**Lab 8 - Cost Distance, Region Groups, more Model Builder & Python**

**Learning Objective**

This lab is designed to further strengthen your understanding and ability to apply local, focal, and zonal operations to solve problems and comfortability with ModelBuilder, this time in the context of delineating sea level rise. In part 1, you will walk through 2 methods of using the point and click interface to accomplish your analysis. In part 2, you will use ModelBuilder to accomplish this same task, and in part 3 you will see how you can import your own python analysis into ArcGIS Pro as tools.

**What you need to submit**

**Lab 8: Answer Sheet**  
Name:  
**Question 1**: How many pixels are inundated?

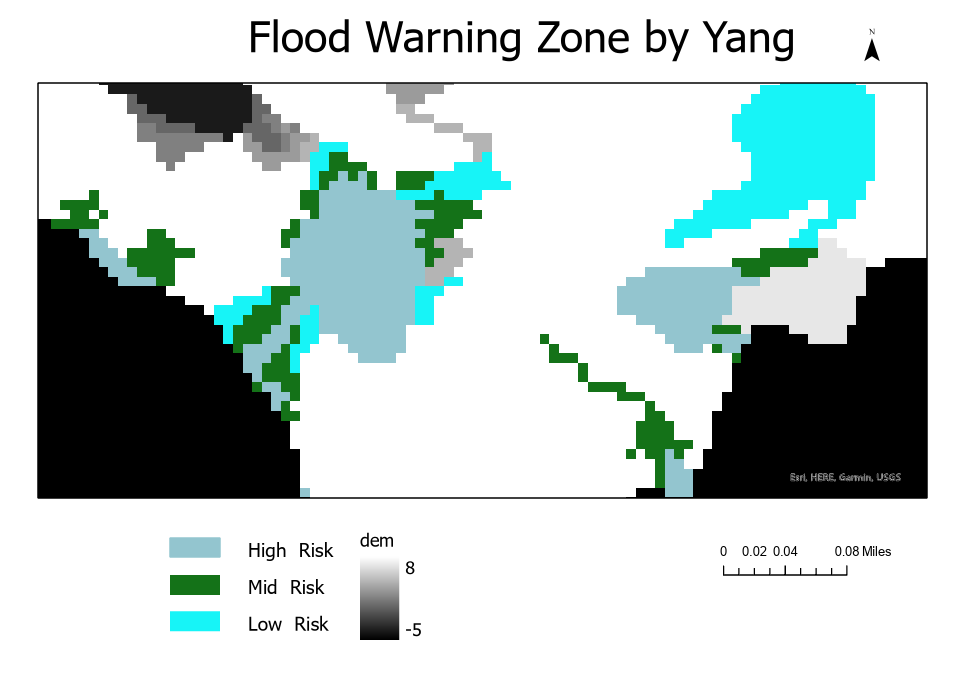
**Question 2**: While you’re building your models with model builder, you create several variables called model parameters that are shown with P letter next to their variables. What happens if you didn’t create those model parameters? What is the benefit of having the model parameters?

**Question 3**: In part 02 of the lab, list all the output layers after you Run Entire Model in the ‘InundationByRegionGroup’ model builder window.

**Question 4**: In part 02, how many cells does the final output have the value of ‘1’?

**Question 5**: Which of the two methods of delineating sea level rise, region group or cost distance, was the more efficient means of performing this analysis? Why do you think this is?

**Question 6**: Submit a coherent map of the output. Use mine below as an example. (give the sea rise level from 1 to 3 and try)



**Question 7**: This time, double click the InundationByRegionGroupPython script tool in part 03. When the model tool dialog box is opened, fill the parameter boxes with the right layer names. Click OK to run the tool. How many output layers do you have? List them.

