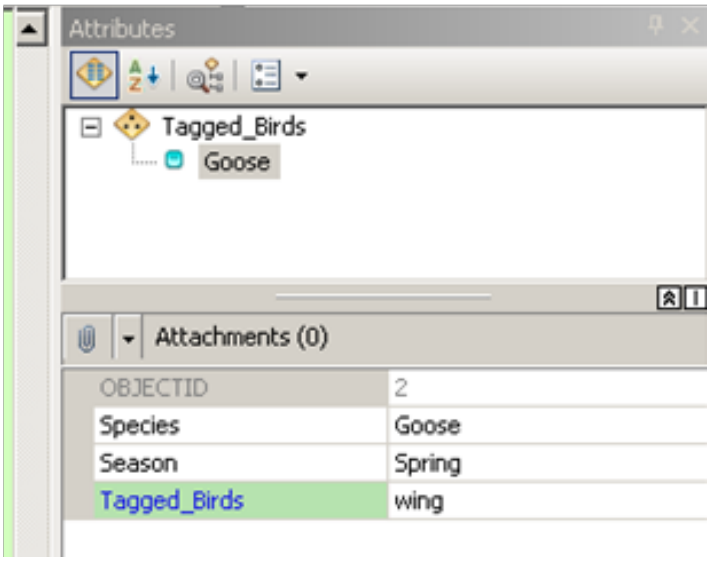
Topic: Attachments

Problem: A biologist is monitoring certain tagged birds around Kachemak Bay, she is also maintaining a point feature class of each location. The attribute table is good for maintaining basic info like name and tag number, but she want other documents and photos to be associated with it.

Solution: Enabling attachments on a feature class in a feature data set allows you to attach any kind of document or image to a feature. These attachments can then be stored in the database and viewed in arcmap.

Data used: Alaska, Tagged_Birds,

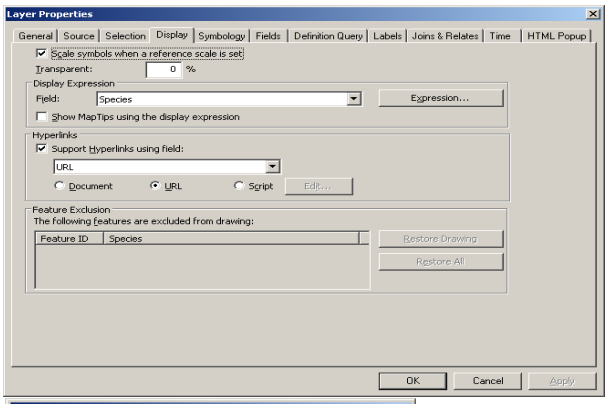
License: Standard



1 Open ArcCatalog and navigate to your database, Right-click the feature class **Tagged_Birds**

Select **Manage > Create Attachments**. Notice two more items where added to the data set, one is a table, the other is a relationship

2 Close ArcCatalog and reopen the previous project. Start an editing session with **Tagged_Birds** and make sure the Attribute window is open. Select on of the bird points and notice that there is a paperclip symbol in the attachment window.



3 Select one of your points

Click on the paperclip in the attributes window to open the Attachments window.

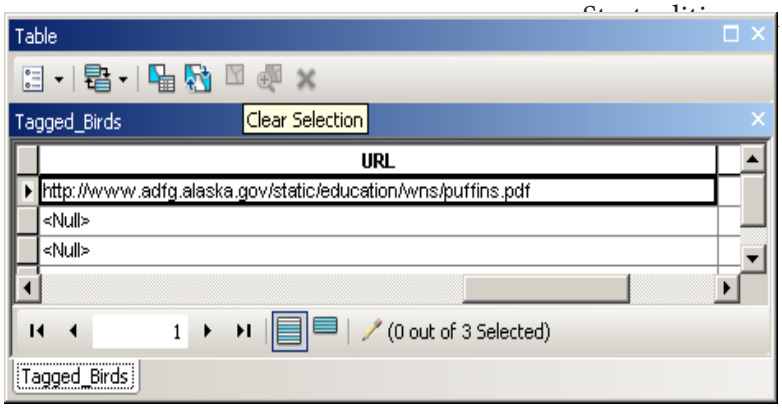
Click **Add** and navigate to the 'Pics' folder in your student folder and find an image for the species you selected.

Add the PDF Document to a Tufted Puffin feature using the same method Save edits and stop editing

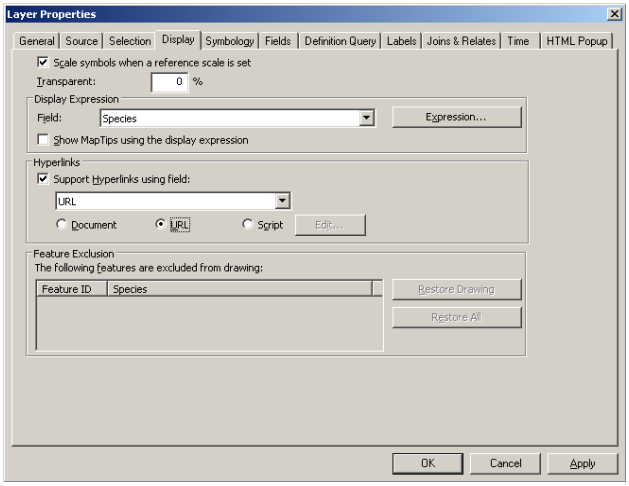
4 Open the Properties for Tagged_Birds and open the HTML popup tab; make sure the box in the top left corner is checked. Close the window and go back in the data view, select the HTML popup tool 

And use it to select the Tufted Puffin point

5 Open the attribute table and add another field called 'URL', Text size = 150

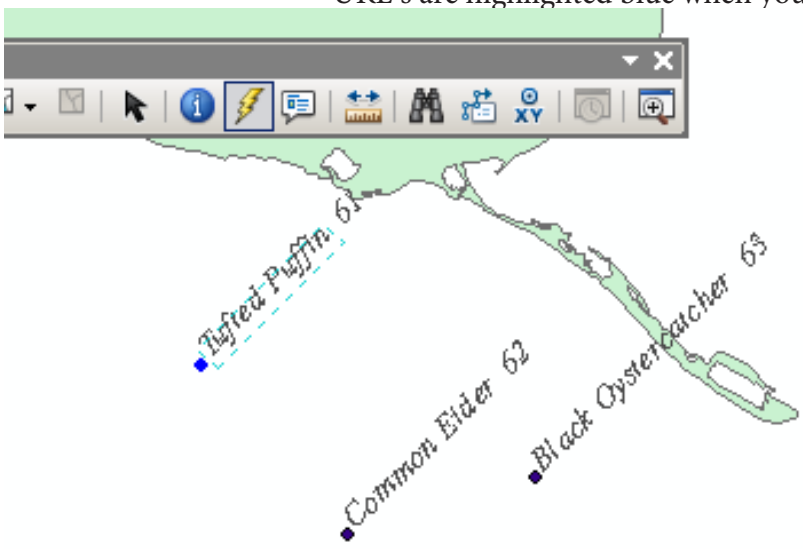


the same feature class, Use the 'Url.txt' file in the Docs/Pics folder to add the URLs to your table



6 Open the properties for 'Tagged_Birds' Open the display tab and check on the **Support Hyperlinks** using field:' and select URL; and check URL below that. Close properties

7 Now you can use the lightning bolt button to select the features that have URLs set and the links will automatically open if you have internet access. The points that have URL's are highlighted blue when you click on the



Topic: Representations

Problem: Sometimes a map may be spatially accurate but look odd at certain scales. Another problem is when a group of people are using the same data but are symbolizing the data differently when you want uniformity.

Solution: Cartographic representations can solve both of these problems. Representations can manipulate the data graphically while maintaining its spatial integrity

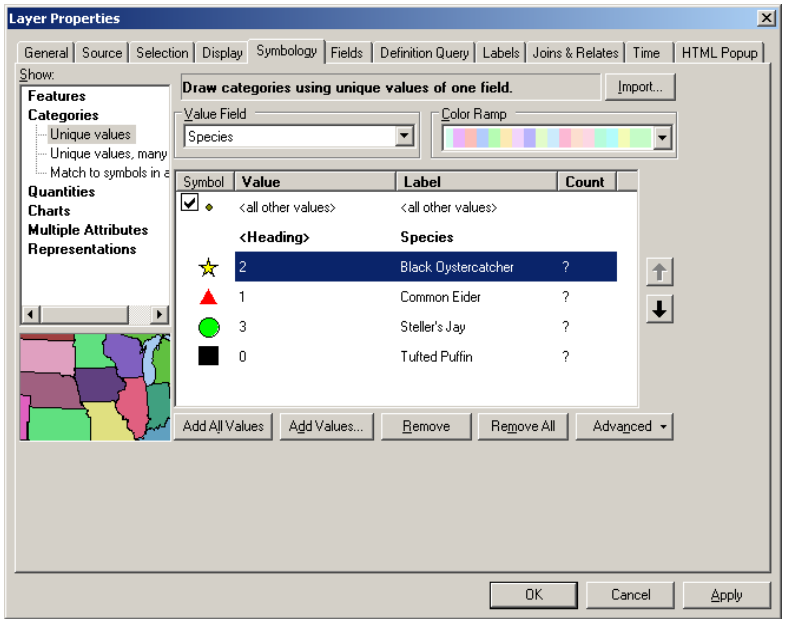
Data used: Alaska, Tagged_ Birds, Hwy_Route, AWC_Clip

License: Standard

---

1 Open the properties for 'Tagged_Birds' and open the Symbology Tab.

In the left window you should have 'Unique Values' selected under 'Categories'



Select 'Species' under 'Value Field' and 'Add All Values'. Give each species a unique icon

Close properties

2 Right click **Tagged_Birds**

Select **Convert Symbology to Representation** Keep defaults and click **OK**. Remove the original layer, notice the change in properties

3 Start an editing session and using the black arrow on the **Representation tool bar** move one of your points

Now select the same point with the arrow on the Editor tool bar and notice where the center 'X' is. Only the symbology has changed.

Try the rotate, resize, and move tools.

Save and stop editing

Convert HWY_Route to representations and try out some tools on that. Any Problems? Save and close your project.

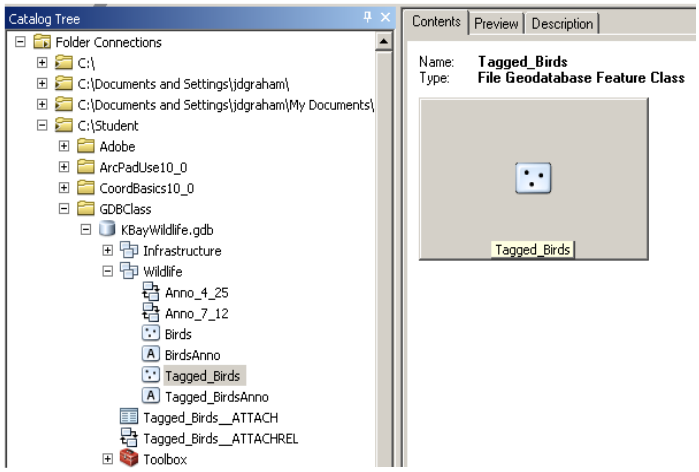
Topic: Metadata

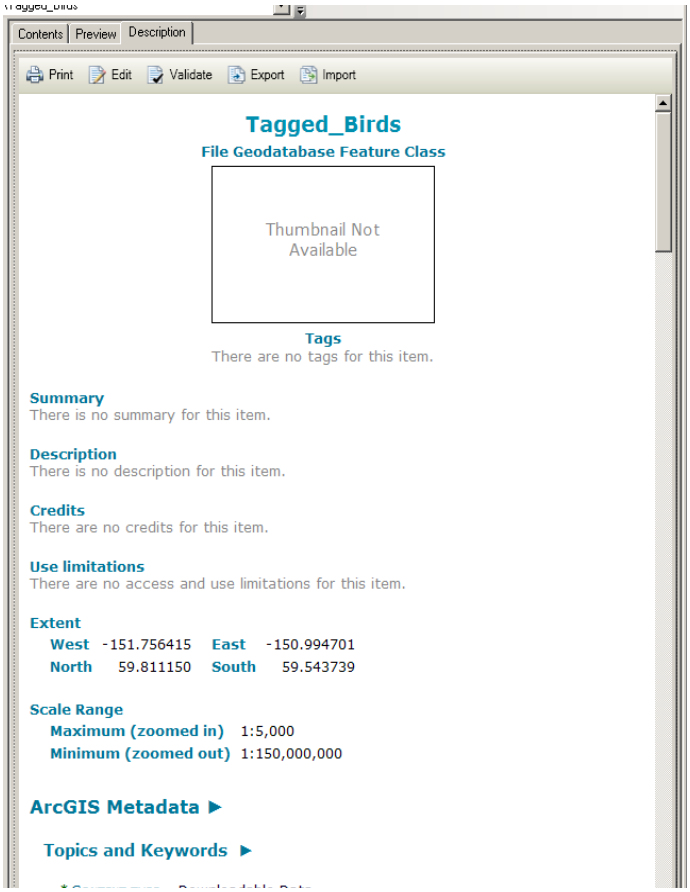
Problem: When sharing data, a lot of question can come up about the source, it accuracy, its timeliness or the meaning of the cryptic attributes. There also may be constraints or limited uses for the data. Some organization consider data with out metadata to be invalid and unusable

Solution: By filling out the metadata you are complying with standard practices and making your data more useful and robust.

Data used: Tagged_Birds

License: Basic





1 Open ArcCatalog and navigate to your KBayWildlife.gdb

Left click to select the **Tagged_Bird** feature class in the catalog tree, then select the description tab

Notice what is annotated and what is not. Some things like projection are automatically added.

2 Click on the 'Edit' button and explore the inputs

Add one of the bird images for the thumbnail

Give a brief description in the **Description(Abstract)** box

The other important one to fill out is the **Contacts,** add in your information

3 To change the Metadata format click on **Customize/ ArcCatalog** options and find the **Metadata** tab

there you will find a drop down list of optional formats

Save and exit

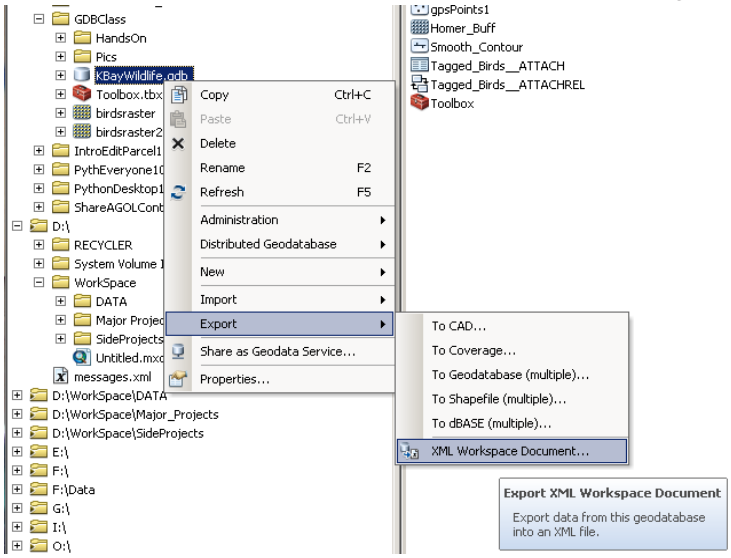
Topic: XML importing exporting
Problem: After putting all this work into your data, eventually you will want to share it. E-mailing shapefiles was always cumbersome with all the associated files.

Solution: Now we can convert our entire geodatabase into one .XML file that can be easily transferred to and from other computers.

Data used: KBayWildlife.gdb

License: Standard

We have already experienced importing .XML when we first created an empty .gdb and imported the class data. Now we will try exporting.



1 Right-Click on your GDB. Select 'Export' > 'XML workspace'
Save it to the same folder but change the name
Keep all the defaults and include all layers

How big is the new .XML file?

2 Create a new file geodatabase and name it 'test'

Right-click on 'test' and import the .XML file you just created

Try importing/exporting datasets or feature classes

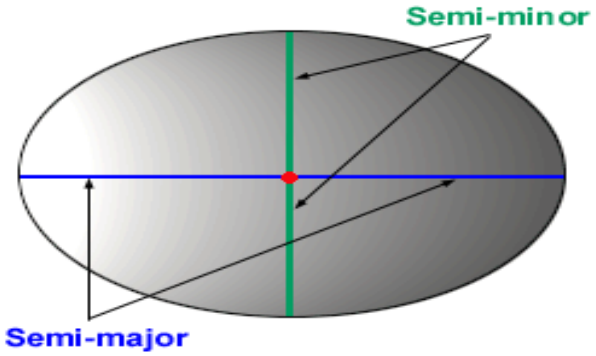
Topic: Projections and coordinate systems
Problem: The earth is not flat

Solution: Understand the shape of the earth and how to relate two dimensional data in a three dimensional scenario

Data used:

License: Basic

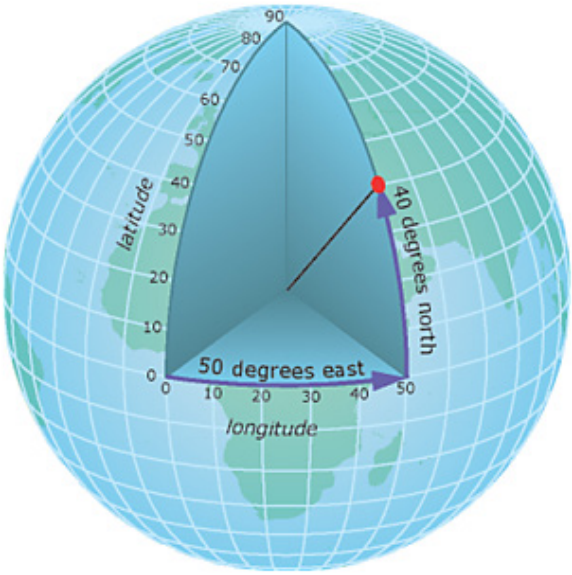
The earth is an oblate ellipsoid but not that extreme. It is actually so slight that we can call it a spheroid



Because of surface irregularities, the earth can not be adequately modeled by a single spheroid. Instead, different spheroids are applied to different parts of the world to achieve local accuracy.

Locations on the earth's surface are measured with lines of latitude and longitude. The mesh of intersecting lines of latitude and longitude is called a graticule.

Latitude and longitude are measured in degrees, minutes, and seconds.



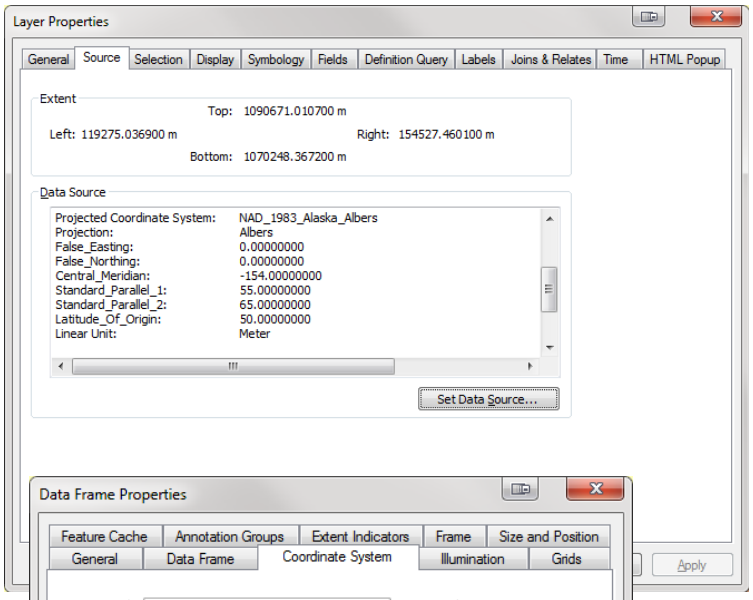
Decimal degrees (DD) are similar to degrees/minutes/seconds (DMS) except that minutes and seconds are expressed as decimal values.

A global or spherical coordinate system such as latitude–longitude. These are often referred to as geographic coordinate systems.

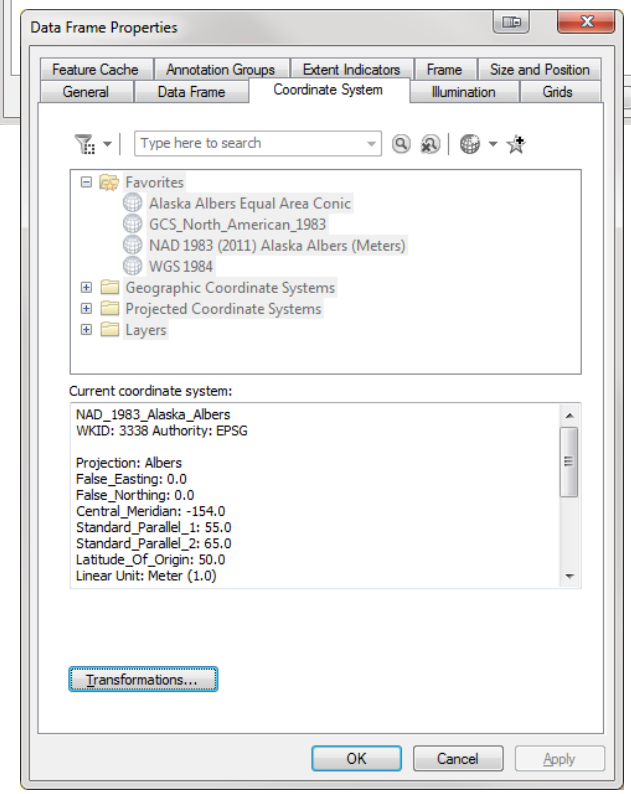
While a spheroid approximates the shape of the earth, a datum defines the position of the spheroid relative to the center of the earth. A datum provides a frame of reference for measuring locations on the surface of the earth. It defines the origin and orientation of latitude and longitude lines.

A projected coordinate system based on a map projection such as transverse Mercator, Albers equal area, or Robinson, all of which (along with numerous other map projection models) provide various mechanisms to project maps of the earth's spherical surface onto a two-dimensional Cartesian coordinate plane. Projected coordinate systems are sometimes referred to as map projections.

A projected coordinate system is defined on a flat, two-dimensional surface. Unlike a geographic coordinate system, a projected coordinate system has constant lengths, angles, and areas across the two dimensions. A projected coordinate system is always based on a geographic coordinate system that is based on a sphere or spheroid.

