**Create ArcGIS Desktop Geoprocessing Scripts using Arcpy**

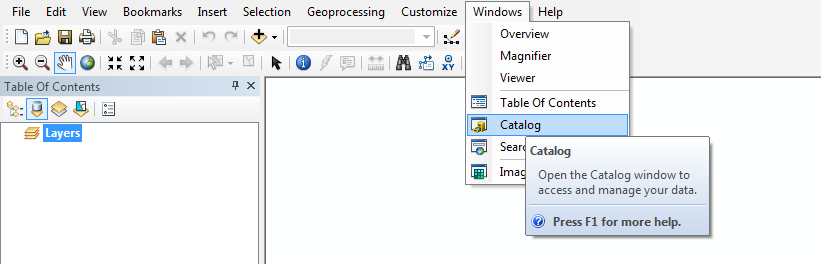
Many geoprocessing tasks can be automated through scripts. Using a Python script, you will automate the process of running an intersect for points falling within polygon features, and then find the frequency of the points that fell within each polygon features.

In this exercise, you will perform the following tasks:

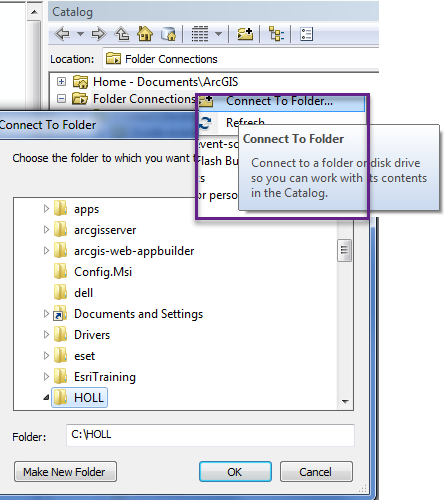
* Run Spatial Join geoprocessing tool in Python window.
* Run Frequency geoprocessing tool in Python window.
* Export Python window code to Python script.

**STEP 1: INTERSECT SCHOOLS WITH CONSTITUENCY IN PYTHON WINDOW**

In this step, you will locate and access a geoprocessing tool from the ArcMap Search window, and then run the geoprocessing tool in the Python window. After the tool runs successfully, you will move on to the next step to find the frequency of how many schools(point layer) are found in the constituency(polygon layer).

1. Start **ArcMap** with a new blank map.
2. Open the **Catalog** panel, if it is not open. 
3. Create a folder connection to **C:\HOLL\Python\Data**

Hint: Right Click Folder Connections and choose **Connect To Folder**



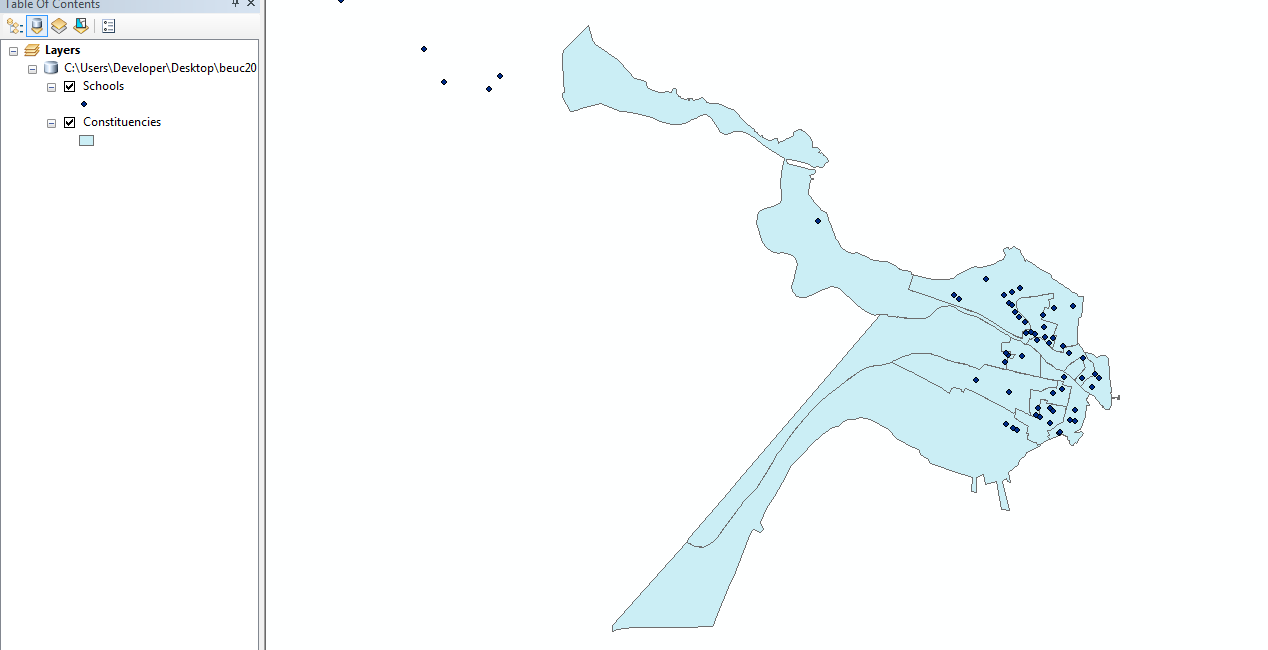
1. Set **Belize.gdb** as default geodatabase.