**Question 8**: In part 03 how many cells does the final output have the value of ‘1’?

**Question 9**: If the Sea Level Rise is set to 4, how many cells does the final output have the value of ‘1’?

**Question 10:** Attach the result of your Part 4.

**Materials**

|  |  |  |
| --- | --- | --- |
| Relevant part | Data Name | Description |
| Part 1-3 | DEM | Raster dataset of the elevation of the area |
| Part 1-3 | Ocean | Raster dataset of the land cover types (Land & Ocean) over the area |
| Part 3 | SeaLevelRiseInundation.py | Sea Level Rise inundation delineation using region group and zonal max |
| Part 4 | elevation | Raster dataset of the elevation of the area |
| Part 4 | landuse | Raster dataset of the landuse types over the area |
| Part 4 | rec\_sites | Feature dataset displaying point locations of recreation sites |
| Part 4 | schools | Feature dataset displaying point locations of existing schools |

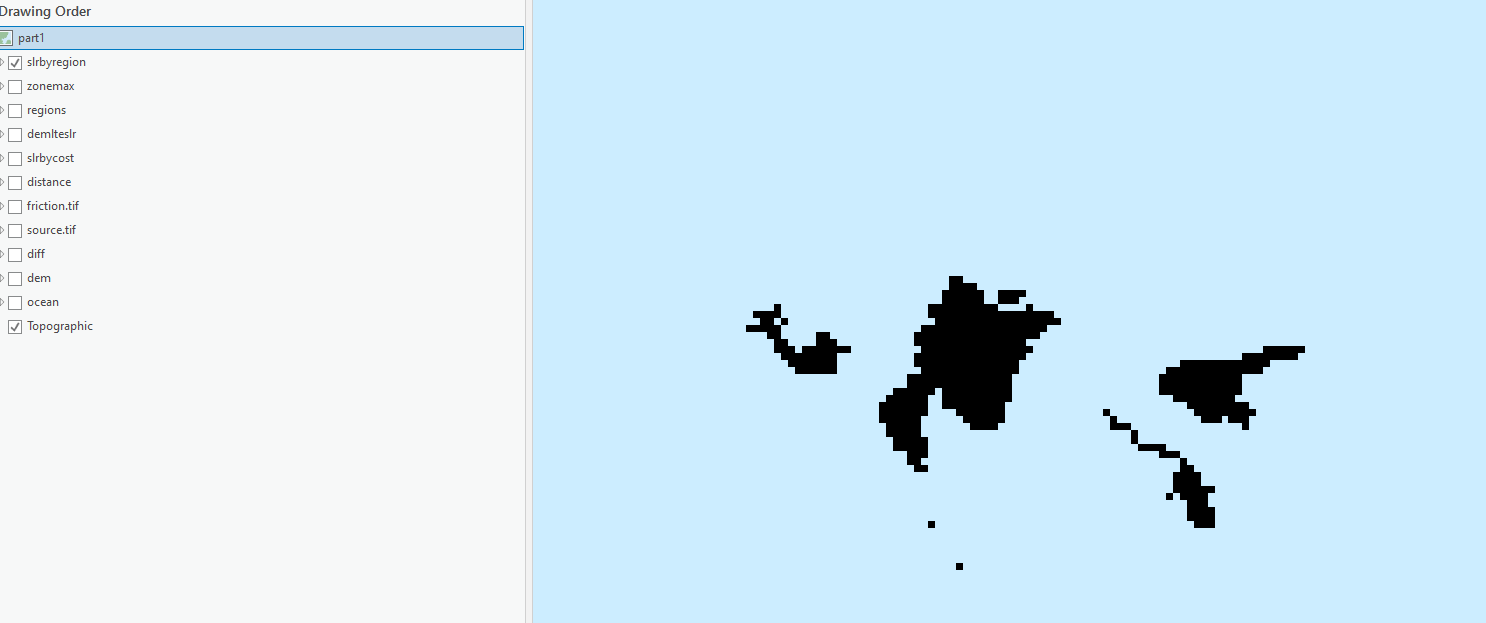
# ****Part 1: Delineating Sea Level Rise Inundation with Region Group****

As you saw in class, it is possible to take two different approaches to map sea level rise in GIS. We can use cost distance to calculate the rise from an ocean source. Here you need to use region group to get the regions of inundation which include an ocean source.