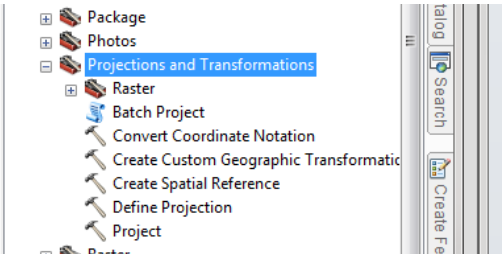


Layer's projection properties



Data Frame projection properties

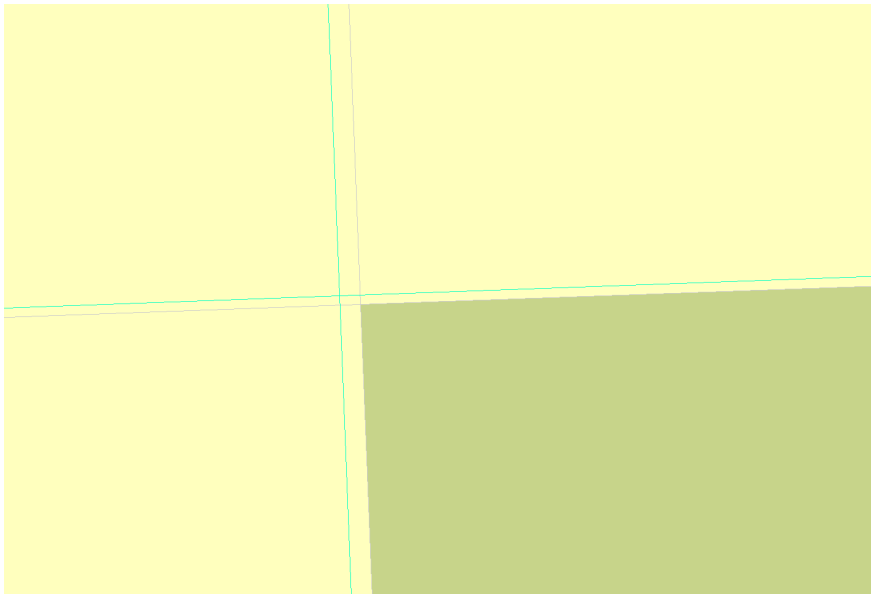
Geoprocessing tools for working with projections



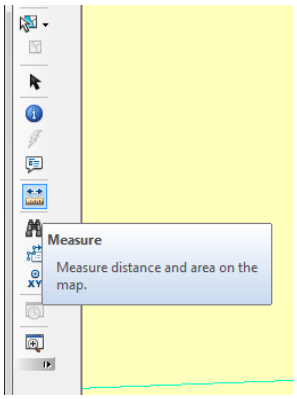
Topic: Reprojecting data
Problem: ArcMap will, in general reproject layers on the fly when the data frames projection is set. However, to avoid small errors it is better to have all the data in the same projection as the data frame.
Solution: Reproject data

Data used: from \GISCLass\Data\Editing\Editing.gdb, add Parcels.
and \GISCLass\Data add KenaiB_SP.shp
License:
Basic

Start a new project and add the two layers. Make the top layers symbology hollow and add color to the other Parcels. Zoom in until you can see the difference



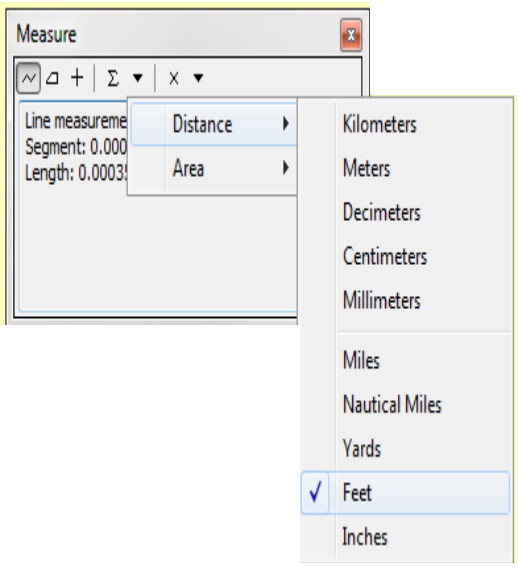
Use the measure tool to find the difference

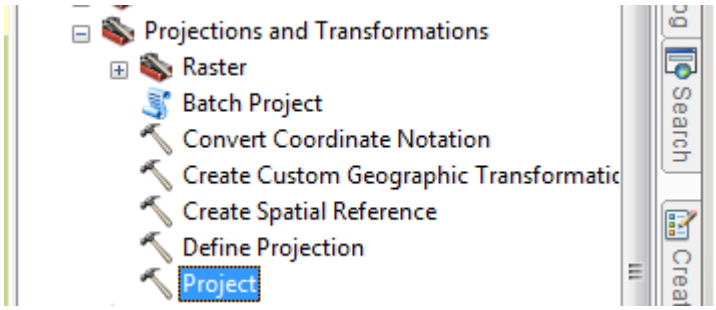


Click on the Σ symbol(sigma) to change the units.

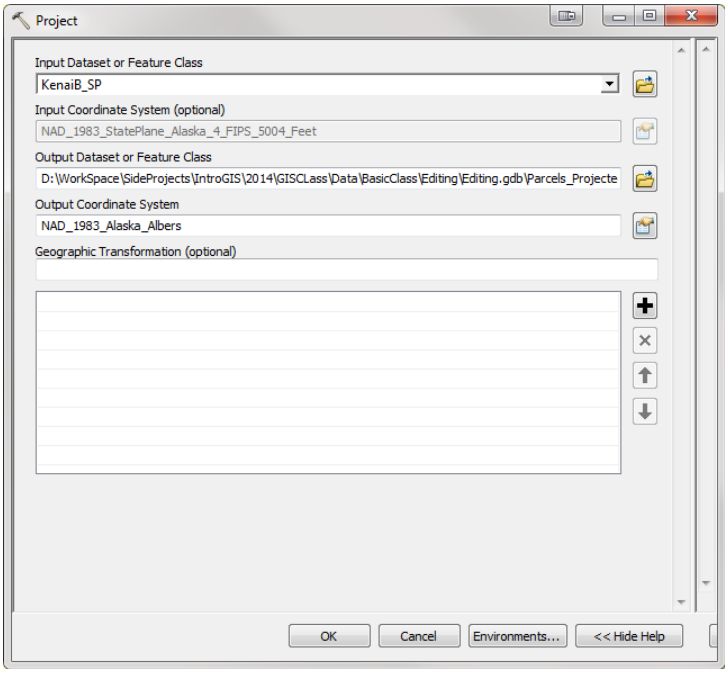
Look at the layer properties for the two layers and find out what projections they are in.

We want to convert them to be the same as the data frame which should be Alaskan Albers





In the ArcToolbox window, find the Projections and Transformations tool box,in the data management tool set select Project



Follow the wizard, add in the layer to be projected, save it to your class DB, and change the output coordinate system to Alaskan Albers

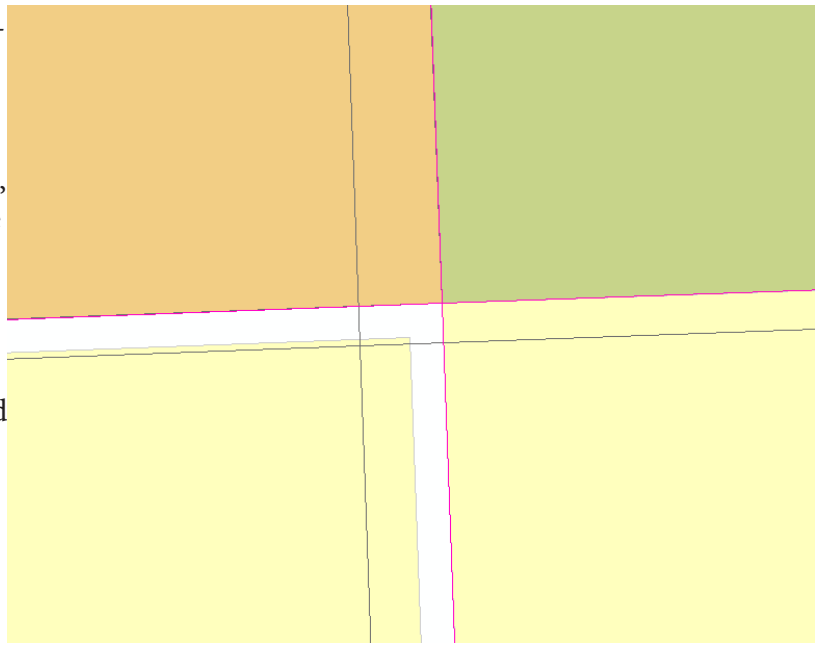
For this one we do not need a transformation, this is important when you are changing datums such as NAD 27 to NAD 83, the new layer should come into your project automatically. Does it line up?

From: \GISCLass\Data\Editing\Editing.gdb add: **Protracted_Townships** and **Kenai_Townships** to the project and zoom to an area to see the difference. Examine there projections. What is wrong with this? any guesses ?

Difference is between protracted townships and surveyed townships, not a projection issue. In general, data that comes from the state conforms to the protracted diagram developed by USGS, this is the statewide township and range system.

However in some areas such as the Matsu Valley and Kenai Peninsula mistakes were made and a new surveyed grid were made for these areas.

Know your data.



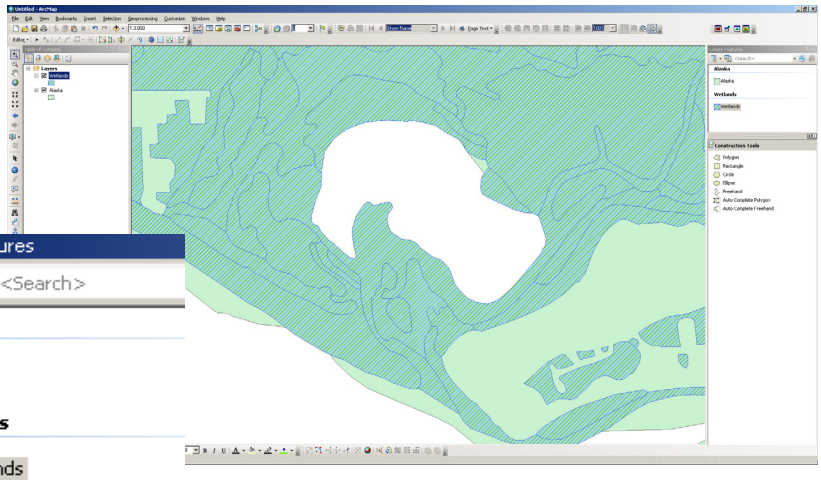
Topic: Editing
Problem: Creating points, lines and polygons and altering existing layers

Solution:

Data used: \KBayWildlife.gdb\Infrastructure and OtherData
Alaska, Wetlands, AWC_Clip, AWC_SC_points
License: Basic

---

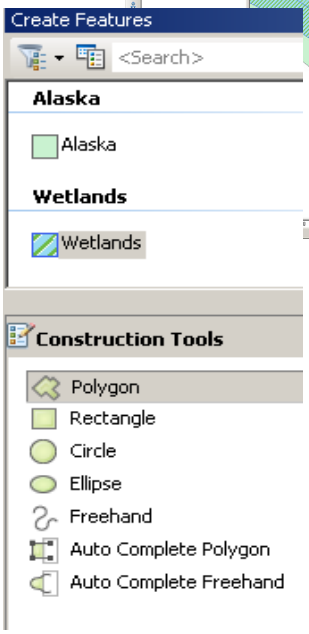
1 Start a new Arcmap Project with the feature classes: Alaska, **Wetlands, AWC_Clip, and AWC_SC_points** from the class geodatabase Zoom into the wetlands near KBRR, create a bookmark for this area



Start an editing session on **Wetlands**

In the **Create Feature Window** Click on **Wetlands** and on polygon in the **Construction Tools** window

Click on a vertex in side the hole in the **Wetlands** layer.



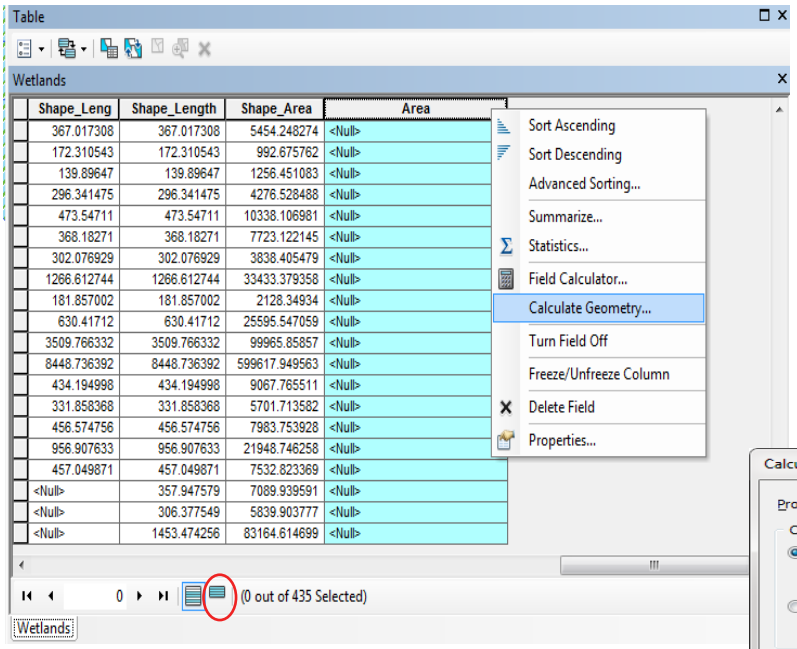
now click on the **Trace** tool

Click on the same vertex and then trace around the hole and double click on the first vertex to finish the polygon. Save and stop editing

Topic: Editing Attributes
Problem:

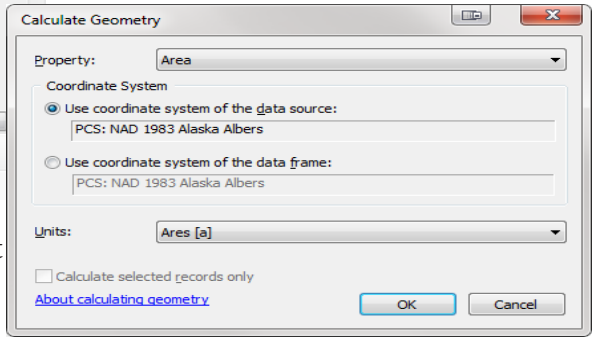
Solution:

Data used: \KBayWildlife.gdb\OtherData Wetlands
and \KBayWildlife.gdb\SpeciesData AWC_clip and AWC_SC_points
License: Basic



Open the table by right clicking on the layer in the TOC and select **Open Attribute table**. Add a new field named **Area**, type = **Double**. Select the field **Area**, right click and select **Calculate geometry**.

Select **Area** and **Acres(us)** and click ok

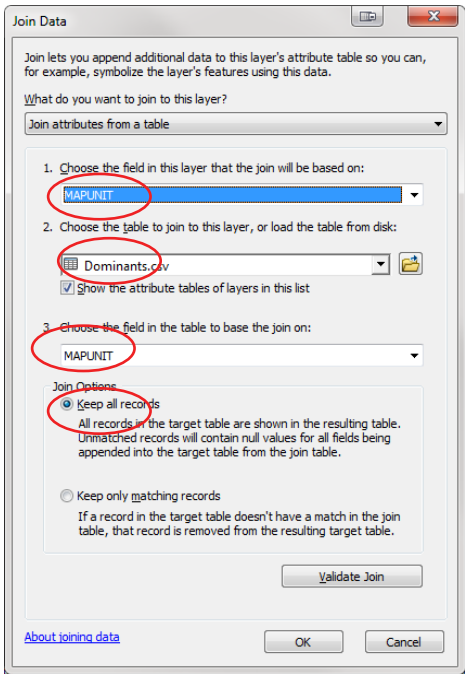


Start an editing session on Wetlands. From the table, select by attribute **ECOSYSTEM = 'Tidel'**

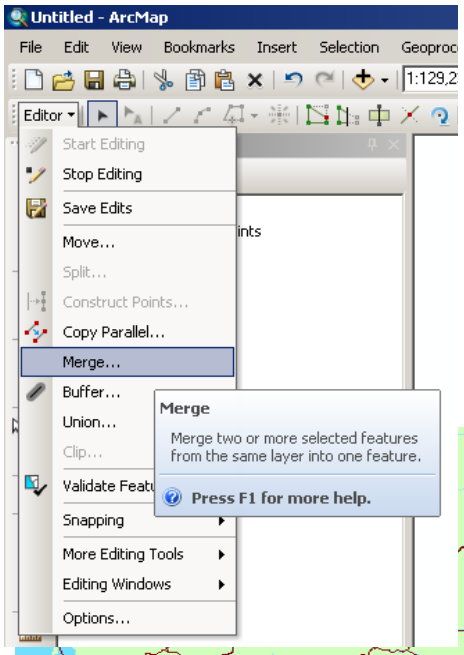
Us e the button on the bottom to see only the selected records. Highlight the ECOSYSTEM Field; from the table menu select the **Find and Replace**, Replace **Tidel** with **Tidal.**

Add the table GISCLass\Data\Editing **Dominants** to ArcMap. You can use either the .csv or the .xlsx.

From the table menu select Join and Relates, make a join using the MAPUNIT field, choose to keep all records and select OK. View the join in the table



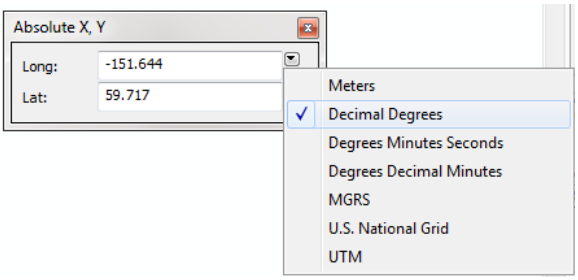
1 Bring in and start an editing session on **AWC_clip** and **AWC_SC_points** , ensure that they are the only selectable layers.



2 Select the arcs for Anchor River and notice it is in three arcs, make them one arc by using the merge tool in the Editor tool bar. Select all three segments then merge. You can hold down the shift key to make multiple selections or use the select by attributes tool. Right Click and select **Edit Vertices**, right click again and select **Sketch Properties** and find how many parts there are.

3 Zoom to the break
Connect the segements
then check the proporties
again

4 Add a Point to the Mouth of Twitter Creek
to add a point at exactly -151.644 59.717 : With the point layer selected from the Create feature window right click any where and select Absolute XY

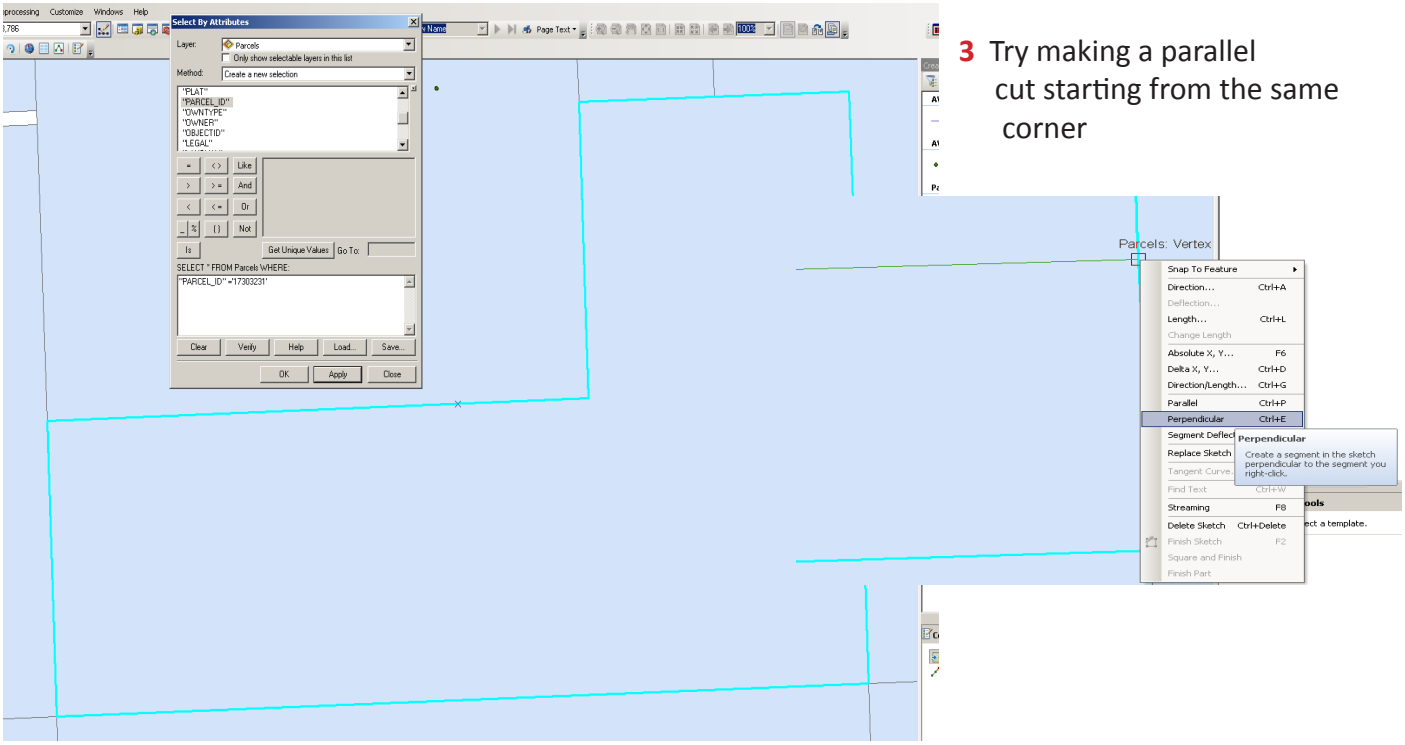


Try again with a more precis point -151.64395 59.717448
To increase the precision in the status bar, open the ArcMap options under Customize and select the Data View tab. Under Coordinate Display In Status Bar, change the decimal places to 6

Save and stop editing

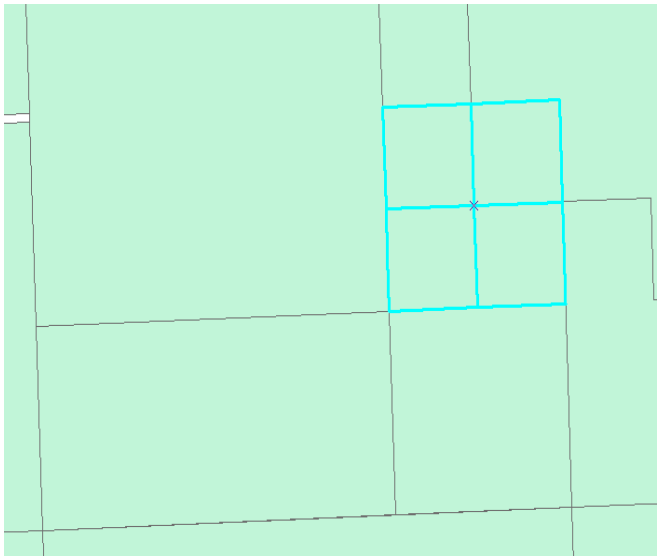
1 Bring in the **Parcel** layer and start an editing session. Find the parcel with the ID '17303231'

2 Find the **Cut Polygon** tool  on the Editor Tool bar and start the cut on the inside corner then place the curser over the right edge of the polygon and right click > select perpendicular, and finish the cut by clicking outside of the parcel. That makes a perfect perpendicular cut.