Hint: Right Click and choose **Make Default Geodatabase**

It will add a home icon to the geodatabase.

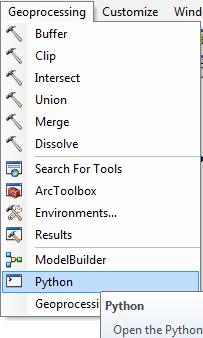


1. In the Catalog window, expand the **C:\HOLL\Python\Data** folder, and then drag **Belize.mxd** into the map. You can now see in the **Table of Contents** all the Constituencies (Polygon) and Schools (Point) layers we will use in this exercise.



1. Open the Python window and dock it to the bottom of the ArcMap display.

Hint: From the Standard toolbar, click the Python window button . Alternately, from the **Geoprocessing** menu, select **Python**.



1. In the Python window, let’s get into the habit of importing arcpy as external IDE requires it. Type,

**import arcpy**

and hit Enter on your keyboard.

1. Next, Set the current workspace environment setting to the Belize geodatabase:

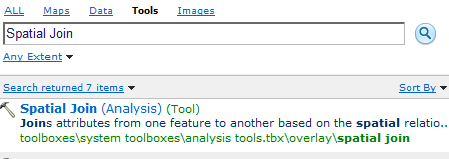
Type

**arcpy.env.workspace = r'C:\HOLL\Python\Data\Belize.gdb'** and then press Enter on your keyboard.

1. **INTERSECT SCHOOL WITH CONSTITUENCY**

We now want to use the **Spatial Join (Analysis)** tool to help intersect which schools falls under what constituency.

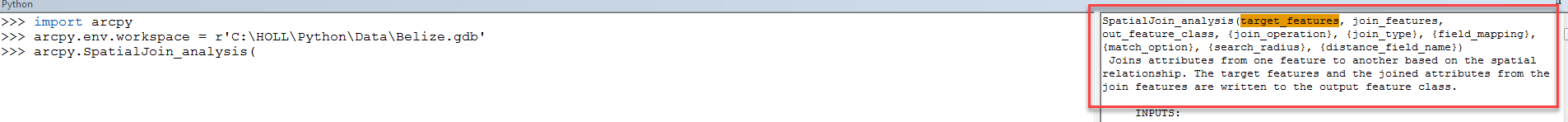
1. Open the **Search** window and click **Tools**.
2. In the Search window, type **Spatial Join**, and then press Enter.



The Search window displays the Spatial Join tools, and other matching items.

1. From the Search window, drag the **Spatial Join (Analysis)** tool into the Python window.

Notice that code complete displays a list of the layers from the table of contents for you to choose from.



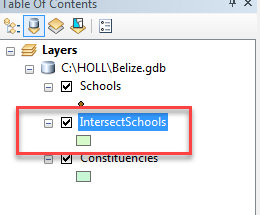
1. From the **Table of Contents**, Drag and drop **'Constituencies',** (add comma after) to enter the **target\_features** parameter value.

Next you will add the **join\_features** parameter value.

1. Drag **'Schools',** (make sure to add the comma)
2. For the **out\_feature\_class** parameter value, type **'IntersectSchools'**,
3. The **join\_operation** value will be **'JOIN\_ONE\_TO\_MANY',**
4. For **join\_type** , type **'KEEP\_ALL',**
5. For the **field\_mapping** parameter value will remain empty, type the quotes and comma **' '**,
6. Type **'INTERSECT'** for the **match\_option** parameter value
7. Enter a closing parenthesis.
8. In the Python window, ensure cursor is outside the closing parenthesis, and then press Enter on your keyboard.