Unzip the lab 8 folder into your one drive, in it you will find folders already set up for part1, part2, part3, and part4. Go ahead and pick a means of delineating sea level rise from below, either cost distance or region groups, and perform a 2-meter sea level rise analysis.

|  |  |
| --- | --- |
| **Cost Distance** | **Region Group** |
| 1. Set the oceans to a one value source | 1. Create a raster of all the areas less than the projected sea level rise. |
| *a. Use the Set null tool on the ocean raster* |
| *b. "Value" = 0* | *a. Hint: Use the set null tool to replace all values > sea level rise to null.  Input raster is dem.* |
| *c. Your false or constant raster can be either the ocean raster or a value of 1* |
| *b. Otherwise, set the value to 1* |
| *d. call the output “source.tif”* | *c. Call the output demlteslr* |
| 2. Set all elevations in the DEM which are above the sea level rise threshold | 2. Group the resulting regions using region group |
| *a. Use 8 connectivity and group using the within method.* |
| *a. Set null tool again, this time on elevation* | *b. Call the output regions* |
| *b. "Value" > %Sea Level Rise% (in the manual case, 2)* | 3. Use zonal statistics to identify the regions which contain an ocean source |
| *c. Set the false raster to 1 (Friction should be the same everywhere otherwise)* |
| *a. We want the value of oceans (input value raster) using the regions as the zone data, and the maximum statistics.* |
| *d. Call the output “friction.tif”* |
| 3. Create the cost distance surface | *b. Call the output zonemax* |
| *a. Cost distance tool…* | 4. We then want to remove the oceans from our calculation |
| *b. Cost is “friction” source is “source”* | *a. Hint: raster calculator (%zonemax%" - "%ocean%").* |
| *c. Call the output "distance.tif"* | *b. Call the output diff* |
| 4. Remove the source cells to create a raster of just inundation | 5. Finally, set everything less than or equal to 0 to null so that we have just the inundated pixels |
|  |
| *a. Set null* | *a. Hint: Set null tool again* |  |
| *b. “Value” = 0* | *b. Call the output SLRbyRegion* |  |
| *c. Again, set the value to 1 otherwise* |  |  |
| *d. Call the output “SLRbyCost.tif”* |  |  |

Note, the result of the two approaches should be the same. Answer question 1.



# ****Part 2: Delineating Sea Level Rise Inundation with Model Builder****

In ArcGIS we can use ModelBuilder to build a visual model that represents an analysis work flow. A ModelBuilder model typically consists of a set of data connected by a series of GIS operations. In this part we will be using ModelBuilder to create a model to delineate sea level rise inundation. Using a ModelBuilder model is helpful in situations where the model can be applied to numerous datasets, as we saw with the last lab. Here you will use model builder to create both variants of the SLR workflow, one for Region Groups and one for Cost Distance.