3 Try making a parallel cut starting from the same corner

Select just the new square parcel and divide it into four peices using the Cut Tool and the Mid Point tool



Topic: Georeferencing /COGO
Problem: Sometimes the data we need is not digital. In this case we have a scanned image of a survey.

Solution: Georeferencing allows us to bring in other imagery or raster data that is not spatially oriented and apply spatial reference to it. We will then use COGO to subdivide a parcel to match the survey

Data used: From \KBayWildlife.gdb\OtherData: Sections, Parcels and from the Docs folder: Survey.jpg

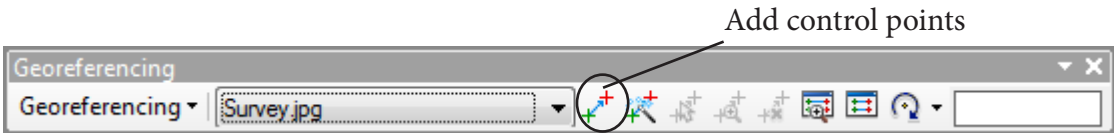
License: Basic/ Standard

---

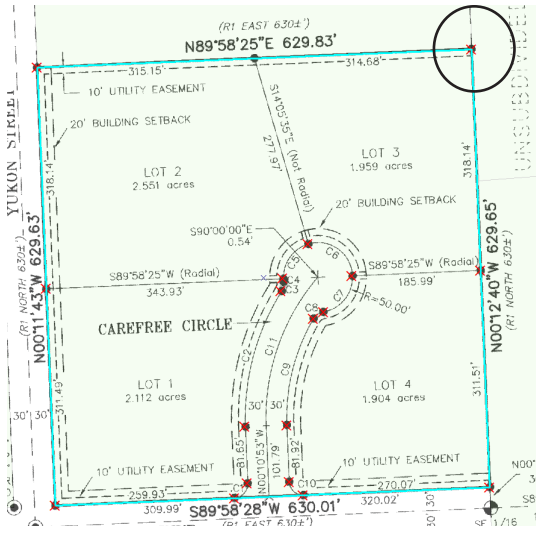
1 Add data 'Sections' and 'Parcels', from the 'doc's' folder bring in the 'Survey.jpg' (say OK when it asks about a missing spatial reference)
Open the properties for 'Section' and make it hollow with red outline.

Zoom to Township S 5s 12w section 28 and locate parcel,
locate Parcel ID '17239113' and zoom into it

2 Open the **Georeferencing** tool bar and pull down the georeferencing menu and select **Fit to Display** This will put the image close to where we want it.



3 Zoom in as necessary to a corner for more accurate points. Use the **Add Control Points** tool to add a point to each corner matching it to the Parcel layer. First click on a surveyd corner of Survey.jpg then to the corresponding corner of the section



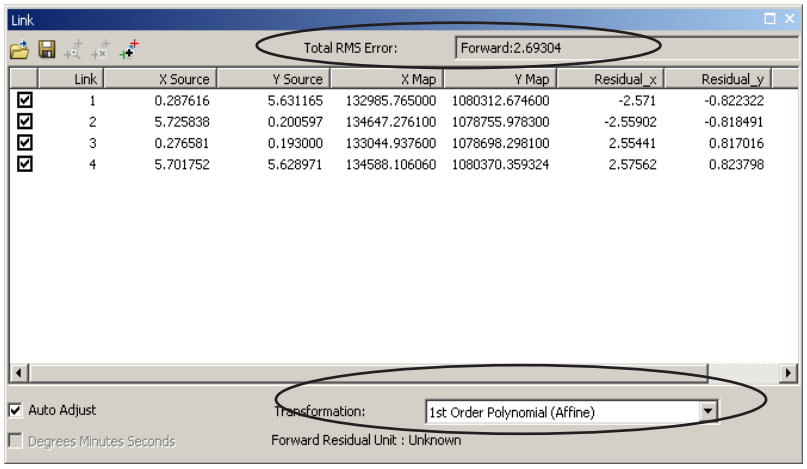
To correct for rotation:
Editing/options/units

North Azimuth
DMS

Direction offset
2-28-57

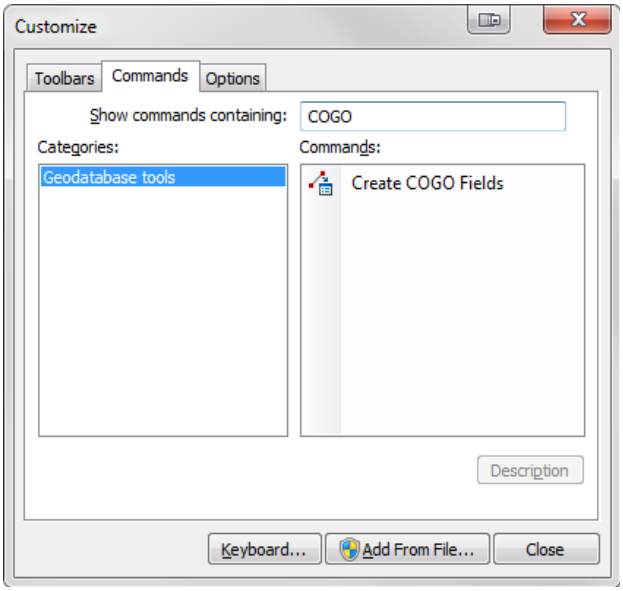
The point should snap to the Parcel. Repeat for the other three corners.

Open the link table and note the RMS error, the lower the better. Points can be removed or turned off to lower the error.



Use the transformation to adjust image. With only four points this will not make much of a difference When you are satisfied with the placement, go back to the georeferencing pull down and select **Update Georeferencing**. This will make the changes semi-permanent.

Open ArcCatalog and open the **COGO.gdb.** If you don't see the Create COGO fields tool, then you need to bring it in with the Customize window. Select the tool and place it on a tool bar in ArcCatalog

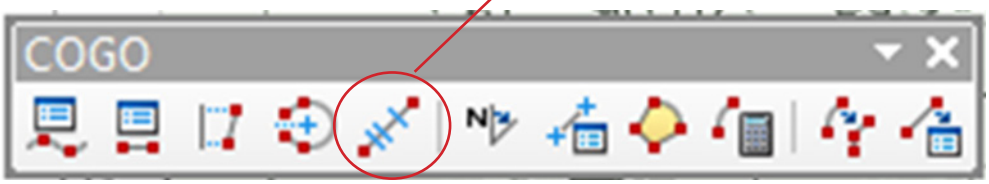


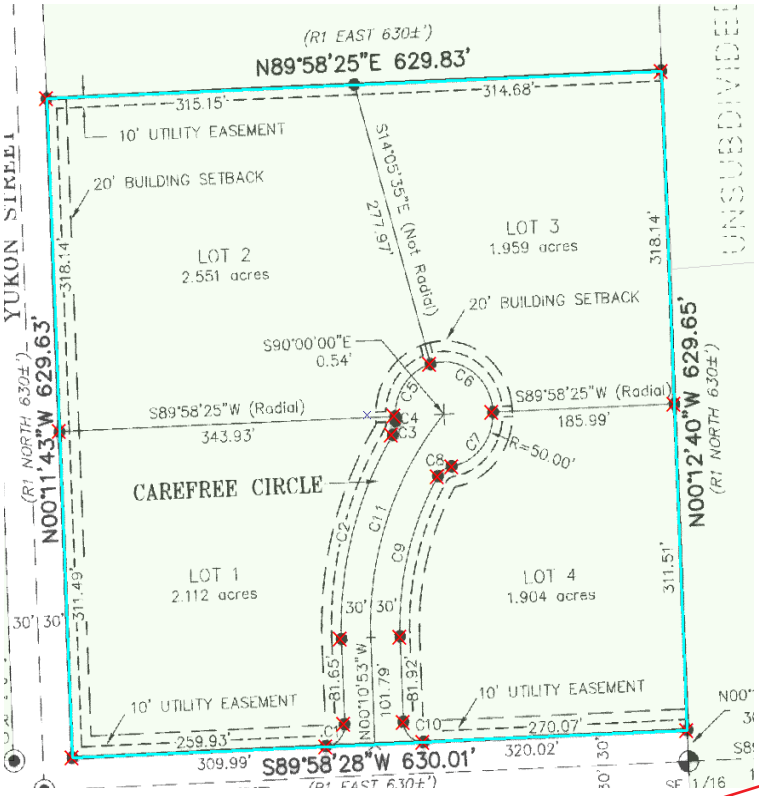
Now select the **CF_COGO_new** layer in the Catalog tree then click on the new Create COGO Fields button, this should create the necessary fields. Now bring this layer into the ArcMap project. Open the COGO tool bar and start an editing session on **CF_COGO_new.**

Draw a line across the top of the entire parcel snapping it to the corners from left to right. With the arc still highlighted, select the proportion tool and enter the distance as indicated by the survey starting with the left side, be sure to enter **ft** after the distance.

Do the same for the other three sides

Proportion Tool



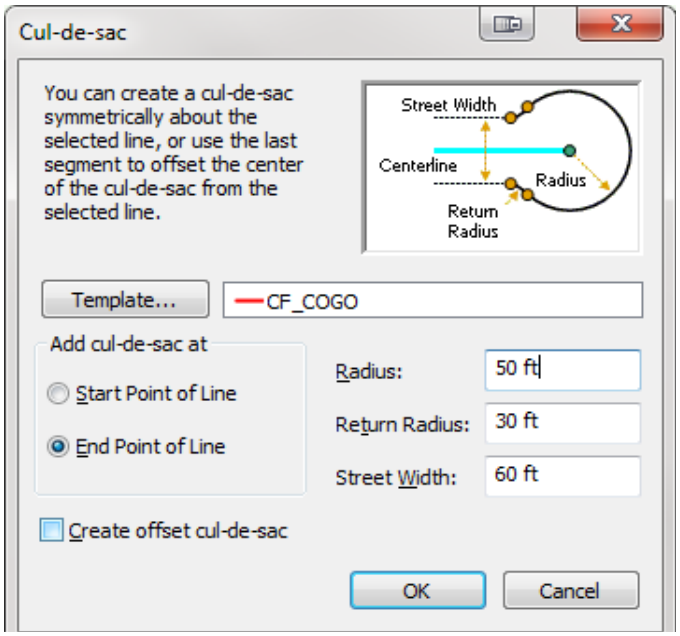


To create the road and cul-de-sac, start by drawing the centerline of the road. Start at the bottom and draw the straight portion first.

Then click on the curve tool and place a point at the lower point then upper point of the road and bend the curve to match the survey

Curve tool

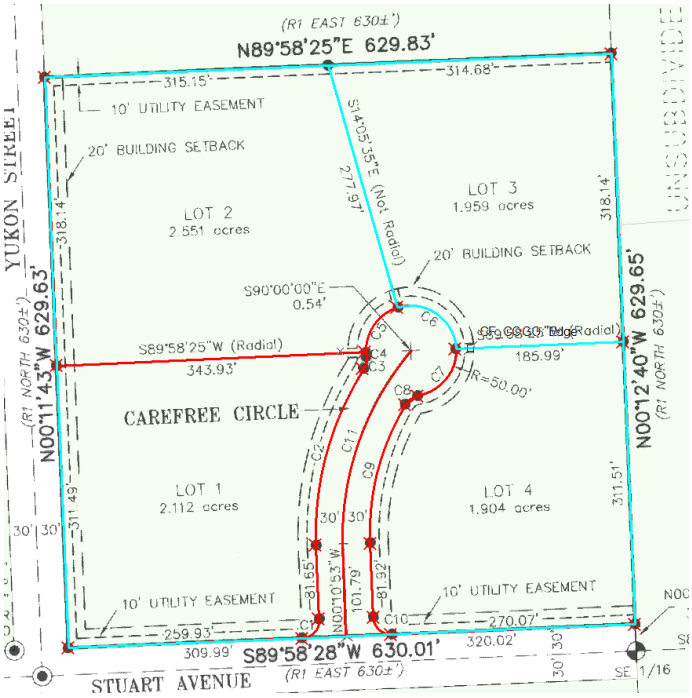
Now, with the centerline highlighted, select the cul-de-sac tool



Enter these parameters and select OK

Try to finish outlining the rest of the parcels and save your edits.

Next we will cut the parcels out using CF_COGO_new as a guide.



Start an editing session on the Parcel layer, make sure it is the only selectable layer.

Select to highlight the parcel, then select the Cut Polygon tool and then select the Trace tool.

Click any where on the arc and trace around one of the parcels, double click on the end to cut the parcel out.

Finish cutting out all the parcels, save and stop editing

Trace tool Cut Polygon tool

Open ArcCatalog and go to the COGO.gdb
Create a new feature class called **Dim**

Change Type **Dimension Features**

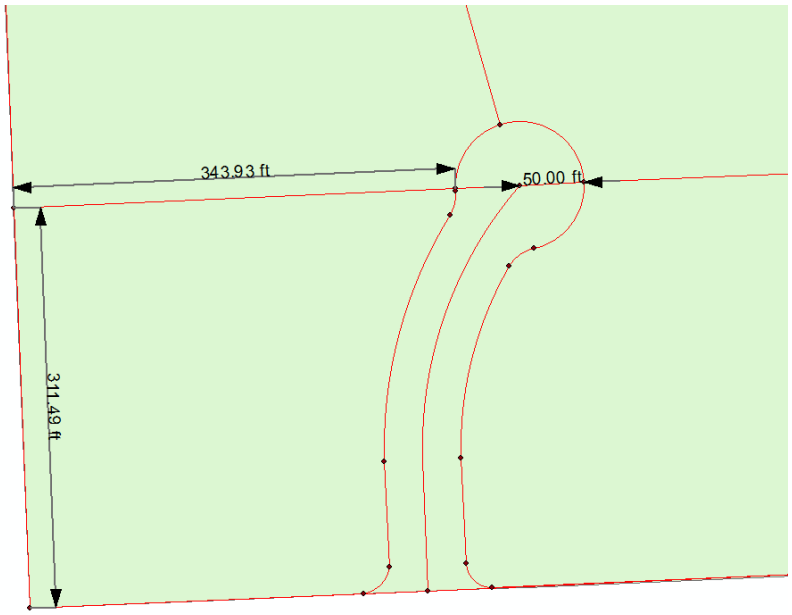
Set Reference Scale to **1000**

Set map units to **meters.** Select **I would like to create my own style** and change the display units to feet and add **ft** to suffix

Use default for everything else

Bring **Dim** into your map and start editing

Turn off the survey drawing and use **Dim** to label a few sides of your COGO drawing



Topic: Environment Settings
Problem: ArcGIS has a default database, but you usually keep your data in other locations, often in several locations.

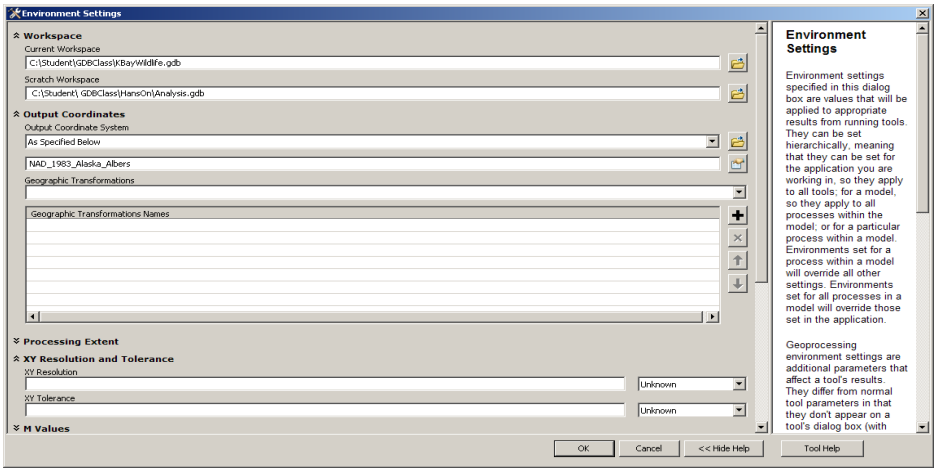
Solution: By using your **Environment Settings** you select where your workspace is, where your scratch workspace, along with many other settings

Data used:

License: Basic

At the beginning of an ArcGIS session, it is important to consider environmental settings that may be used as default values throughout a project. These may include the directory path to your data, standard protocol for displaying types of data, and what tools or extensions will be needed. Analysis properties set at the application level will apply to all tools. However, before we can set Environmental Settings, we need to create a geodatabase to hold our data.

1 Open ArcCatalog. Click on File > Connect to Folder, navigate to the c:\Student\GDBclass directory and click OK.
2 Right-click on HandsOn and select New > File Geodatabase. While this new geodatabase is highlighted in the Contents tab, change the name to Analysis.gdb.
3 Click on Customize> ArcCatalog Options, in the General tab, and make sure the box next to Hide File Extensions is unchecked.
4 Click Geoprocessing > Environments settings
Click Show Help at the lower right of the dialog box and review the information about Environmental Settings at this level. As you can see, settings established at this level will act as application-wide variables throughout subsequent analyses.



Click on Workspace to expand the associated category and enter the following information:
• Current Workspace: C:\Student\GDBClass\KBayWildlife.gdb

When working with raster or other data in ArcGIS, intermediate data sets are often created. Setting the scratch workspace gives the user control over where these files are stored.

• Scratch Workspace:C:\Student\GDBClass\KBayWildlife.gdb or you can create a new .gdb to use here

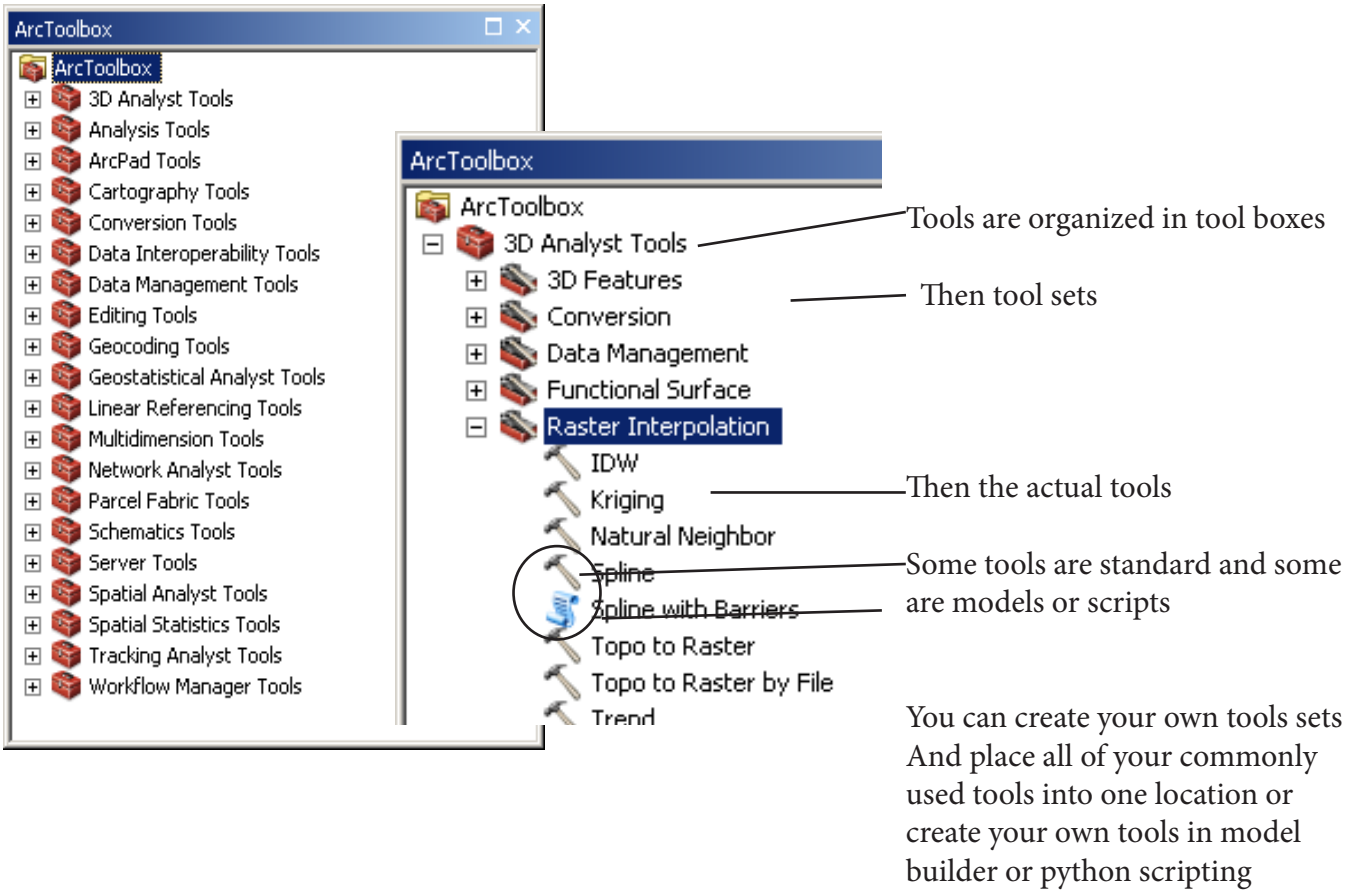
• Output Coordinate System: As Specified Below
• Alaskan Albers Equal Area Conic
5 You can explore the other settings but for now we will leave them as default

Topic: Geoprocessing tools intro
Problem: When you need to do more then create points, lines, and polygons you have geoprocessing tools to do the work for you.

Solution: Geoprocessing tools are used for data analysis, bulk data edits and to improve productivity

Data used:

License:



Topic: Geoprocessing tools - spatial analyst
Problem: Creating a contour layer from a digital elevation model used to denote the linear representation of constant elevation

Solution: Using spatial analyst geoprocessing tools

Data used: Alaska, Homer, NHDFlowline, AWC_Clip, AWC_SCN_Points, Ned_Homersml.tif
License: Basic with Spatial Analyst extension

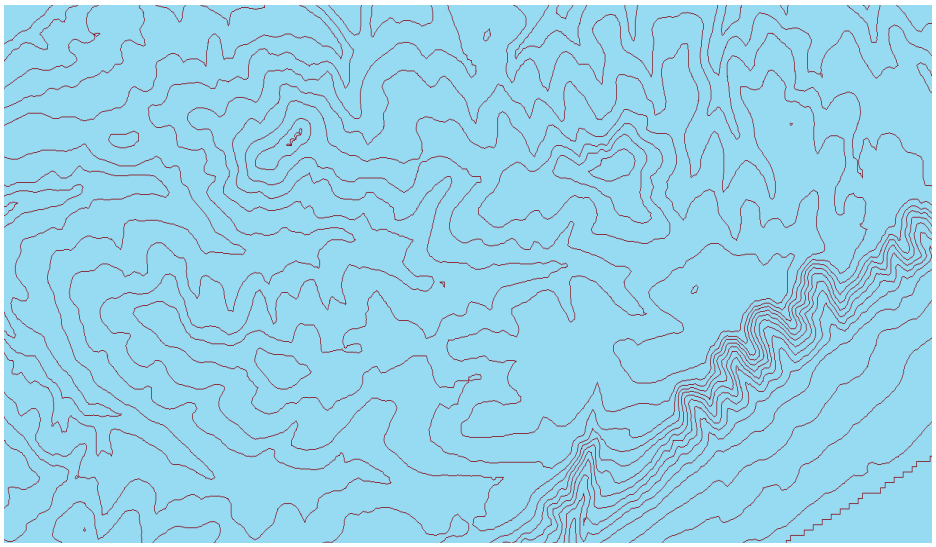
Building Contours

1 Start Arc Map make sure the Spatial analyst extension is on

2 Add data from the **Kachemak.gdb**: **Alaska**, **Homer**, **NHDFlowline**, **AWC_Clip**, **AWC_SCN_Points**, Ned_Homersml.tif

3 Open the toolbox and navigate to Spatial Analyst Tools>Surface>Contour

4 input: **Ned_Homersml.tif**
Output: **C:\Student\GDBClass\HomerContour.shp**
Interval: ***100 ft***.
click **OK**



Topic: Geoprocessing tools
Problem: A biologist wants to locate state wetlands that are un-impacted by infrastructure that can be placed into protection.