Note, after execution, you can see the status of the execution on the bottom right in ArcMap.

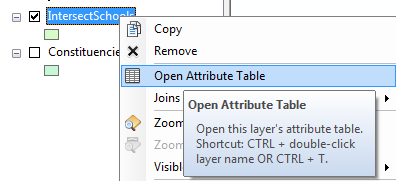




When successfully executed, the Spatial Join tool will return a result that will be added to both the Table of Contents and default geodatabase.

If the results are not successful, return to the Python window, recall the last command (by pressing the Up arrow on the keyboard), change the incorrect parameters, and then rerun the tool.

1. You can go to the **IntersectSchools** layer in the **Table of Contents**, and right click and then choose **Open Attribute Table,** here you can see the intersect table which includes attributes from both Schools and Constituency layers.



**III. FIND SCHOOL FREQUENCY**

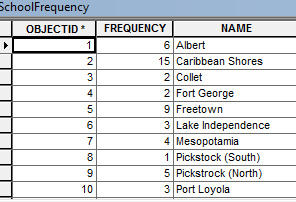
Next, we want to find the frequency for schools that are found in each constituency.

1. From the Search window, choose **Tools** and type **Frequency** then drag the **Frequency (Analysis)** tool into the Python window.
2. For the **in\_table** value, type **'IntersectSchools',** (make sure to add the comma)
3. Then, for the **out\_table** parameter value, type **'SchoolFrequency'**,
4. The **frequency\_fields** value will be **'NAME'**
5. Enter a closing parenthesis.
6. In the Python window, ensure cursor is outside the closing parenthesis, and then press Enter on your keyboard.

When executed, the Frequency tool will return a table result that will be added in both the Table of Contents and to the default geodatabase.

Congratulations! You have just written your first python script using geoprocessing tools!

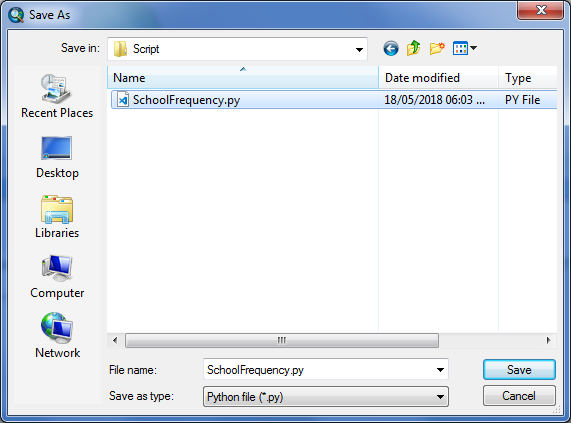
Your **SchoolFrequency** table should look similar to the below result table:



**Optional Steps to make your script become a tool in your toolbox in ArcMap**

Right click the Python Editor and choose **Save As**.

Next, save the document as **SchoolFrequency.py**



Note to add this script to the toolbox, we will need to update the script to accept parameters dynamically. Feel free to go to the Script folder under **C:\HOLL\Python\** folder and use the **FrequencyTool.py** Script already there to proceed with the following steps.

import arcpy #import arcgis library

#Dynamic Parameters

arcpy.env.workspace = r'%s' % arcpy.GetParameterAsText(0) #accept gdb workspace parameter

Polygon = r'%s' % arcpy.GetParameterAsText(1) #polygon layer parameter

Point = r'%s' % arcpy.GetParameterAsText(2) #point layer parameter

FrequencyField = r'%s' % arcpy.GetParameterAsText(3) #

#Manually assign parameter values

IntersectLayer = "IntersectLayer" #default intersect layer

PointFrequency = "PointFrequency" #default point frequecy table

#Show message in console of parameters

arcpy.AddMessage(arcpy.env.workspace)

arcpy.AddMessage(Polygon)

arcpy.AddMessage(Point)

arcpy.AddMessage(IntersectLayer)

arcpy.AddMessage(PointFrequency)

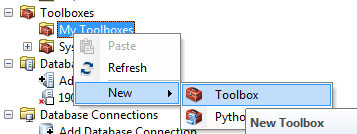
arcpy.AddMessage(FrequencyField)