## **Set up the folders for the lab**

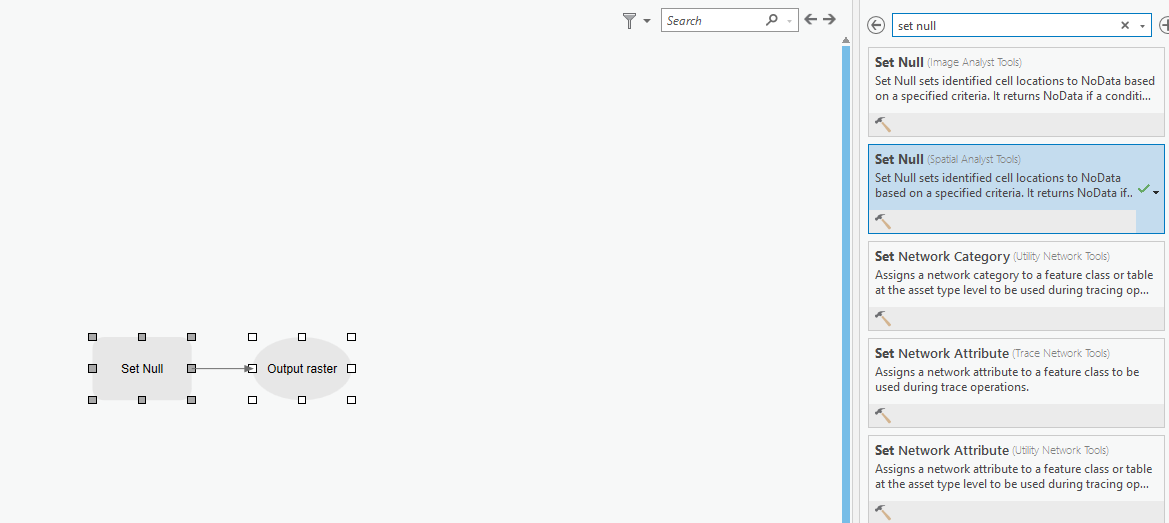
* In your lab 8 part 2 folder, create a new File Geodatabase.
  + R-Click > New > File Geodatabase
  + This will be your new default workspace
* Add in the DEM and Ocean to the ArcMap, and examine them. When done, remove them so you have a blank map.
* It will also be helpful if you set overwrite to on
  + Geoprocessing > Options > Overwrite the outputs…

## **Creating a Toolbox and a Model**

* R-click on your lab 8 part 2 folder and create a **new toolbox**. Name the new toolbox InundationTools.tbx.
* R-click on the toolbox and create a **new model**.
* Rename the new Model (**ModelBuilder | Model| Properties|Label**) to InundationByRegionGroup.
* Under the Environments tab, make the **Extent** the same as the DEM raster in your data folder, and the **Cell Size** the same as DEM as well (they both have the same extent and cell size, if you check the data’s properties).
* Click OK on the Environment Settings.

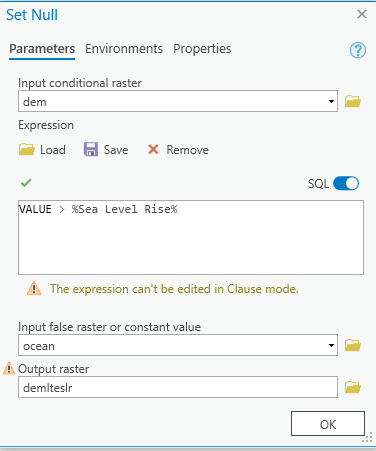
## **Setting up the Workflow & Building the InundationByRegionGroup Model**

You will create a workflow diagram in the Model window that will show all the steps and operations/tools needed to delineate the sea level change. We start the process by dragging tools from ArcToolbox into the Model window to perform a specific operation.