Solution: Using a series of tools and methods, we can clip and extract the area of interest from larger data sets

Data used: Roads, Parcels and wetlands

License: Advanced

Finding un-impacted state wetlands

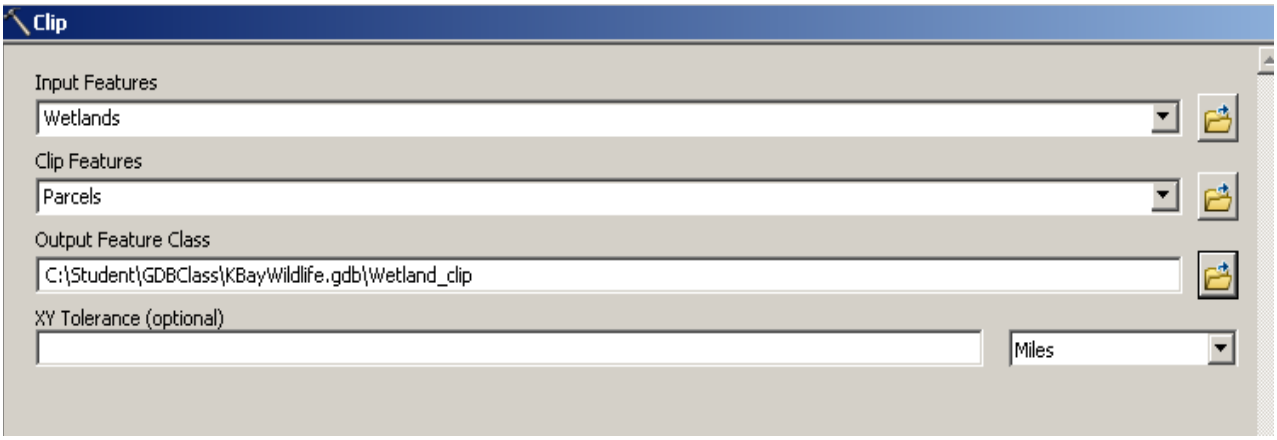
1 Add layers: Roads, Parcels and wetlands

2 Navigate to the buffer tool in the toolbox, **Analysis > Proximity > Buffer**
Buffer the roads layer at 50 ft.
Save as **Road_Buffer**

3 Open the **Parcels** Layer Properties window, go to the Definition Query tab and set a query as "OWNTYPE" = 'STATE'

4 Find the Clip tool, **Analysis > Extract > Clip**

Clip the 'Wetlands' to the 'Parcels' Layer and call it **Wetland_clip**



5 Find the Erase tool, **Analysis > Overlay > Erase**

Input : **Wetland_clip**
Erase: **Roads_Buffer**

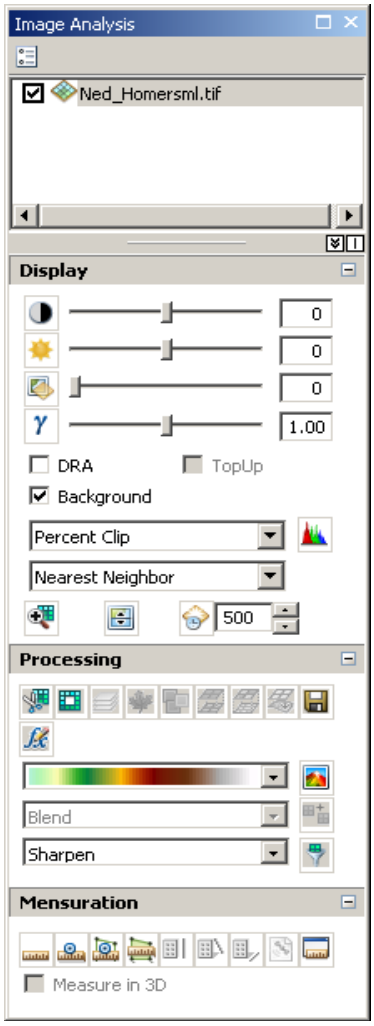
What remains are state owned wetlands unimpacted by roads

Topic: Image Analysis Window
Problem:

Solution: The Image Analysis window supports the analysis and exploitation of image and raster data in ArcMap with a collection of commonly used display capabilities, processes, and measurement tools.

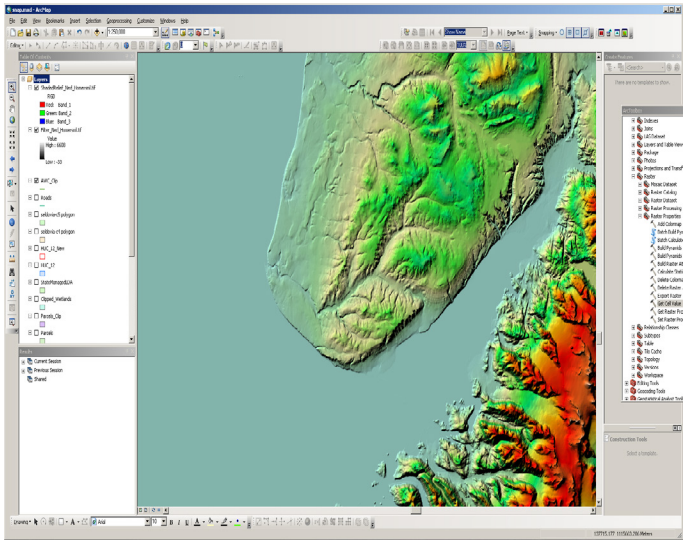
Data used: Ned_homersml.tif, SelVeg

License: Basic with spatial analyst



1 Start ArcMap and add Ned_homersml.tif and open the image analysis window Under **Windows > Image Analysis**

2 Select Ned_homersml in the window
Check the Background box

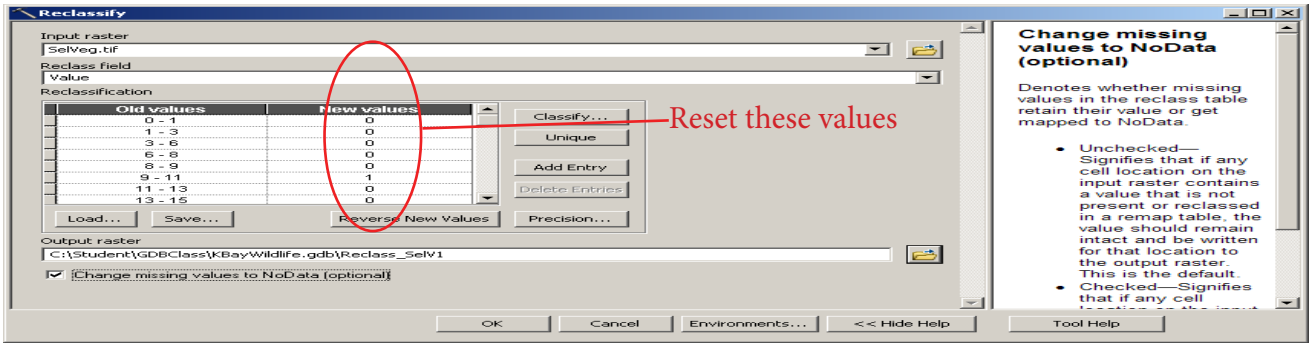


3 In the processing box, change the bottom window to Sharpening 3x3 and click the filter button next to it. Then click the button next to the color ramp.

4 Add SelVeg. Use the Image Analysis to remove the black background and change the color values. Use the identify button and identify some pixel values, Can that be used for analysis?

5 In the toolbox select Spatial Analyst Tools > Reclass > Reclassify
Input:SelVeg Reclass Field: Value and set the values as below

6 In the Docs folder there is a file called **Alaska Land Cover Mapping Projec1. doc** Which value is Lichen Tundra? Under **Conversion Tools** select **From Raster** Click **Raster to Polygon** and convert Lichen to Poly.



Topic: Model Builder

Problem: Sometimes it is necessary to run several geoprocessing tools over multiple sets of data. If you don't do it right the first time you have to start over from the beginning.

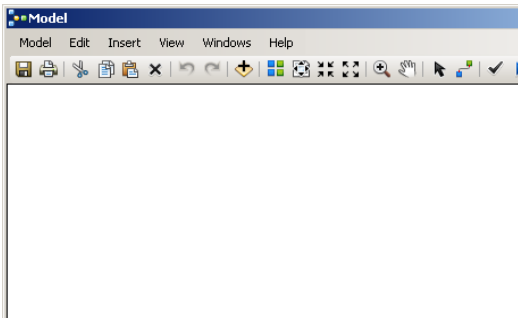
Solution: Using model builder allows you to create a tool that uses many tools and can be reused with other sets of data.

Data used: Tagged_Birds, \Kachemak_Bay_DEM_1239 : kachemak_bay_ak.asc

License: Basic

1 Open ArcCatalog,

3 Open Model Builder,  icon

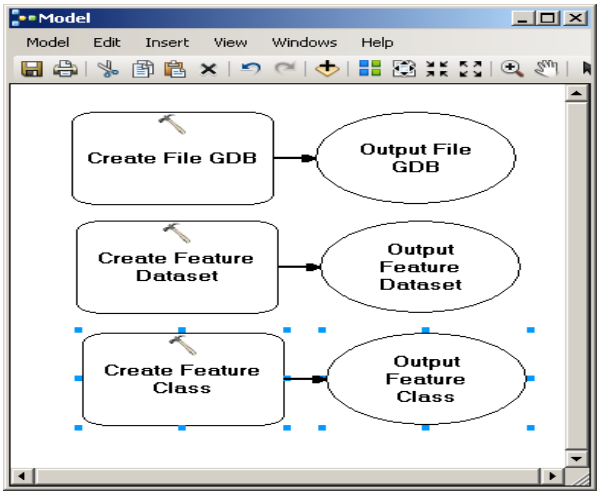


4 In ArcCatalog, open the search Window 

Select 'Tools' above the search box search for **Create File GDB**

Drag and drop the top result into the model builder window.

5 Search for **Create Feature Dataset** and **Create Feature Class** and bring them into model builder, you can just drag and drop it from the search window.



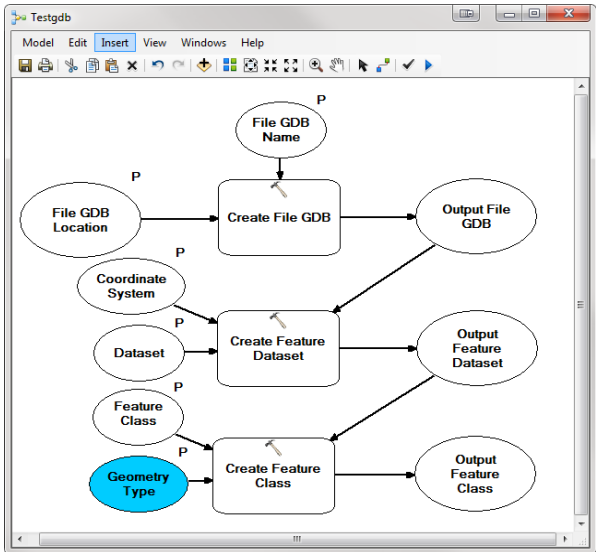
6 Select the **Connect** button  on the tool bar and draw a link between **Output File GDB** and **Create Feature Data Set,** in the same way, connect the **Output Feature Dataset** to the **Create Feature Class**

7 In Model Builder, right click on **Create File Geodatabase** and select **Make Variable > From Parameter**
Bring in both **Location** and **Name**,
Right click on the new balloons and check **Model Parameter**

8 For Feature dataset make variables for **dataset** and **coordinate**

9 For feature class bring in **feature class** and **geometry** type and remember to set all the new balloons as **Model Parameters**

10 Right Click on your class folder and select **New>Toolbox**,



11 In Model Builder select **Model** > **Save** and save it into the new toolbox . Close model builder and open the model in the tool box

Now the model can be used over and over again with different parameters.

The next one requires the Spatial Analyst Extension. In Arc Catalog left click **Customize** > **Extensions** and check Spatial Analysis

---

In this scenario we are adding the depth of water to the location of birds, using your **Tagged_Birds** layer

Close the search window and open the Arctoolbox window

Start a new model builder

1 Look in the tool box **Spatial Analyst** > **Extraction**, and drag and drop **Extract values to Points**.

2 In model builder, bring in the raster **kachemak_bay_ak.asc** and the **Tagged_Birds** points and set as parameters

3 Make sure the output goes into your KBayWildlife.gdb

4 Validate and run

5 In ArcCatalog right click on the model and open the properties/ general tab. Change the name and add a description of the tool

---

This model finds parcels classified as tidal wetlands

Start a new model Builder. Begin by creating a new feature dataset In your GDB called '**Targeted_Lands**'

Next, bring in the select tool from the search window.

In model builder, from the insert tab bring in the 'Parcels' and 'Wetlands' from the 'Otherdata' dataset

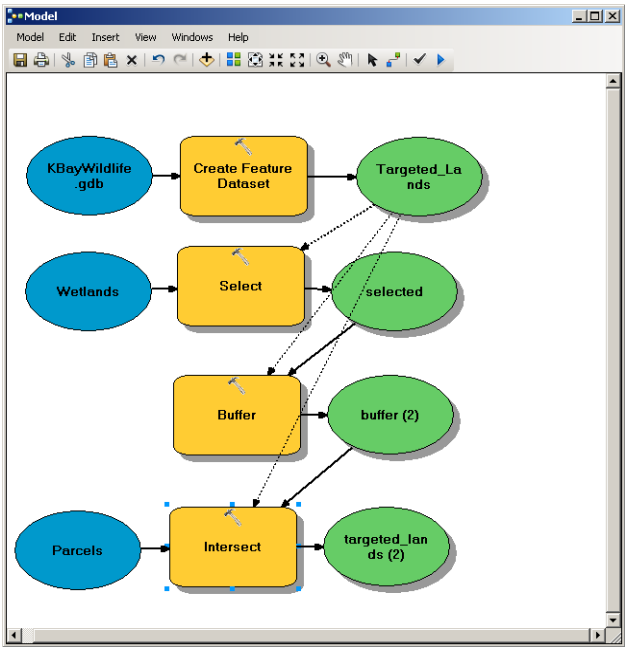
Use Wetlands for the input on the select tool with the expression **"ECOSYSTEM" = 'Tidal'**

Bring in the Buffer Tool, the input is the last previous selected lands, the output goes to the same dataset, buffer 50 ft.

Finally, bring in the Intersect tool, inputs are the parcel layer and the output from the buffer, call the final output **Targeted_Lands/TidalLands**

Right click the 'Select' tool and select properties, go to the preconditions tab and check 'Targeted_Lands'

Validate and Run



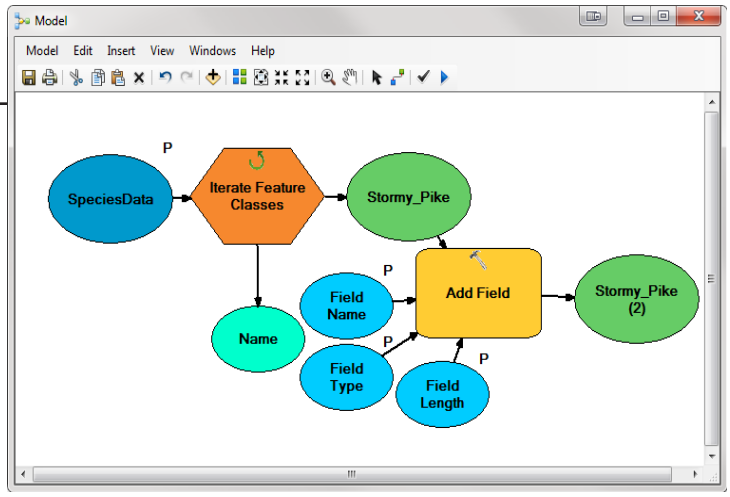
Topic: Model Builder/ Iterations
Problem: You have multiple feature classes that you want to add a new id field to, this could be a time consuming task

Solution: The iteration tool will allow a model to run through multiple files, layers, or rows of a table

Data used: Kachemak.gdb/SpeciesData

License: Basic

Start a new model builder
Select **insert > iterators > Feature classes**

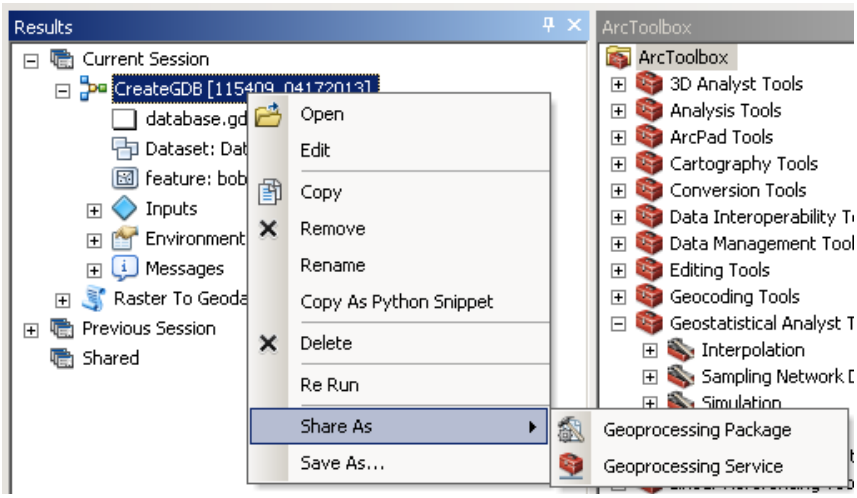


Double click the iterator and select the **SpeciesData** dataset,

Find the '**Add Field**' tool and add it.

Create a new field called '**DOCID**'

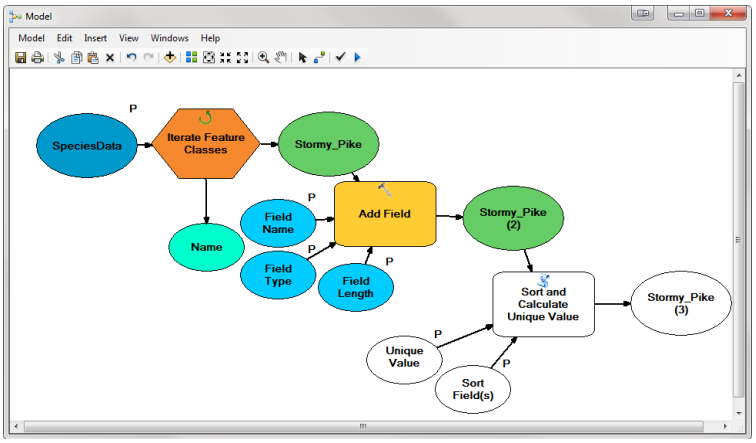
Validate and run



Sharing Models:

Geoprocessing packages will put together your model and all the data involved, that way you can have other people check your results. You create this by right clicking on the model in the results window and select Share as > Geoprocessing package.

To share just the model, save your model into a Tool box . A tool box can be Zipped and e-mailed easily



Try adding the python tool located \GISCLass\Data\MTR\Sort and Calculate Unique Value.pyt

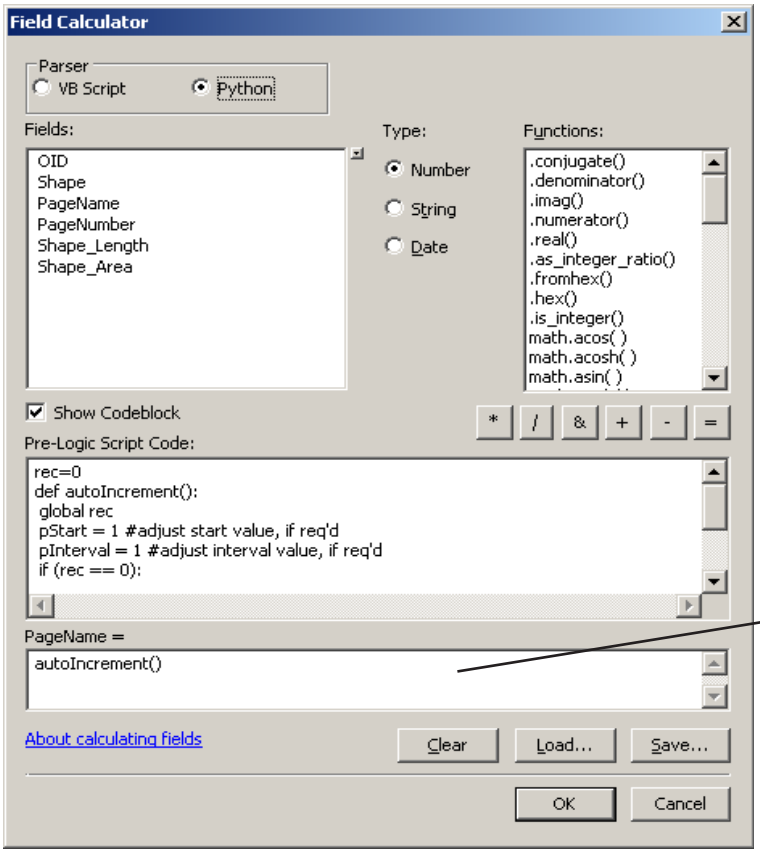
Topic: Python / Python Scripting
Problem: Though there are a lot of geoprocessing tools, sometimes you just need more functionality

Solution: Python scripting lets you utilize all of the geoprocessing tools as well as additional functions, classes, and modules that allow you to create simple or complex workflows quickly and easily

Data used: KBayWildlife.gdb/SpeciesData/CampbellCreek

License: Basic

Python scripting is found all over ArcMap, one way is through the field calculator:
Open ArcMap and bring in the layer **CampbellCreek**. Open the attribute table and then select the **ComID** and open the Field Calculator



Enter the code as seen here, be sure to set the parser at top of page to **Python** first and check the **show code block** box. Run the code and notice the changes, watch the indentation!

```
rec=0
def autoIncrement():
 global rec
 pStart = 1 #adjust start value, if req'd
 pInterval = 1 #adjust interval value, if req'd
 if (rec == 0):
  rec = pStart
 else:
  rec = rec + pInterval
 return rec
```

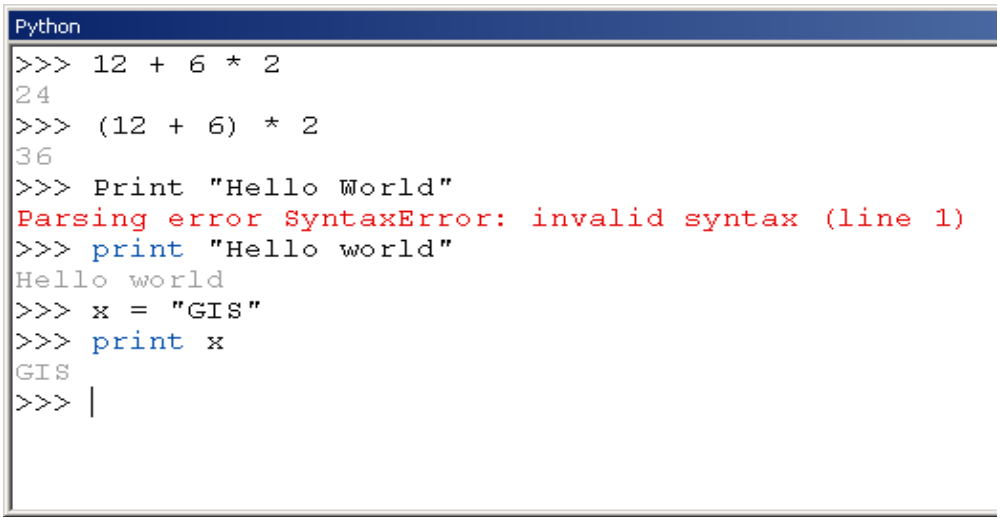
```
Expression = autoIncrement()
```

Use the **Sort** geoprocessing tool on the field if the FID are rearranged differently. Otherwise it follows the original order of input

for pre-written text \GISCLass\Docs\Interval

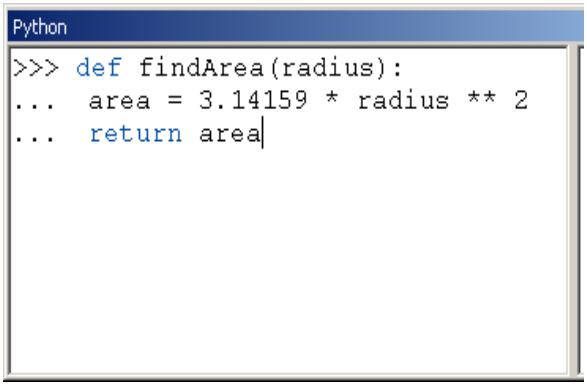
Open the Python Window

Try using these bits of code in the python window



Try a few math functions in the window

Why did the first print not work?