#utilize geoprocessing tools

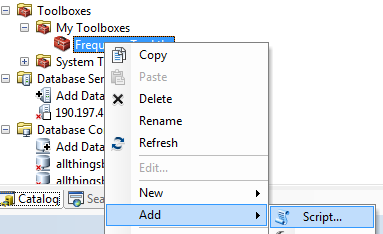
arcpy.SpatialJoin\_analysis(Polygon, Point, IntersectLayer,"JOIN\_ONE\_TO\_MANY", "KEEP\_ALL","", "INTERSECT" )

arcpy.Frequency\_analysis(IntersectLayer, PointFrequency, FrequencyField)

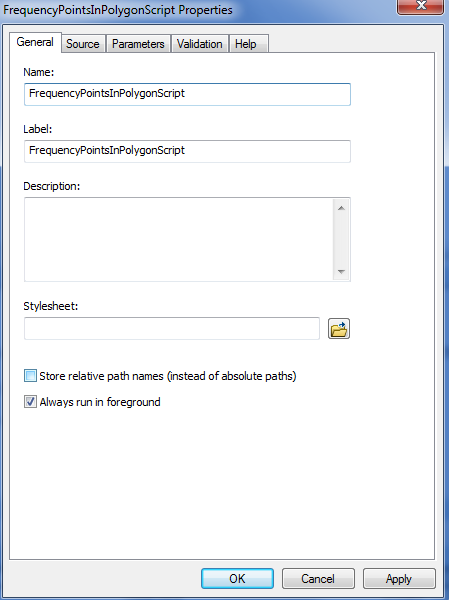
In the Catalog window, choose **My Toolboxes > Toolbox > New Toolbox and** Rename as **FrequencyTool.tbx**



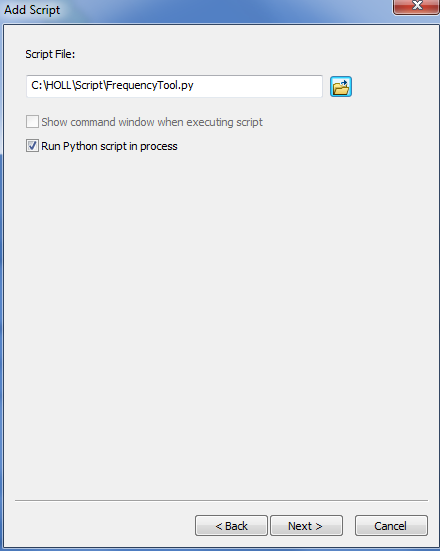
Right click the new toolbox created and click **Add > Script**



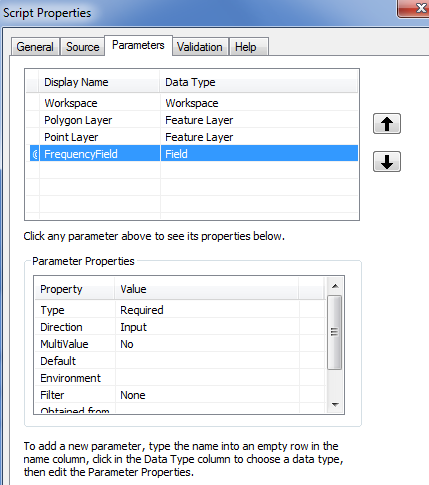
Name your tool script **FrequencyPointsInPolygonScript** as seen below and click **OK**



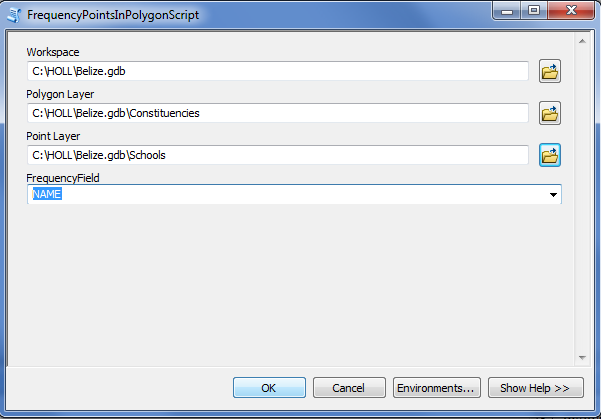
Next, browse to **Add Script** and click **Next**



Next, Add the following input parameters and click OK.



Next, Double click the script under the newly created toolbox, a window will open to prompt the dynamic fields.



Congratulations! You have a Frequency toolbox script you can utilize for any future projects.

Just enter the fields for the polygon and point layer that you would like to find the frequency for and this tool will execute the frequency table.