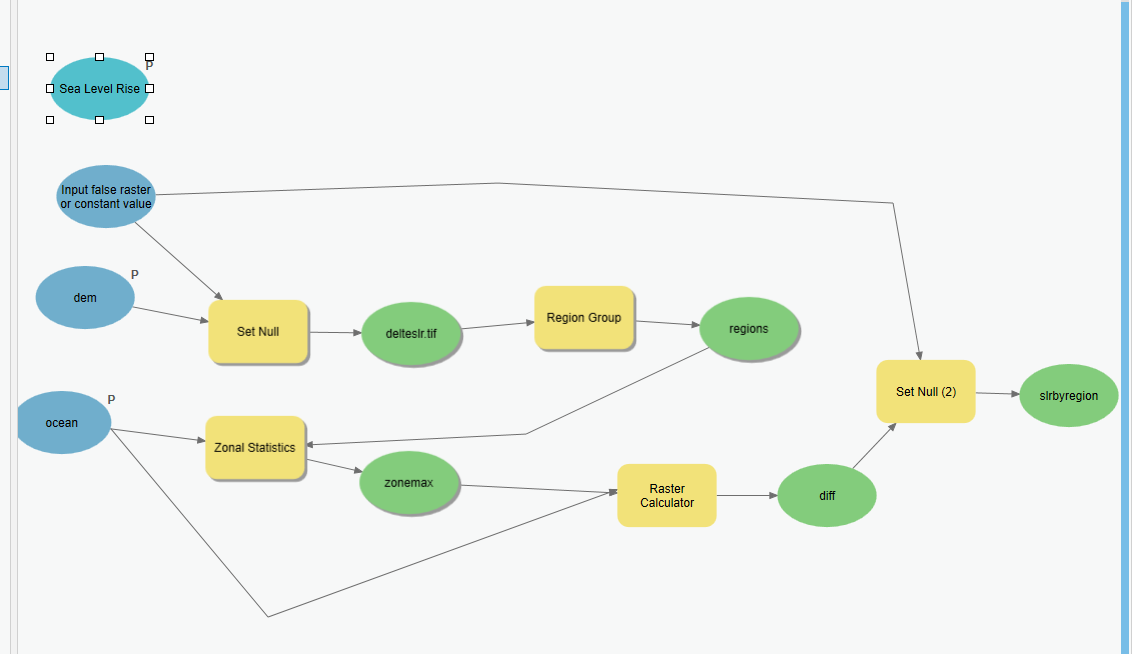
* Open **Tools** and set up your model window and Tools window so that you can see both Tools and the model window at the same time.
* Add the **Set Null** tool by finding it under Spatial Analyst and then dragging it onto the model window.
* 
* Before you set the options for the Set Null tool, create a variable which data type is double. Go to **Diagram | insert | Variable,** Select the data type as Double. R-Click and rename it ‘Sea Level Rise’, double click it and give the value of ‘2’ to the variable, and set it as a Model Parameter by R-Click and check Model Parameter option. The letter P appears beside the variable, indicating it is a model parameter.



* Double click on the Set Null tool. The Set Null dialog will open, and it should be familiar.
* We can reuse the false raster we used in the first set null.
* Set the input conditional raster to DEM, and set the output to *demlteslr (sometimes you may have errors without .tif, what you can do is to add the .tif)* and make sure it is saved in the appropriate place. As for the expression, type in *“Value” > %Sea Level Rise%*. Click OK. Make sure that the out put folder is your part2 folder.