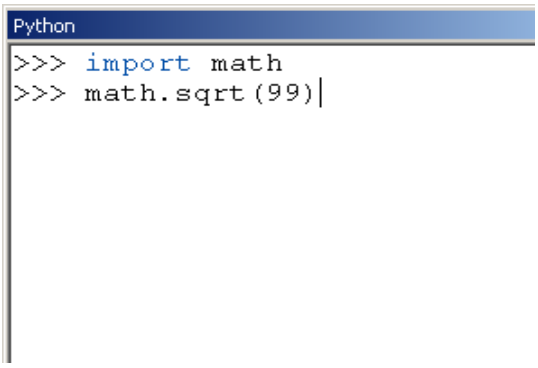


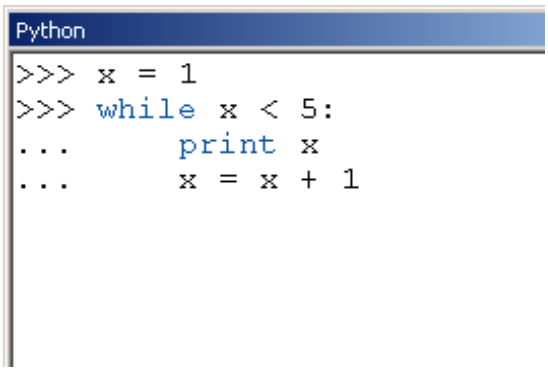
You can define your own variables and pass them through.

How do you use this function?

Try: findArea(80)

Higher math functions can be accessed through the 'math' module





For and while loops are useful for iterating through records

Add the feature class **CambellCreek** to your project . Use the field calculator to set the ComID values to zero. Try this python window version of the enumeration code.

```
Python
>>> import arcpy
... with arcpy.da.UpdateCursor("CampbellCreek", "COMID") as rows:
...     for i, row in enumerate(rows, 1):
...         row[0] = i
...         rows.updateRow(row)
```

Next we use a tool. To use any tool you must first import the arcpy module. Setting the environment workspace helps with the auto-completion.

```
Python
>>> import arcpy
>>> arcpy.env.workspace = "c:\Student\GDBClass\Kachemak.gdb"
>>> arcpy.Buffer_analysis("AWC_Clip", "AWC_Buffer", "20 Feet", "LEFT", "", "All")
```

To use a tool you must know its proper syntax, you can find this in the ArcGIS Desktop Help

The syntax for Buffer_analysis:

Buffer_analysis (in_features, out_feature_class, buffer_distance_or_field, {line_side}, {line_end_type}, {dissolve_option}, {dissolve_field})

Notice that when one element is not used there are still quotations there as a place mark

For more on syntax:

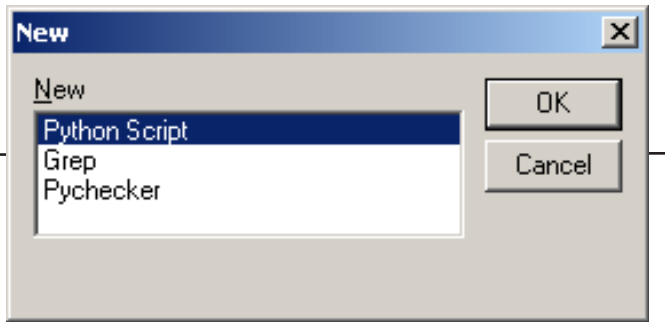
http://resources.arcgis.com/en/help/main/10.1/index.html#/Writing_Python_scripts/002100000021000000/

Topic: Python Scripting
Problem: It can be tedious to type all the code every time you want to run some unique function you have created.

Solution: Writing an actual script that can be used just like any other geoprocessing tool.

Data used:

License: Basic



Turning your code into a script

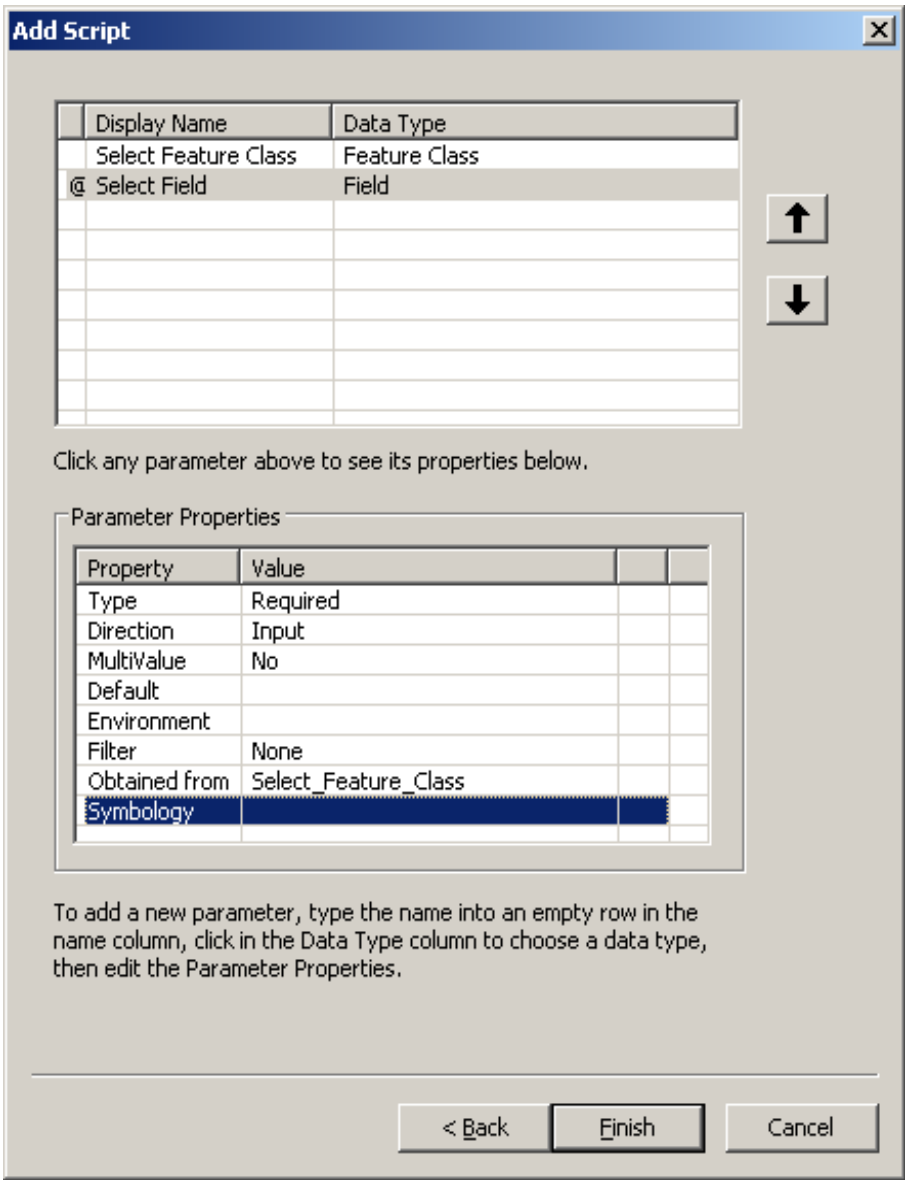
Open PythonWin or note pad

Start a new project as a Python Script

Enter the following code, can you guess what it is going to do? Save it as Enum.py

```
# Import arcpy module
import arcpy

# Script arguments
CampbellCreek = arcpy.GetParameterAsText(0)
if CampbellCreek == '#' or not CampbellCreek:
    CampbellCreek = "CampbellCreek" # provide a default value if unspecified

Field_Name = arcpy.GetParameterAsText(1)
if Field_Name == '#' or not Field_Name:
    Field_Name = "ComID" # provide a default value if unspecified

# Local variables:
CampbellCreek__2_ = CampbellCreek

# Process: Calculate Field
arcpy.CalculateField_management(CampbellCreek, Field_Name, "autoIncrement()", "PYTHON_9.3", "rec=0\\
ndef autoIncrement():\\n global rec\\n pStart = 1 #adjust start value, if req'd \\n pInterval = 1 #adjust interval
value, if req'd\\n if (rec == 0): \\n  rec = pStart \\n else: \\n  rec = rec + pInterval \\n return rec")
```

for pre-written text \GISCLass\Docs\Interval



In the toolbox you created earlier, right-click and select add script when prompted.

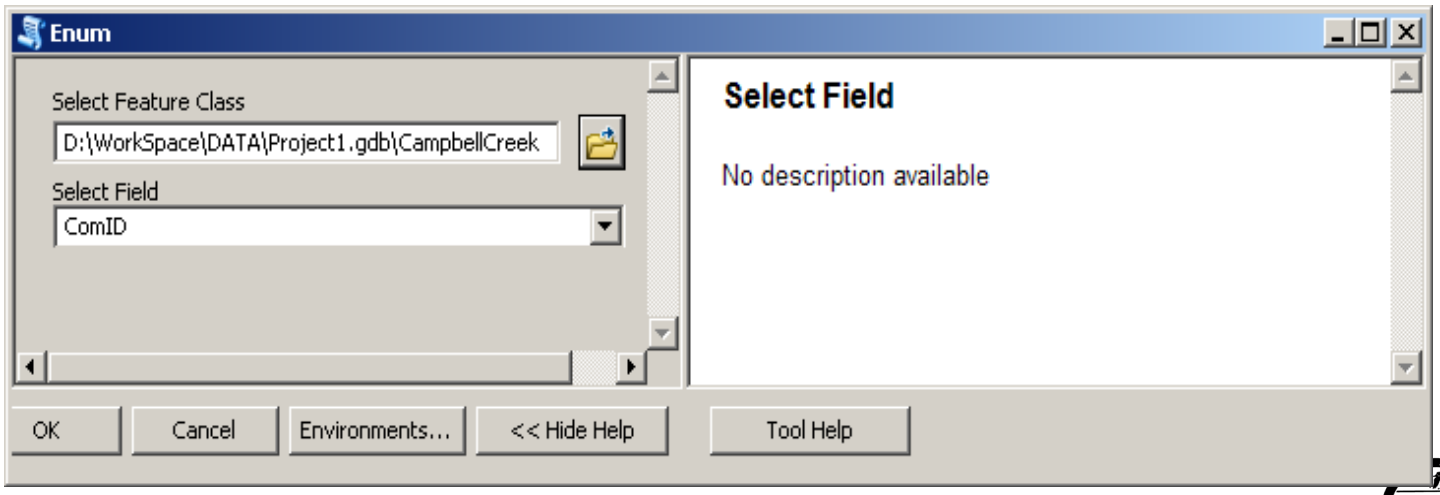
Name it

In this window we need to set up inputs and outputs. For this particular script we have two inputs, one for the layer and one for the field.

Add the data as it looks here and be sure to set the **Obtained from** for the select field input in the lower window

Now open up an ArcMap and bring in the **CampbellCreek** feature class and set the field **ComID** to all zero using the field calculator

Now run the script on **Cambellcreek : ComID.**



Topic: Python tools and addins
Problem: Creating a tool that you can share with others

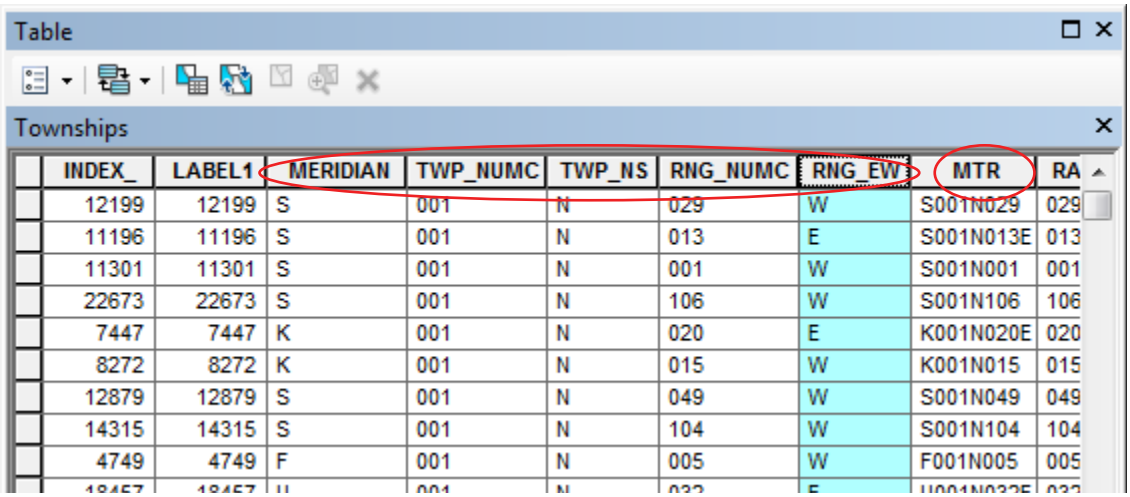
Solution: Using the addin tool wizard to create a tool that others can use in their own projects

Data used: \GISCLass\Data\MTR\MTR.gdb : Townships

License: Basic

First, you have to find a need for a tool, something that makes common tasks even easier perhaps. Then you write a script as we did before, a script can be shared and is sufficient as a geoprocessing tool, however, an actual tool bar that runs your tool at the press of a button is far more convenient.

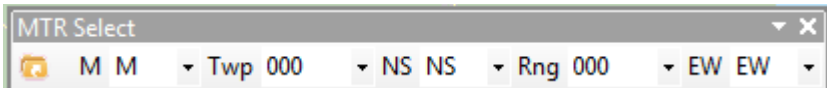
It is common to use the Townships layer to zoom to a selected area in interest. It is very clumsy to scroll through the table and can be difficult to use the select by attribute if you forget all the zero's.



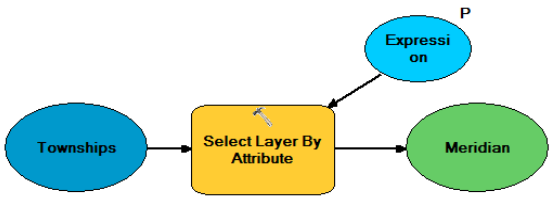
INDEX_	LABEL1	MERIDIAN	TWP_NUMC	TWP_NS	RNG_NUMC	RNG_EW	MTR	RA
12199	12199	S	001	N	029	W	S001N029	029
11196	11196	S	001	N	013	E	S001N013E	013
11301	11301	S	001	N	001	W	S001N001	001
22673	22673	S	001	N	106	W	S001N106	106
7447	7447	K	001	N	020	E	K001N020E	020
8272	8272	K	001	N	015	W	K001N015	015
12879	12879	S	001	N	049	W	S001N049	049
14315	14315	S	001	N	104	W	S001N104	104
4749	4749	F	001	N	005	W	F001N005	005

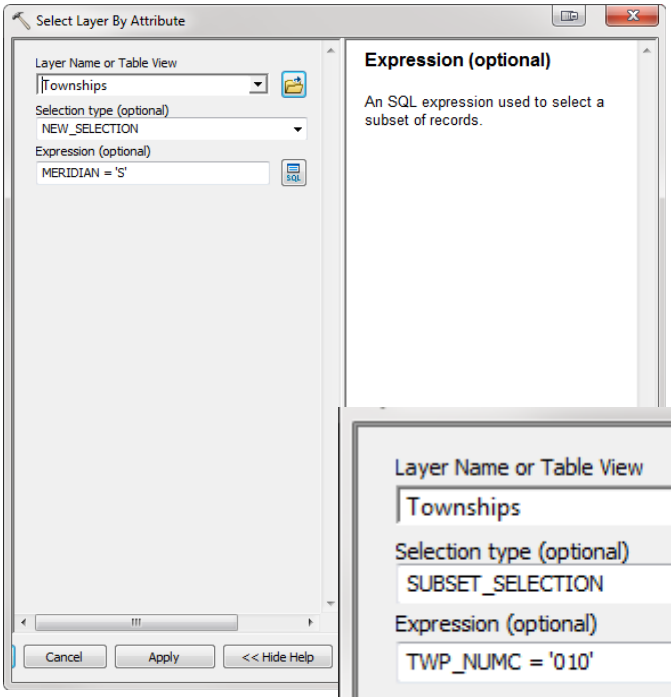
The **MTR** Field is a unique **ID** Field that can be created using 5 other fields

What I want is a tool bar that anyone can use just by using pull downs to select the attributes



To start, you have to follow a logical process. Using model builder is a good starting point.
Start a new model builder and add the Townships layer (Data/MTR/MTR.gdb)
then start model builder and add Select Layer by Attribute tool

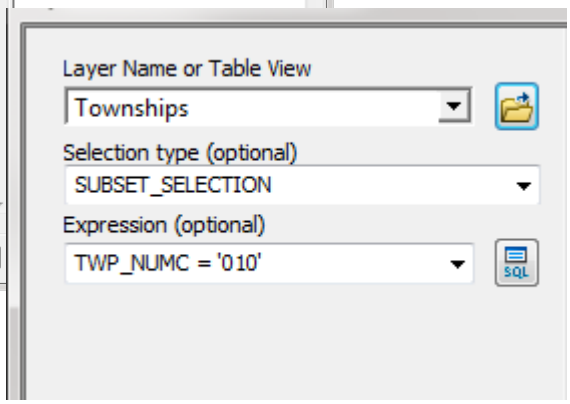




Set the tool as it is to the left and then set the variable, **Expression** as a parameter.

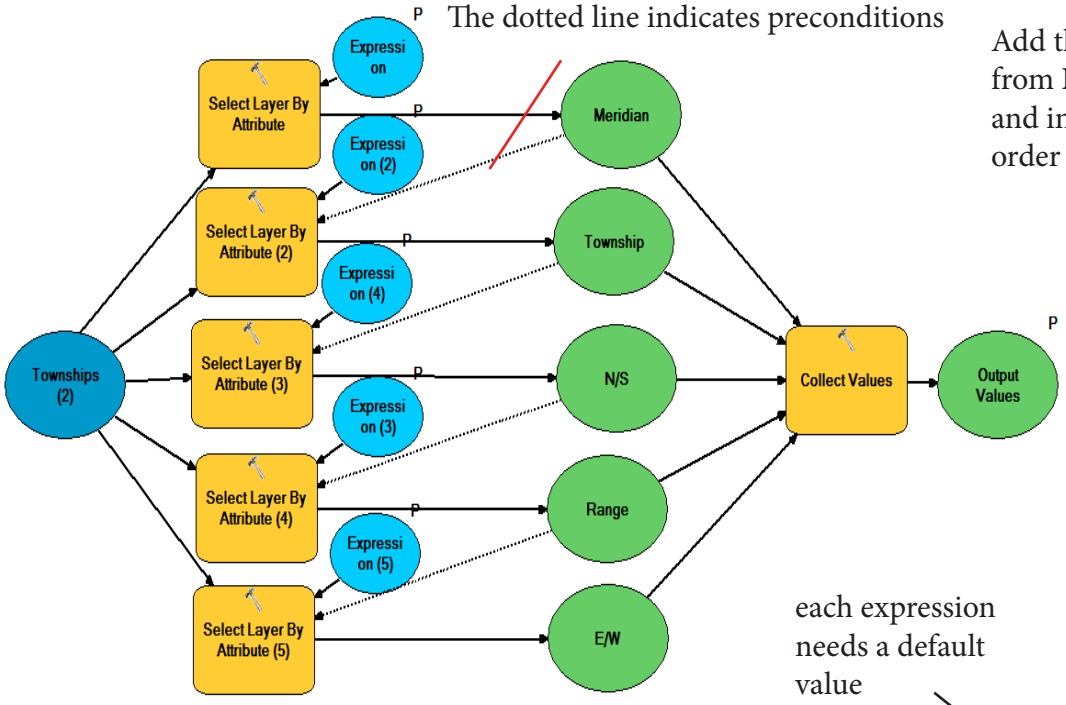
save the model, run it and see what happens

add a second **Select Layer by Attribute**.
Set the tool as below, also with Expression as a variable save and run the tool again to see what happens



add the other three parameters in a similar fashion.

MERIDIAN = 'S'
TWP_NUMC = '010'
TWP_NS = 'N'
RNG_NUMC = '010'
RNG_EW = 'W'

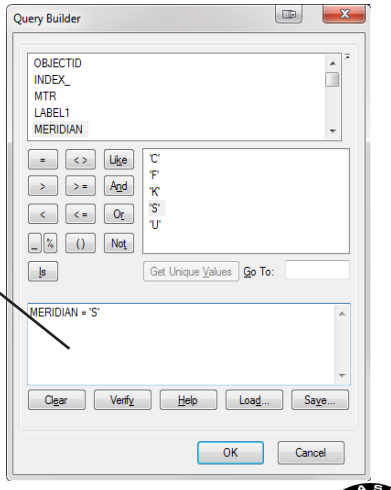


The dotted line indicates preconditions

Add the **Collect Values** tool from **Insert/Model Only Tools** and insert all of the results in order

Now save and run the whole tool and see what happens.

each expression needs a default value



Select Model from the tool bar and Export as Python Script, you can then open the script in a note pad.

Clear selected Townships and open the python window

Copy the code from the note pad and past them into the python window, hit enter twice to run the code. Does it work?

Read what the errors are and see if you can resolve them

Select out the following line of code by putting # in front of it in the note pad.

arcpy.ImportToolbox("Model Functions")

Copy and paste and try again. Success?...sort of .
Looks like the 'CollectValues does not work here

So we have the select part but not the zoom part. We can use this to start out python code but we need to rearrange things

Lets start by adding some important elements.....

```
import arcpy
from arcpy import env

#set workspace, not necessary for script, only python window
mxd = arcpy.mapping.MapDocument("CURRENT")

#location of Township layer
env.workspace = r"N:\GISCLass\Data\MTR\MTR.gdb"
```

set this part to point at your township layers location

Lets change the variable names to make it more readable

```
Expression = arcpy.GetParameterAsText(4)
if Expression == '#' or not Expression:
    Expression = "MERIDIAN = 'S'" # provide a default value if unspecified
```

Follow this example for the rest of the parameters

```
M = arcpy.GetParameterAsText(0)
if M == '#' or not M:
    M = "S" # provide a default value if unspecified
```

Delete the rest of the code, that won't work where we are going with this.
Next we need to concatenate all the parameters to equal the MTR Field.

Here is an example of concatenating in python: http://www.pythonforbeginners.com/concatenation/string-concatenation-and-formatting-in-python

so lets try

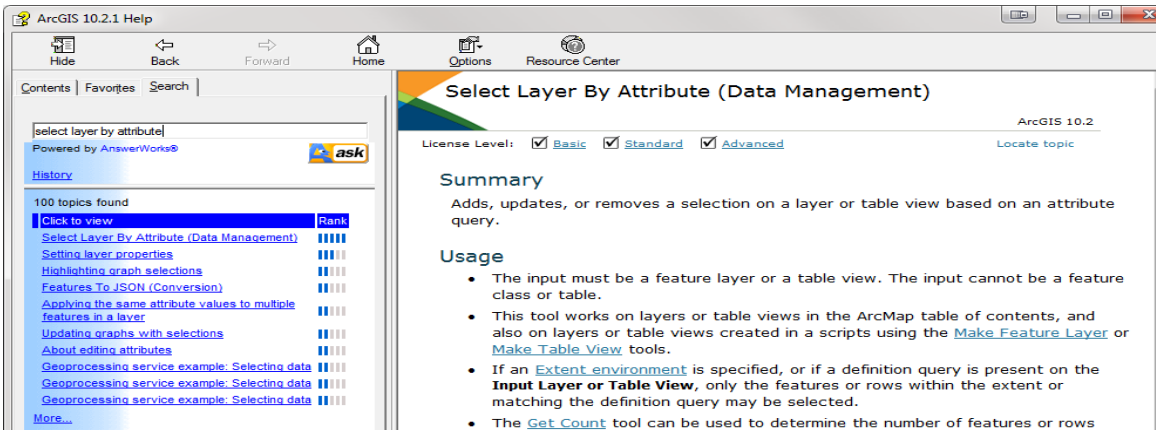
```
# Concatenate user input and define the variable
MTRsel = M + Twp + NS + Rng + EW
```

So now we have gathered the users input for the parameters plus we have default values and put them into one parameter. Now we can use a tool to select an attribute. As before, we can use Select By Attribute tool.

To find the syntax, In ArcMap search for the tool and drag it into the python window

arcpy.SelectLayerByAttribute_management(
As you start typing it will auto fill with your options. As it appears select "Townships" add a comma, and select "NEW_SELECTION", this stuff writes itself! The last part is an SQL expression

Open up Help/ ArcGIS Desktop Help and search for
"select layer by attribute", this will show the whole syntax for the tool



There lots of good example there but nothing we can use. Python is an open source language so there are a lot of resources out there. This blog is a good place to ask questions or find answers: http://gis.stackexchange.com/questions

This is the final piece
'"MTR" =' + "'%s'" %MTRsel
This will select the township that the user requests
Try copying the code so far and paste it into the python window, still working?
Now we need to zoom.

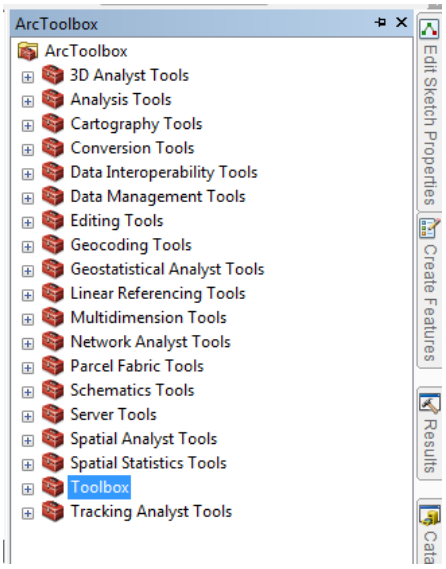
Search in the ArcMap help for **Zoom to Selected Features**, anything? Take a look at DataFrame (arcpy.mapping). this method: df.zoomToSelectedFeatures() requires us to define the dataframe, important if there are more than one dataframe .

df = arcpy.mapping.ListDataFrames(mxd)[0]

Another method that does not require defining the dataframe is:

arcpy.mapping.ListDataFrames(mxd)[0].zoomToSelectedFeatures()

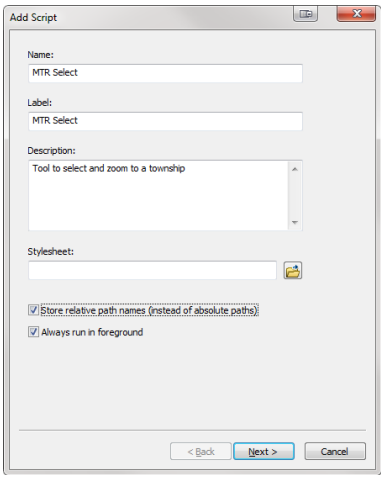
Choose a method and try it out. This works in the python window, now lets make it a more functional tool script.

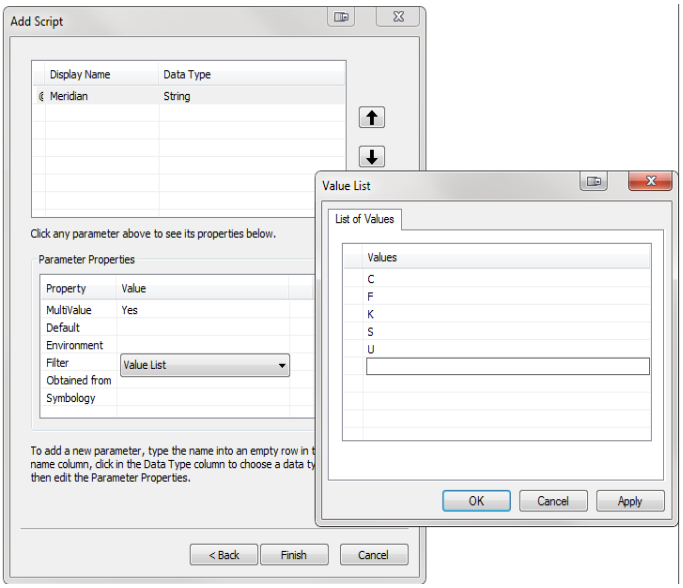


Open the ArcTools window Right-Click and add a new toolbox Then right click on the new toolbox and add a new script

This will open a wizard that will set of tool. Give it a name, description then click next.

The next screen asks for the script that we created earlier.





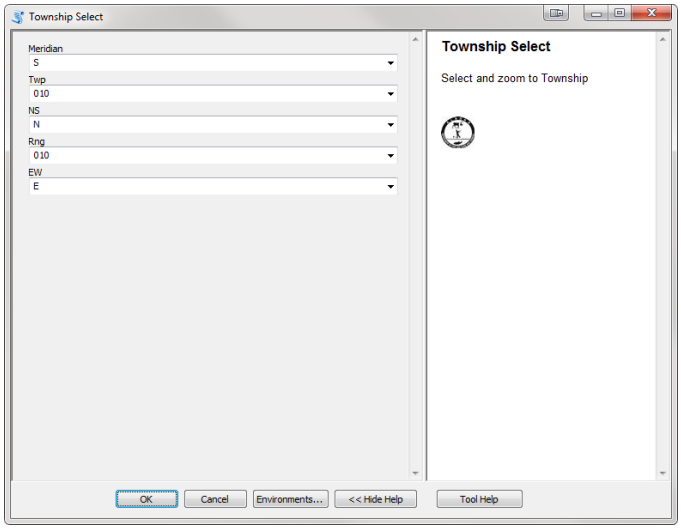
All of the parameters will be strings supplied by a list that we have to create.

Add a value to the default as well

Add the rest in the same order as the script

When you get to township and range, just enter 001 thru 010

Click finish, then double click on it in the Arc-Tool window



You can go back and edit the properties, add more description or help files to the tool.

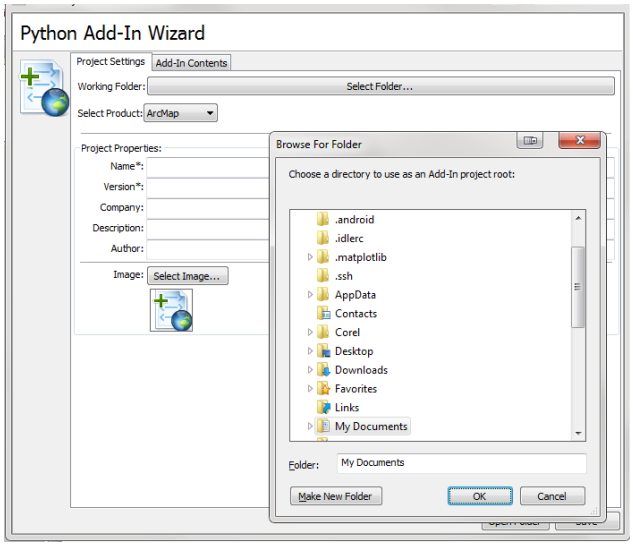
Topic: Create a Tool in python and use the Add-In Wizard to put the tool on a tool bar
Problem: The previoius script works pretty well, but we can take it one step further and make it a fully functional tool bar that will be more convieniant

Solution: Here is a bit of software we need to make this happen: GISCLass\Data\Software\bin addin_assistant.exe

Data used: \GISCLass\Data\MTR\MTR.gdb Township

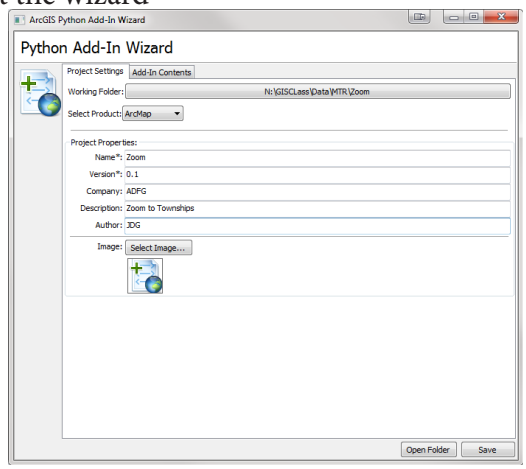
License: Basic

Run the addin_assistant.exe



First, you must create a new folder to save this project in. Name it **Zoom**

Fill out the wizard



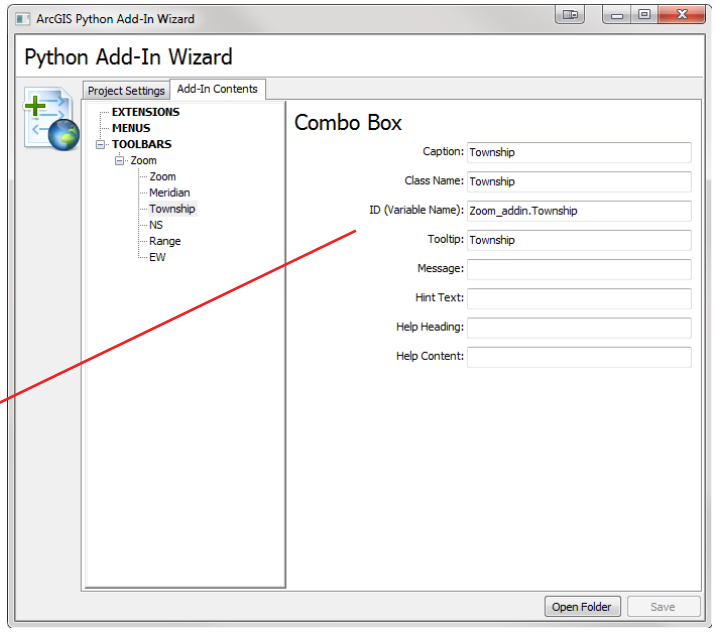
Go to the **Add-in Contents** tab, click on **Toolbars** then right click and select **New Toolbar**
Name it **Zoom**

then we need to create a combobox for each parameter as well as one button to zoom.

Right click on zoom and select **New Combo Box** we will need five of these. Then add a **New Button**

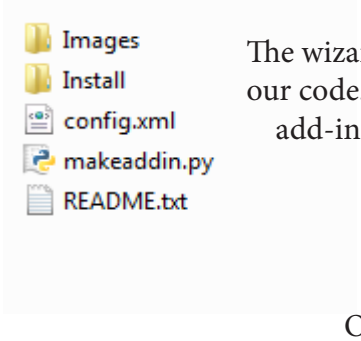
Fill out Captions for each combobox, Add tool tips

if you change the class name and the id, it will be easier to follow in the code



For the Button you can add an image add the more.png (\GISCLass\Data\MTR\Images)

Once you fill out all the combo boxes and button, Save and open the folder.



The wizard creates two folders and three files. The config.xml will be used in our code. The makeaddin.py file is the actual python code that will create our add-in package once we are finished with the code

Open the install folder and open the file with Idle (Start/ArcGIS/Python)

Much of the code is written for you using the wizard. But it may not be in the right order. Arrange the comboboxes in the same order we created them and place the button at the bottom

You can use # or “”” to comment out text

```
class Meridian(object):
  """Implementation for Zoom_addin.Meridian (ComboBox)"""
  def __init__(self):
    self.items = ["C", "F", "K", "S", "U"]
    self.editable = True
    self.enabled = True
    self.dropdownWidth = 'WW'
    self.width = 'WW'
  def onSelChange(self, selection):
    pass
  def onEditChange(self, text):
    pass
  def onFocus(self, focused):
    pass
  def onEnter(self):
    pass
  def refresh(self):
    pass
```

The names need to match the config.xml file this is the variable being passes

Open the config.xml file to compare

Add the meridian letters to self.items, this is the list for the drop down

adjust the width to just two ‘W’

Follow this format for the NS and EW

Township and range need some extra code

Add the following to the front of the township combobox

```
Twp = []
for row in arcpy.SearchCursor("N:\GISCLass\Data\MTR\MTR.gdb\Townships"):
   Twp.append(row.TWP_NUMC)
```

this needs to point to where your township layer is

```
class Township(object):
   """Implementation for Zoom_addin.Township (ComboBox)"""
   def __init__(self):
      self.items = (Twp)
      self.editable = True
      self.enabled = True
      self.dropdownWidth = 'WWWW'
      self.width = 'WWWW'
   def onSelChange(self, selection):
      self.sel = selection
   def onEditChange(self, text):
      pass
   def onFocus(self, focused):
      if focused:
         values = (Twp)
         uniqueValues = sorted(set(values))
         self.items = []
         for uniqueValue in uniqueValues:
            self.items.append(uniqueValue)
   def onEnter(self):
      pass
   def refresh(self):
      pass
```

Since township and range are very large lists, we want to populate the combobox automatically, we also need to sort it and only show unique values.

This is the variable being passes

This bit of code does all the sorting and brings only unique values

Do the same for Range

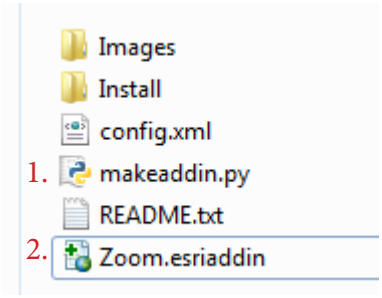
```
class ButtonClass3(object):# Activate Select and Zoom
   """Implementation for MTR_addin.button (Button)"""
   def __init__(self):
      self.enabled = True
      self.checked = False
   def onClick(self):
      #concatenate inputs
       MTRsel = Meridian.sel + Township.sel + NS.sel + Range.sel + EW.sel
      # Select user input value
      arcpy.SelectLayerByAttribute_management("Townships","NEW_SELECTION","MTR" =' + "'%s'"
      %MTRsel)
      # Zoom to selected township
      arcpy.mapping.ListDataFrames(mxd)[0].zoomToSelectedFeatures()
```

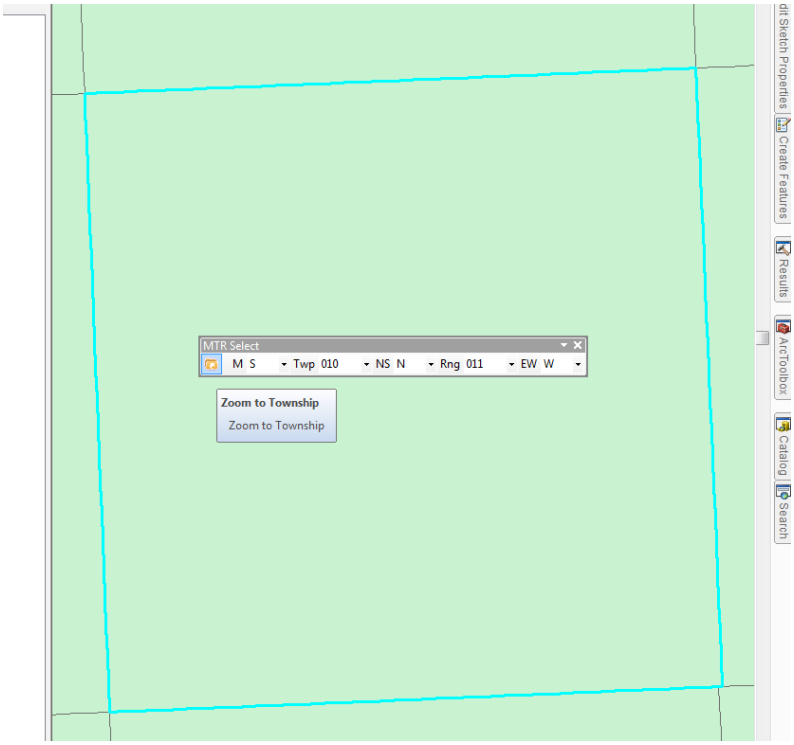
add up the variables

```
# ---------------------------------------------------------------------------
# MTR select : Township Selector
# Created on: 2014-02-24
# Create by Jason Graham ADF&G
# Description:Add-in tool to easily select a township then zoom to the selection
# using a series od combobox to gather input from the user
# Township layer must be in the MXD
# ---------------------------------------------------------------------------
import arcpy
import pythonaddins
from arcpy import env


# This tool uses a specific "Townships" layer located here.
env.workspace = r"N:\GISCLass\Data\MTR\MTR.gdb"

mxd = arcpy.mapping.MapDocument("CURRENT")
```

Add this last bit to the top of the code, edit the path to fit your computer, then it should be ready to run

Save your edits and close the file. Now click on the makeaddin.py, this makes the .esriaddin file that you use to install your addin, click it and install the addin.





Now you can open a new project, add the Township layer and add your new tool bar, give it a try.

Success? If it does not work, you can close the project fix the code and re-make the addins. Every time you change the code, run the makeaddin first then run the Zoom.esriaddin again, you need to restart ArcMap.

Topic: Topology
Problem: Overlaps or gaps in your data can yield poor results and can be tedious to find and repair.

Solution: Topology will find all the Overlaps and gaps as well as many other issues between points, lines and polygons.

Data used: Parcels, sections, AWC_Clip, AWC_SCN_Points

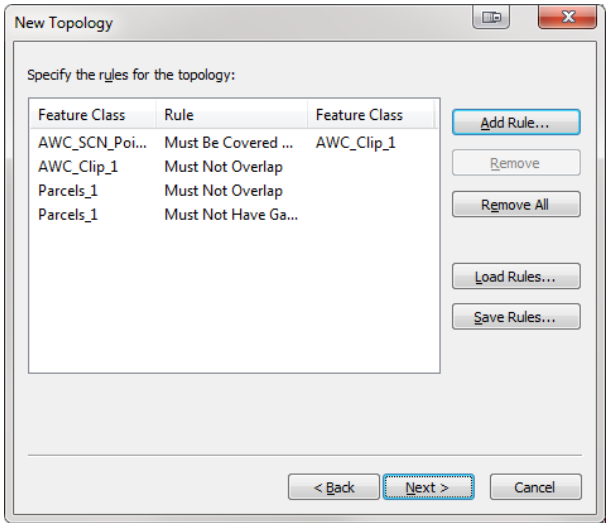
License: Advanced

---

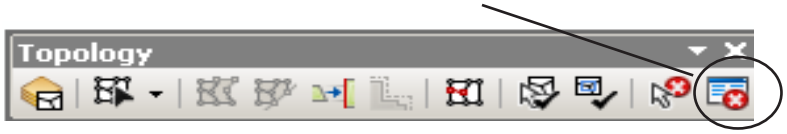
1 Open ArcCatalog, create a new feature dataset called ClassTopo and import the layers; Parcels,Sections, AWC_Clip, and AWC_SCN_Points

2 Right click on the ClassTopo data set and choose New > Topology
Follow the wizard and name it Homer_Topology, accept the defaults and check all the layers

3 Add the Rules **'Must not have Gaps'**, and **'Must not Overlap'** for the Parcels layer. Add '**Must not overlap**' to the AWC_Clip, and AWC_SCN_Points '**Must be covered by Line**' of AWC_Clip, and AWC_Clip '**must be covered by end point**' of AWC_SCN_PointsFinish and validate the topology.



4 Start a new ArcMap and add the new topolgy and 'Parcels' layer, and the sections layer Open the layer properties for the topology and go to the Symbology tab Check 'Symbolize by error type' and make the symbols more contrasting

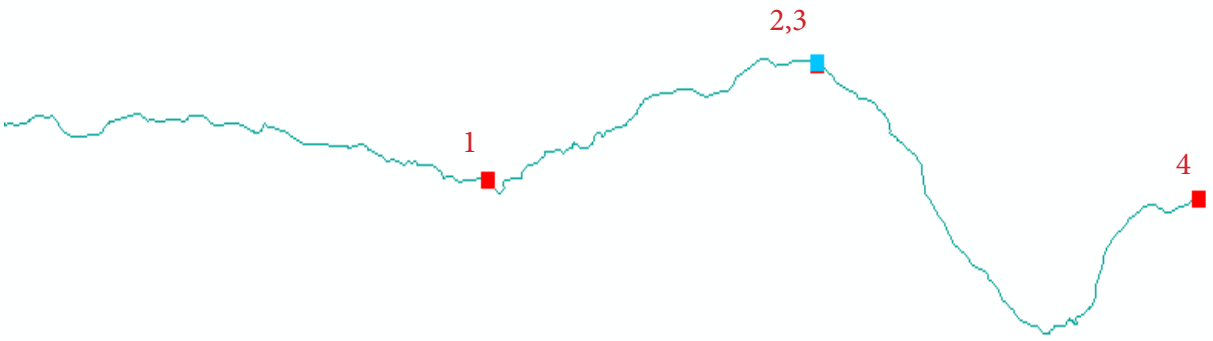


In the error inspector window show: errors from all rules and select search now. How many errors are there?
Zoom in and inspect some of the '**Must Not Have Gaps**' errors. Can you find any that are not roads?

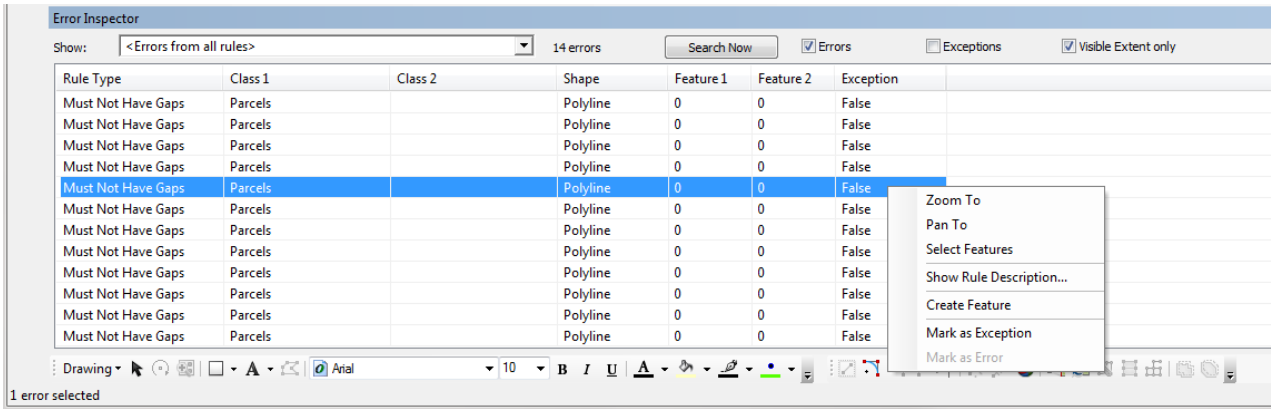
6 Zoom to some of the of the **Must Not Overlap** errors and try to fix them, when done go to the Topology tool-bar select 'Validate Topology in Current Extent'

In the table of contents right-click on parcels and select zoom to layer. Validate the topology then view the errors for the AWC layers

7 zoom to the upper portion of Beaver Creek



8 Determine the Error of these four points and either correct them or mark them as an exception



use the Error Inspector to mark as an exception

Save and stop editing

Topic: GPS

Problem: GPS/xy data comes in many formats, some work natively in ArcMap and some need some manipulation.

Solution: One of the best tools to work with GPS data is DNR GPS. It is a free third party program that quickly converts directly from a gps device or files

Data used: Alaska, GISCLass\Data\GPS: Streampoints.xlsx > _xlnm#Database, or Streampoints .csv
Current.gpx, Waypoints_01-Aug-12.gpx

License: Basic

We will start with an Excel table first

1 Start a new project and the 'Alaska' feature class then go to the GPS folder and bring in the StreamPoints.xlsx > _xlnm#Database or streampoints.csv. Open the table and check out the data. We can tell its in decimal degree's but what projection is it in?

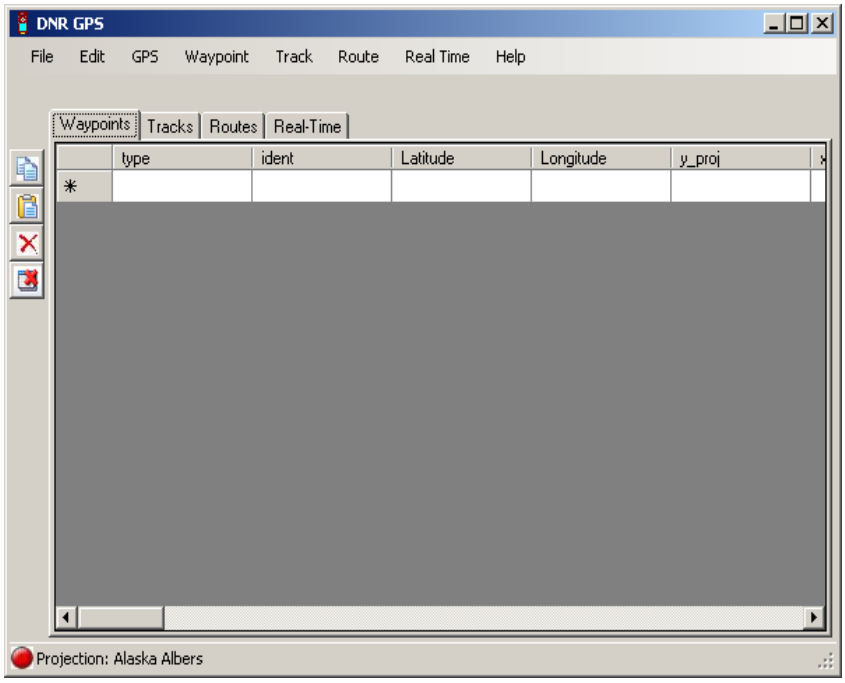
You need to know what projection it was collected in order to set the projection
Otherwise you might have to guess a few times.

2 Right click on the excel table and select Display XY Data

Set the X and Y fields and set the projection to Alaskan Albers

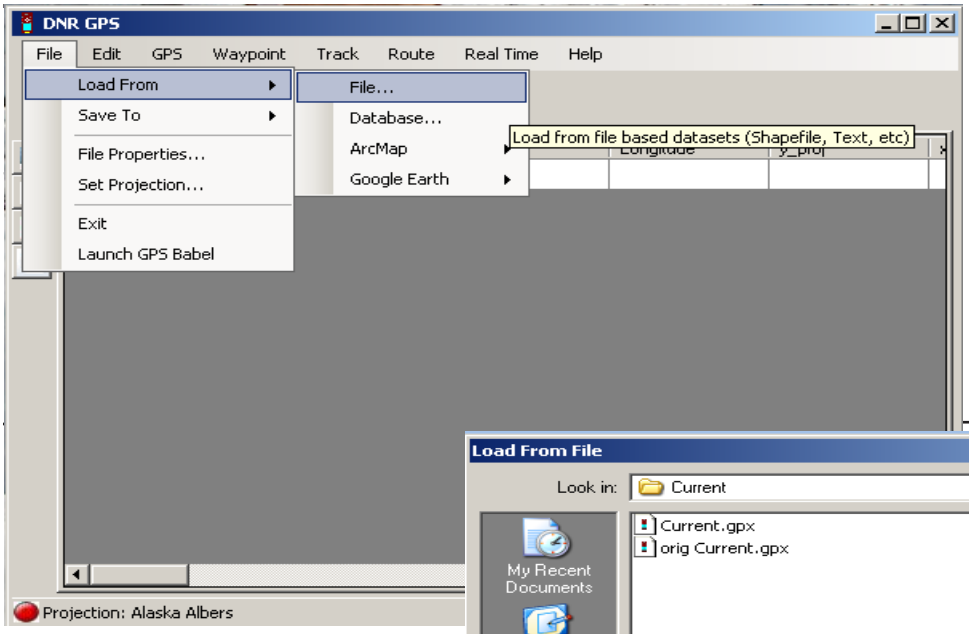
Export the events layer to your database and bring that feature class into your project

Where's the Data? Not where it's supposed to be. Open the ArcToolbox
Go to DataManagement > Projections > Define Projection Redefine the projection as GCS_North_American_1983

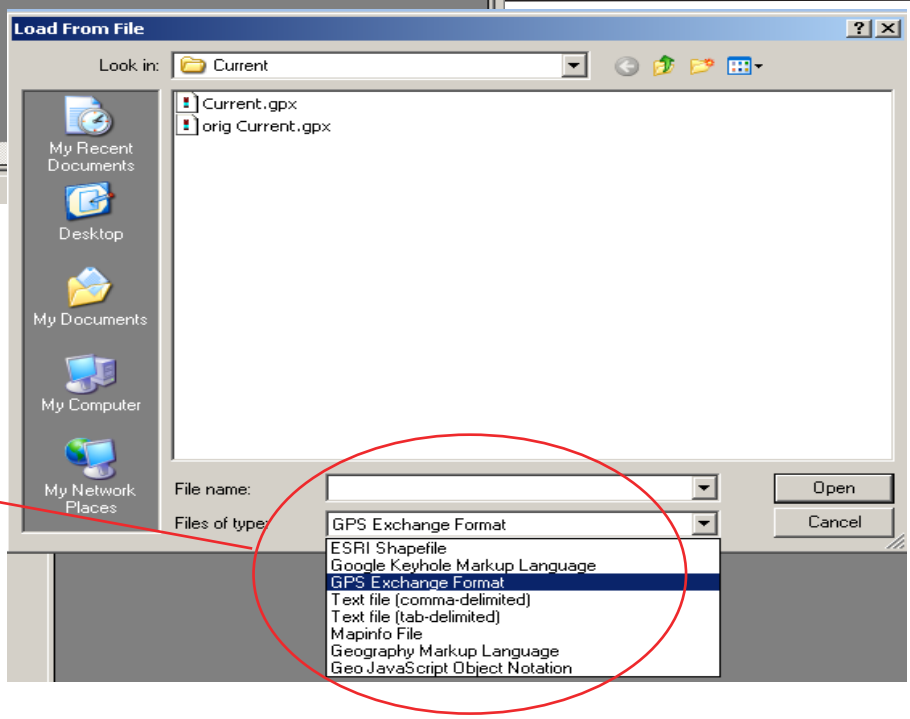


Most GPS Devices use the. GPX format
The best tool for this is DNR GPS, or EasyGPS, both free software

Under File, set the projection to Alaska Albers
POSC code : 3338
Datum: NAD83

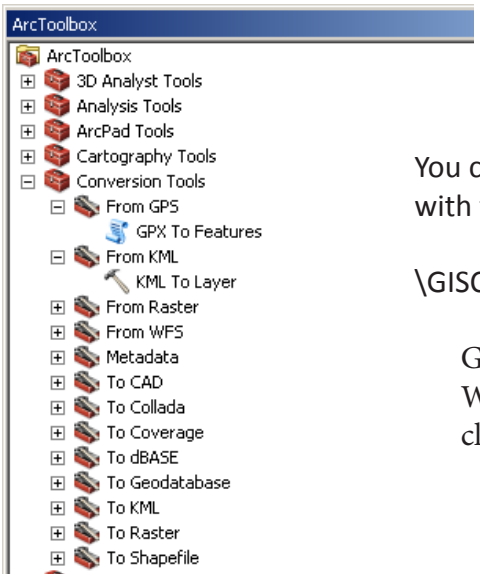


Click file > Load From > File and navigate to the gpx folder and bring in the 'Current.gpx' If you had a GPS device connected, this is where you would find your current track log. The points are usually labeled by date.



Always make sure to set the data type that you are looking for or the one you are saving to

Now save to file, change the format to shapefile and save it to your student folder. You can also use this method for many data types like KML or text files. The geodatabase functionality is not yet working.



You can also use geoprocessing tools to work with the GPX data :

\GISClass\Data\GPS\GPX

Go to arctool box and run the GPX to Features tool, convert the Waypoints_01-Aug-12.gpx into your geodatabase, bring the new feature class into an ArcMap project and view the data

Topic: Bathymetry / Lidar
Problem: 3-D data is more complex then just points, lines, and polygons

Solution: There are many tools in ArcMap that work with Bathymetry and LIDAR

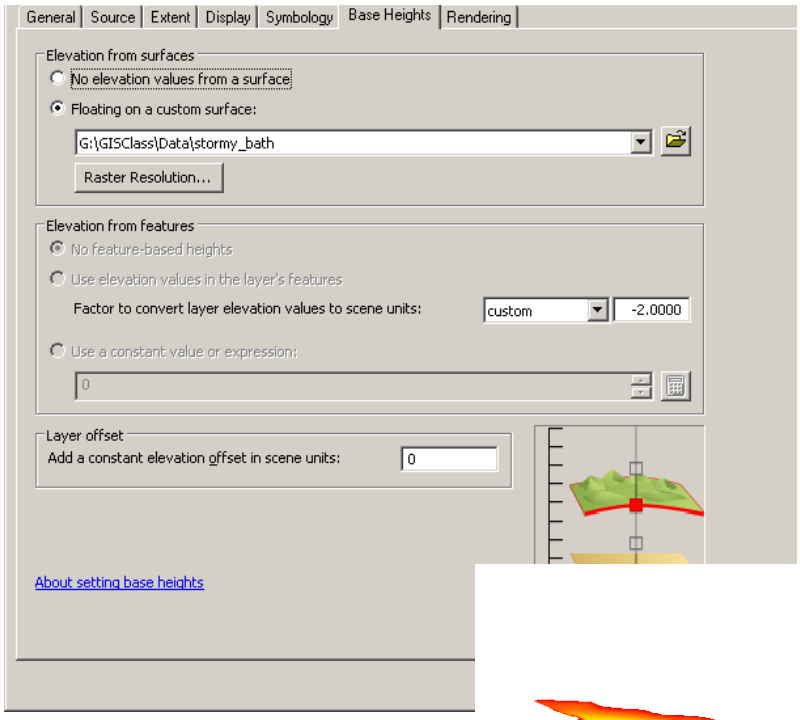
Data used: Stormy_Bath, Stormy_Pike

License: Basic Spatial Analyst extension, 3-D analyst extension

---

Bathymetry can come in various formats, DTM's, sounding point data, BAG(Bathymetric Attributed Grids) and Lidar

Open ArcScene and add the Stormy_Bath file. This is bathymetry for Stormy Lake. Open the properties and go to the **Base Heights** tab

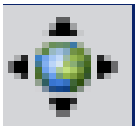


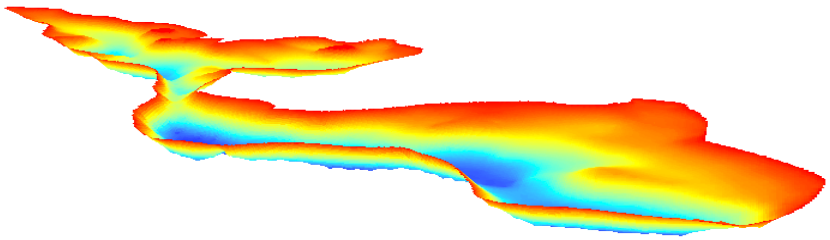
Check floating on custom surface

Set the height to -2

Change the color in symbology

Use this button to view the lake





http://www.ngdc.noaa.gov/mgg/bathymetry/multibeam.html

In ArcScene, open ArcToolbox

Go to Spatial Analyst tools > Surface > Contour
Select Stormy_bath and set the contour to 1

Run it again and set the contour to 5

Set the base heights of the contour layers

Close ArcScene and bring those layers into ArcMap
Also add Stormy_Pike

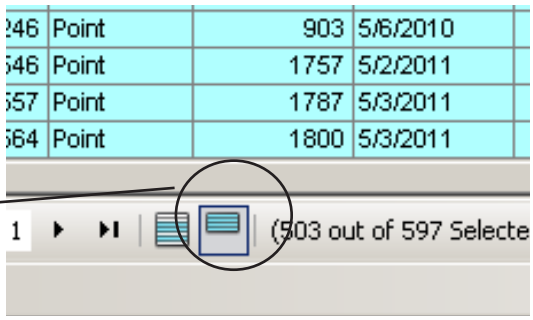
Open ArcToolbox and go to Spatial Analyst tools > Extraction > Extract Multi Values to Points

Set the input as the points layer and the Stormy_Bath layer

Open the table and view the results

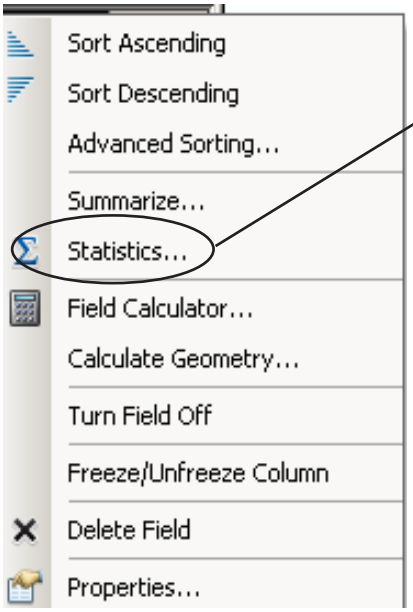
Select the column with the results and sort ascending

Select all records without NULL Values choose the selected layers



What is the mean depth to find pike in stormy Lake?

Hint



Bonus:

Try this when you have some extra time (it can take a long time to process)

In Arctoolbox , select Spatial analyst > Interpolation > Natural Neighbor

For input use Norton_soundfromXY

X value is in field 3 keep the defaults and run

Topic: Lidar
Problem: Lidar data sets can have millions of points in a cloud that can be cumbersome to view and can be overwhelming .

Solution: ArcMap/ ArcScene can easily view and manipulate LAS datasets allowing you to view all or specific level of data returns

Data used: AK_Kenai_2008_2133.las, Mat83.lasd

License: Basic / 3-D analyst extension

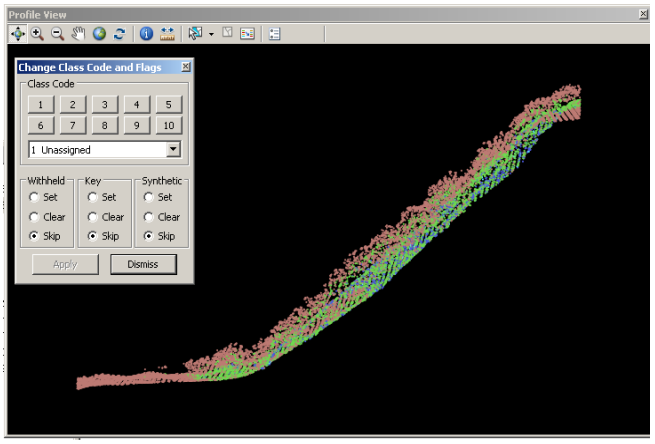
Open ArcCatalog

1 In your student folder (not the .gdb) right-click and create a new LAS dataset, call it 'Kenai'

2 In ArcToolBox go to **Data management Tools > LAS Dataset > Add File to LAS Dataset**
Add the 'Kenai' for the input, navigate to the Kenai folder in the data folder and select **AK_Kenai_2008_2133.las**. You will need the **3-D analyst Extension** to fully utilize ArcMap's capabilities.

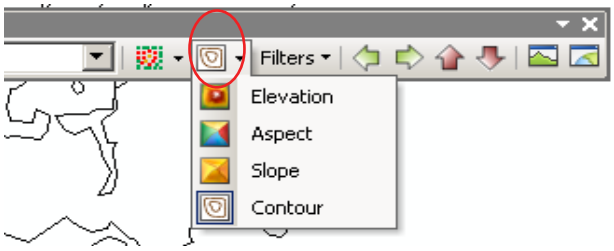
3 Start ArcMap and bring in the new LAS dataset and open the LAS Dataset toolbar

4 Click on the **Profile View** button and draw a path somewhere on the data



View the data and try to edit some of the points
Close the profile window

5 Change the view to Contour, this looks a little rough



Fix this by changing the filter to **Ground**

Close ArcMap and open ArcScene
Create a new LAS dataset and add the MAT_083_SW_PtCl point cload to it

Try find the tower.

for more MatSu Lider/DEM go to http://matsu.gina.alaska.edu/LiDAR/

Topic: Data Driven Pages

Problem: Many mapping projects require a series of maps depicting similar data in different areas. It can be very tedious to create a new project for each map in the series

Solution: Data driven pages allows you to automate the process of creating a series of maps with dynamic text. Once you set up the initial project, you can quickly create hundreds of maps

Data used: Parcels, Roads, AWC_Clip

License: Basic

---

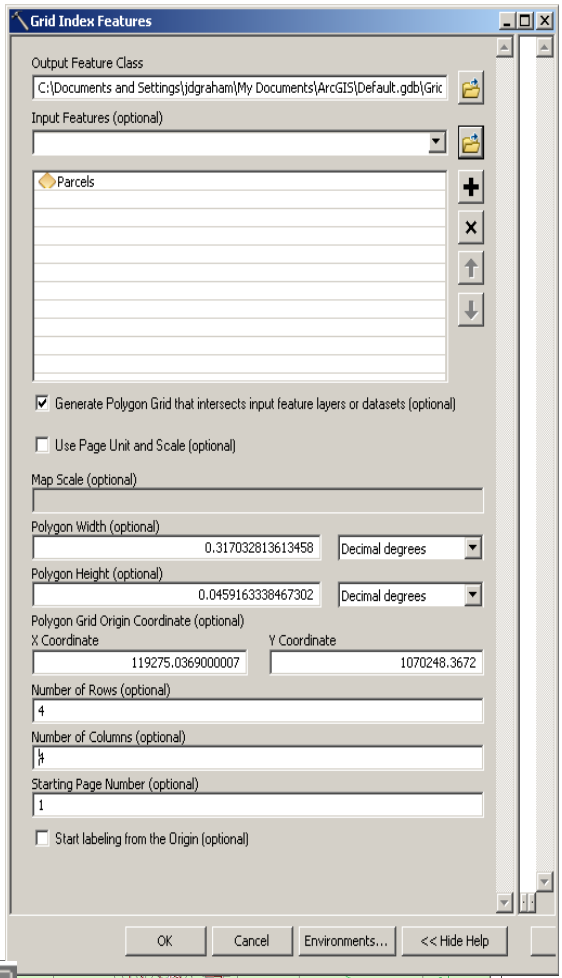
1 Open a new Arc Map and add Parcels, Roads, and AWC_Clip

First we need to create a Grid Index layer

2 Open ArcToolbox and go to **Cartography tools > Data Driven Pages > Grid Index Features**

3 Add in the Parcel layer and set it to a 4 by 4 grid

4 Add the Data Driven Pages Toolbar and click on the first button, Check the Enable data driven pages box and make sure the **GridIndexFeatures** is in the Layer box and go to the Extent tab, Set the margin to 100%, close the window

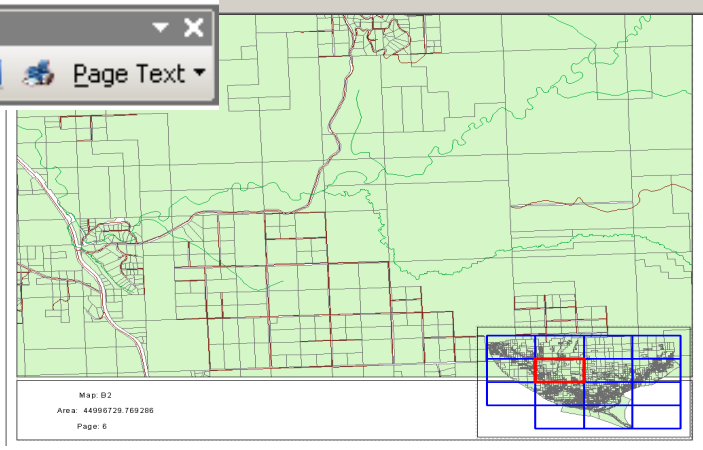


5 Go to the Layout view. Under File, go to Page and Print Setup Set the paper to Landscape and check the Scale Elements box near the bottom, adjust the data frame to fit the page.

6 Go to **Insert > Dynamic Text** and insert **Current date, Data driven Page Number, Data driven Page Name, Data driven Page Attribute**, set the attribute to shape area.

7 Open the properties of Data Driven Page Name and type "Map " in front of the code. Arrange your text and flip through your maps using the Data driven Page toolbar.

8 Insert a new Data Frame and add Parcels and the index grid. In the data frame properties set the extents to the other data frame. Add labels to the index grid. Scroll through the maps again to check the index extents



Topic: Tips & Tricks
Problem:

Solution:

Data used:

License:

My Places:

Add the **Data Frame** toolbar to your project
Click on the flag to open **My Places.**
Click on the load button and import the MyPlaces.dat file in the data folder.

Try adding some of your own places using the add from button

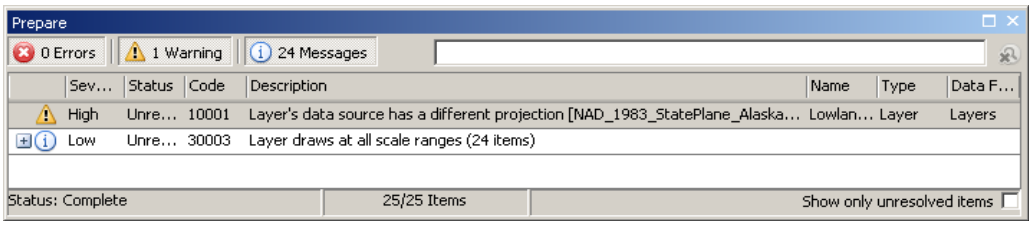
Clip to Frame to shape:

Create a new polygon feature class and call it 'Frame'

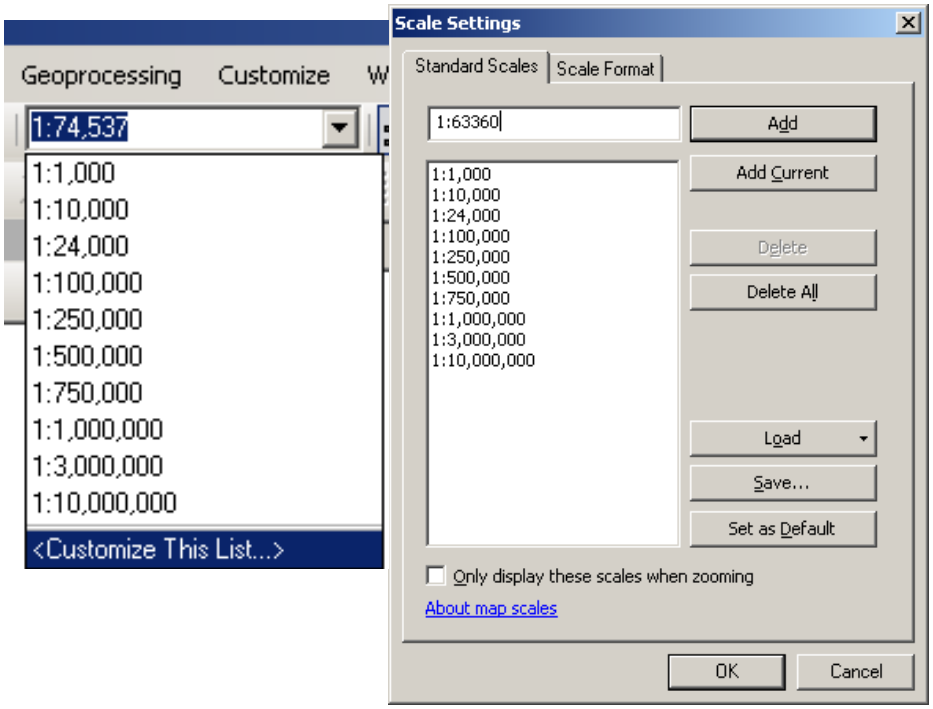
Start editing that layer and draw a circle around Homer. Turn the layer off. In the data frame properties open the data frame tab. Set the clip options to clip to shape and specify the shape as 'Frame'

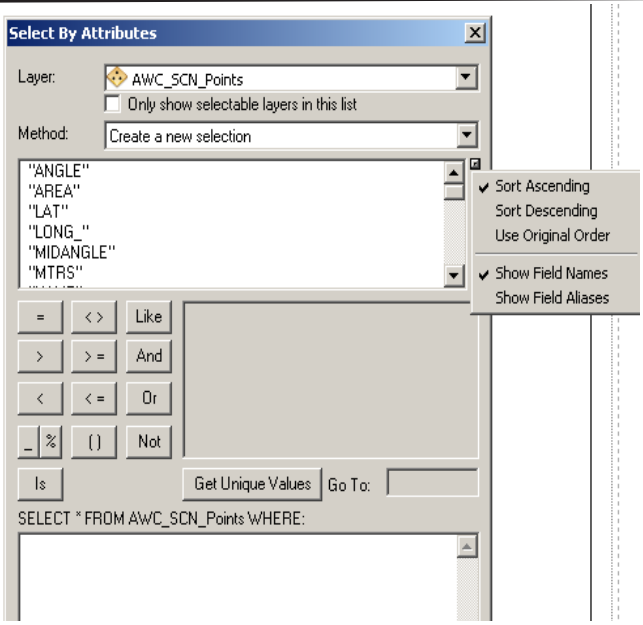
Analyze Map

Under file go to Analyze Map, Does your map have any issues? Fix these to speed up your project



Custom Scale:
Open the Map Scale and select Customize, Add a new scale

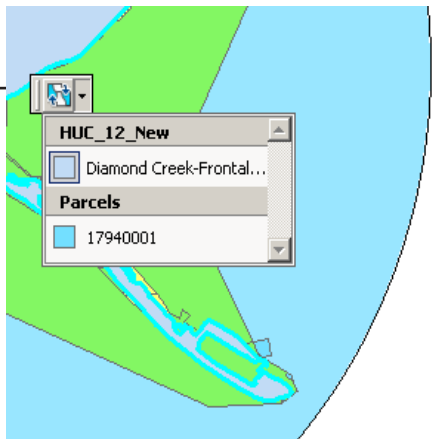




Sorting:

Open the Select attributes by Properties Window and select the AWC_Clip. Click on the tiny button next to the list window, choose sort ascending

Selecting:
Bring in the HUC_12_New layer and start an editing session on the Parcels layer. When you attempt to select a layer, a layer select button shows up, when you click on it, you will see the option to choose which layer to select

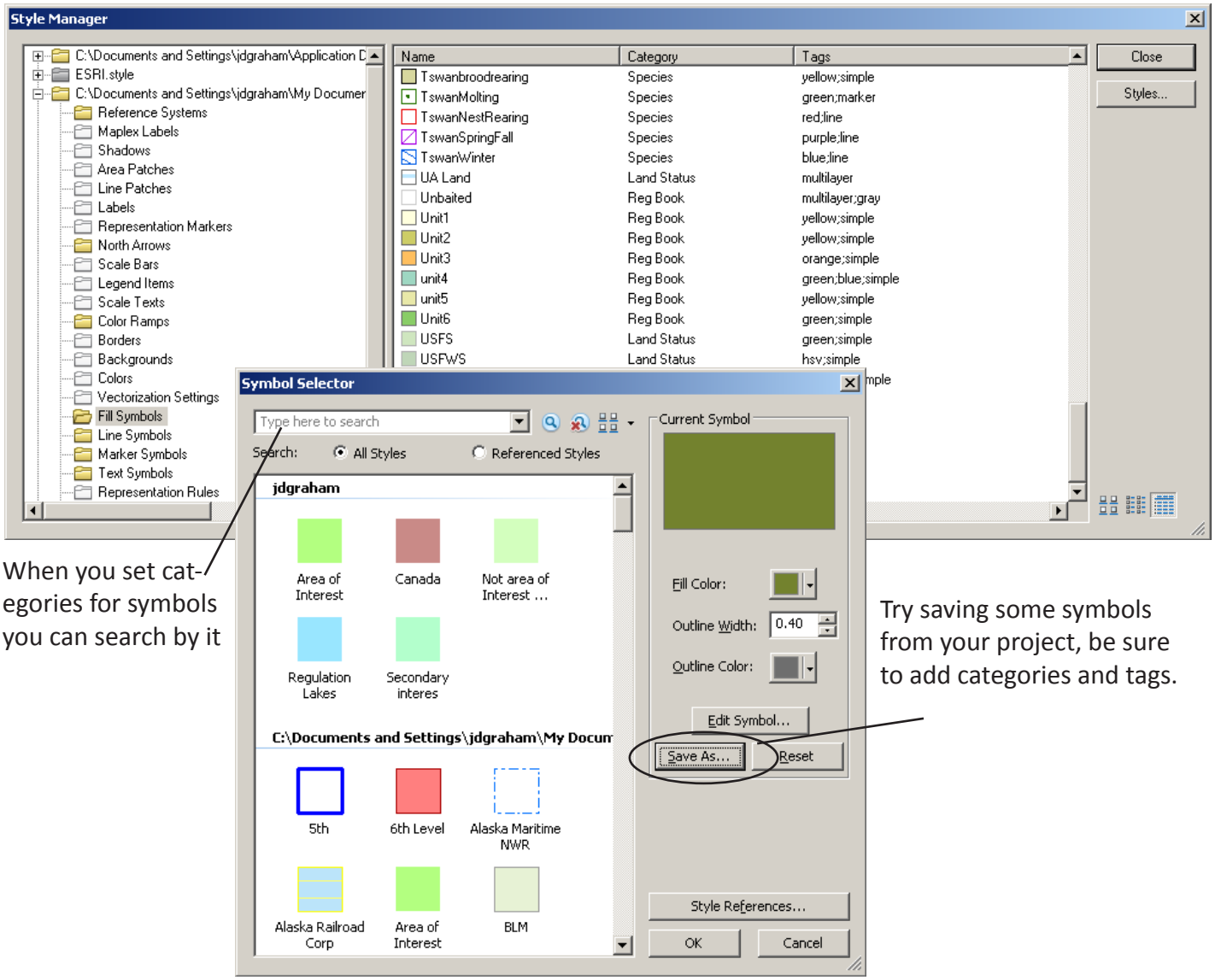


Turn off all layers and zoom to the HUC layer. Open its properties and go to the labels tab. Click into the Expression tab. Set the Parser to python and check the advanced box. Enter the following code

```
def FindLabel ( [HU_12_Name] , [Area_Acres]  ):
  if long ([Area_Acres]) >= 25000:
    return "<CLR red='255'><FNT size = '14'>" + [HU_12_Name]  + "</FNT></CLR>"
  else:
    return [HU_12_Name]
```

You can get other code examples by clicking on the Help button.

Click on Customize and open the style manager. Open your style folder and go to Marker Symbols. Right-click in the window with the symbols and add a new marker. Change type to picture symbol and add the Pike.png from the Pics folder. Now go to the North arrow section and add new north arrow , click on the symbol button and select the new Pike symbol



When you set categories for symbols you can search by it

Try saving some symbols from your project, be sure to add categories and tags.

```
# ---------------------------------------------------------------------------
# Township Selector For Python Script
# Created on: 2014-02-24 15:28:23.00000
# Create by Jason Graham
# Usage: Select2 <NS> <EW> <Rng> <Twp> <M> <MTR>
# Description:Add-in tool to easily select a township then zoom to the selection
# ---------------------------------------------------------------------------
# Import arcpy module
import arcpy
from arcpy import env

#set workspace, not necessary for script, only python window
#mxd = arcpy.mapping.MapDocument("CURRENT")

#location of Township layer
env.workspace = r"X:\JasonGraham\Landstatus\Data\ls2014.gdb\BaseData"


# Collect user input for Meridian, township and range
M = arcpy.GetParameterAsText(0)
if M == '#' or not M:
    M = "S" # provide a default value if unspecified

Twp = arcpy.GetParameterAsText(1)
if Twp == '#' or not Twp:
    Twp = "010" # provide a default value if unspecified

NS = arcpy.GetParameterAsText(2)
if NS == '#' or not NS:
    NS = "N" # provide a default value if unspecified

Rng = arcpy.GetParameterAsText(3)
if Rng == '#' or not Rng:
    Rng = "010" # provide a default value if unspecified

EW = arcpy.GetParameterAsText(4)
if EW == '#' or not EW:
    EW = "E" # provide a default value if unspecified

# Concatenate user input
MTRsel = M + Twp + NS + Rng + EW

# Select user input value
arcpy.SelectLayerByAttribute_management("Townships","NEW_SELECTION","MTR" =' + "'%s'" %MTRsel)

#Zoom to selection
arcpy.mapping.ListDataFrames(mxd)[0].zoomToSelectedFeatures()
```

```
# ---------------------------------------------------------------------------
# MTR select : Township Selector
# Created on: 2014-02-24
# Create by Jason Graham ADF&G
# Description:Add-in tool to easily select a township then zoom to the selection
# using a series od combobox to gather input from the user
# Township layer must be in the MXD
# ---------------------------------------------------------------------------

import arcpy
import pythonaddins
from arcpy import env

# This tool uses a specific "Townships" layer located here.
env.workspace = r"N:\GISCLass\Data\MTR\MTR.gdb"

mxd = arcpy.mapping.MapDocument("CURRENT")

# Select Meridian
class Meridian(object):
    """Implementation for Zoom_addin.Meridian (ComboBox)"""
    def __init__(self):
        self.value = "M"
        self.items = ("C", "F", "K", "S", "U")
        self.editable = True
        self.enabled = True
        self.dropdownWidth = 'WW'
        self.width = 'WW'
    def onSelChange(self, selection):
        self.sel = selection
    def onEditChange(self, text):
        pass
    def onFocus(self, focused):
        pass
    def onEnter(self):
        pass
    def refresh(self):
        pass

# Select Township
Twp = []
for row in arcpy.SearchCursor("N:\GISCLass\Data\MTR\MTR.gdb\Townships"):
    Twp.append(row.TWP_NUMC)

class Township(object):
    """Implementation for Zoom_addin.Township (ComboBox)"""
    def __init__(self):
        self.value = "000"
        self.items = (Twp)
        self.editable = True
        self.enabled = True
        self.dropdownWidth = 'WWWW'
        self.width = 'WWWW'
    def onSelChange(self, selection):
        self.sel = selection
    def onEditChange(self, text):
        pass
```

```
    def onFocus(self, focused):
        if focused:
            values = (Twp)
            uniqueValues = sorted(set(values))
            self.items = []
            for uniqueValue in uniqueValues:
                self.items.append(uniqueValue)
    def onEnter(self):
        pass
    def refresh(self):
        pass

# Select North South
class NorthSouth(object):
    """Implementation for Zoom_addin.NS (ComboBox)"""
    def __init__(self):
        self.value = "NS"
        self.items = ("N", "S")
        self.editable = True
        self.enabled = True
        self.dropdownWidth = 'WW'
        self.width = 'WW'
    def onSelChange(self, selection):
        self.sel = selection
    def onEditChange(self, text):
        pass
    def onFocus(self, focused):
        pass
    def onEnter(self):
        pass
    def onEnter(self):
        pass
    def refresh(self):
        pass

# Select Range
Rng = []
for row in arcpy.SearchCursor("N:\GISCLass\Data\MTR\MTR.gdb\Townships"):
    Rng.append(row.Rng_NUMC)

class Range(object):
    """Implementation for Zoom_addin.Range (ComboBox)"""
    def __init__(self):
        self.value = "000"
        self.items = (Rng)
        self.editable = True
        self.enabled = True
        self.dropdownWidth = 'WWWW'
        self.width = 'WWWW'
    def onSelChange(self, selection):
        self.sel = selection
    def onEditChange(self, text):
        pass
    def onFocus(self, focused):
        if focused:
            values = (Rng)
            uniqueValues = sorted(set(values))
```

```
            self.items = []
            for uniqueValue in uniqueValues:
                self.items.append(uniqueValue)
    def onEnter(self):
        pass
    def onEnter(self):
        pass
    def refresh(self):
        pass

# Select East West
class EastWest(object):
    """Implementation for Zoom_addin.EW (ComboBox)"""
    def __init__(self):
        self.value = "EW"
        self.items = ("E", "W")
        self.editable = True
        self.enabled = True
        self.dropdownWidth = 'WW'
        self.width = 'WW'
    def onSelChange(self, selection):
        self.sel = selection
    def onEditChange(self, text):
        pass
    def onFocus(self, focused):
        pass
    def onEnter(self):
        pass
    def refresh(self):
        pass


# Zoom Button
class ButtonClass1(object):
    """Implementation for Zoom_addin.button (Button)"""
    def __init__(self):
        self.enabled = True
        self.checked = False
    def onClick(self):
        # concatenate inputs
        MTRsel = Meridian.sel + Township.sel + NS.sel + Range.sel + EW.sel
        # Select user input value
        arcpy.SelectLayerByAttribute_management("Townships","NEW_SELECTION","MTR" =' + "'%s'" %MTRsel)
        # Zoom to selected township
        arcpy.mapping.ListDataFrames(mxd)[0].zoomToSelectedFeatures()

#-------------------------------------------------------------------------------------------
```

Data Sources

http://www.asgdc.state.ak.us/

http://www.alaskamapped.org/

http://nationalmap.gov/

http://nhd.usgs.gov/data.html

http://www2.borough.kenai.ak.us/GISDept/

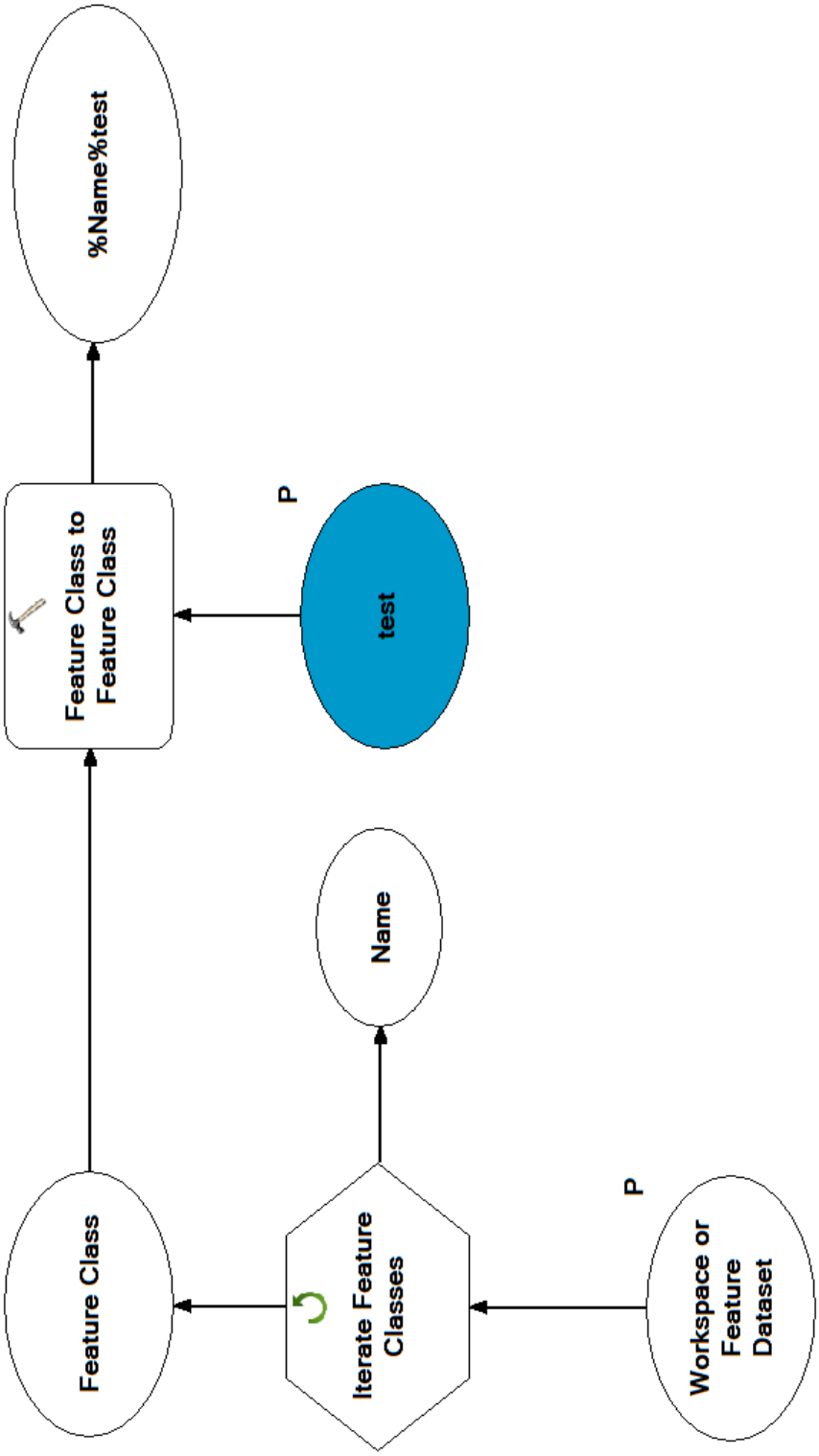
https://github.com/

http://gis.stackexchange.com/questions

Resources
http://www.esri.com/training/main

esri
http://www.esri.com/

ArcGIS Desktop Free Trial
Http://www.esri.com/software/arcgis/arcgis-for-desktop/free-trial


%Name%test
Feature Class to Feature Class
P
test
Name
Feature Class
Iterate Feature Classes
P
Workspace or Feature Dataset