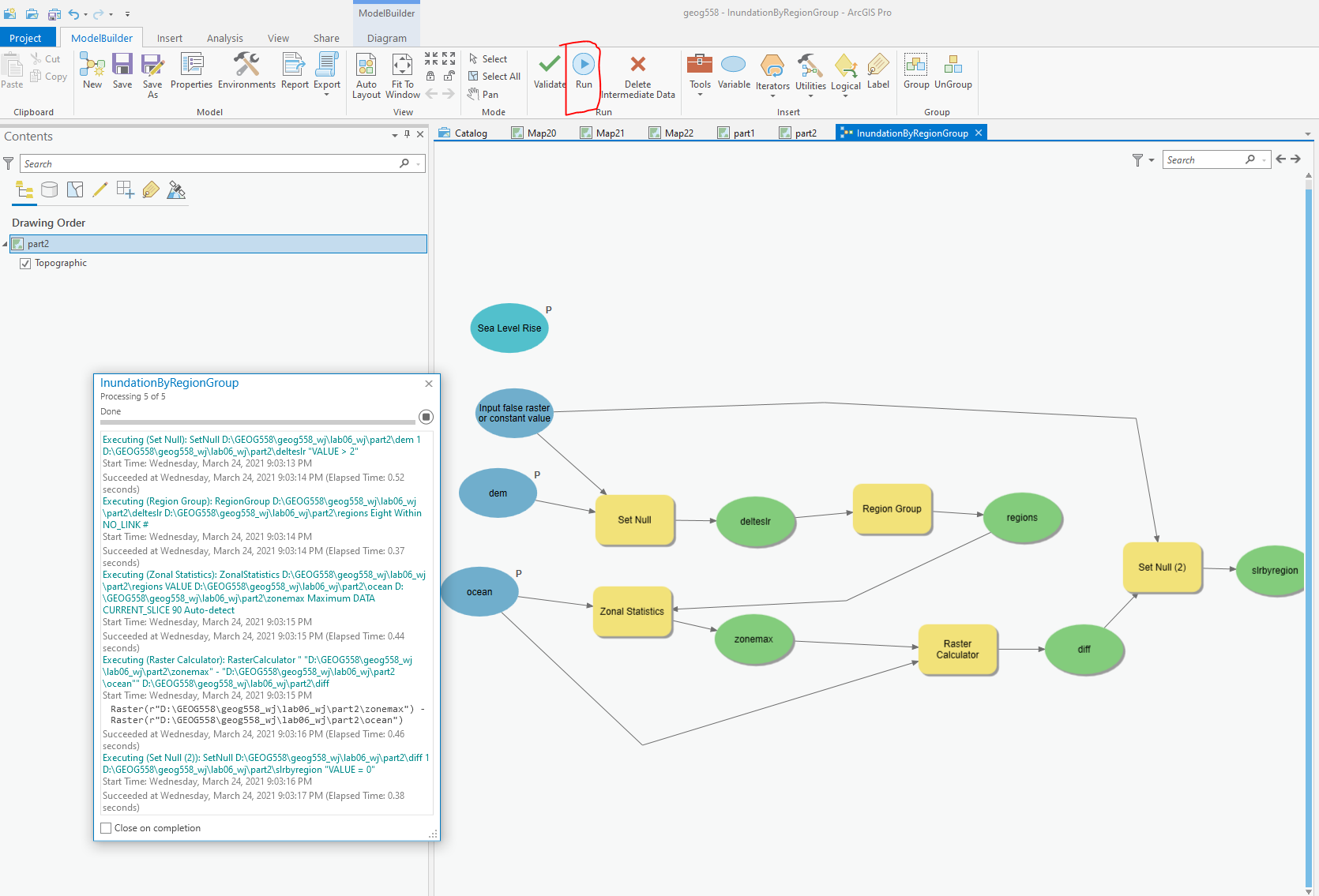


* **Right-click** the Set Null and select **Run** to test your model.
* Add, examine, and then remove the created Raster from the part2 data frame.
* Add a **Region Group tool** (Spatial Analyst Tools ) to your model.
  + Set *demlteslr* as its input raster
  + Set *regions* as an output.
  + The number of neighboring cells to use is eight and use within for zone grouping method.
  + Uncheck not to add link field to output and…
  + click OK to finish the Region Group dialog.
* **Right-click** the Region Group tool and select **Run** to test your model.
* Add examine, and then remove the created Raster from the part2 data frame.
* Add a **Zonal Statistics tool** (Spatial Analyst Tools ) to your model.
  + Set *regions* as its input
  + Set *VALUE* as a zone field
  + Use *zonemax* as an output.
  + Set *Ocean* as an input value raster.
  + Use maximum for statistics type, then click OK.
* Set *Ocean* as a **model parameter**.
* **Right-click** the Zonal Statistics and select **Run** to test your model.
* Add examine, and then remove the created Raster from the ArcMap window
* We now need to use the **Raster Calculator** to calculate the difference between the *zonemax* and the *Ocean*.
  + Hint: *(“%zonemax%” - “%Ocean%”)*.
  + Save the output as *diff*.
* **Right-click** the Raster Calculator and select **Run** to test your model.
* We are going to add a **Set Null** again to get our final output Inundation. Create another **Double model variable**, to define the input false raster or constant value and give it a value ‘1.’ Set *diff* as the tool’s input and type in the proper expression to query the values less and equal to zero.
* **Right-click** the Set Null and select **Run** to test your model.
* The final layout should look like so:



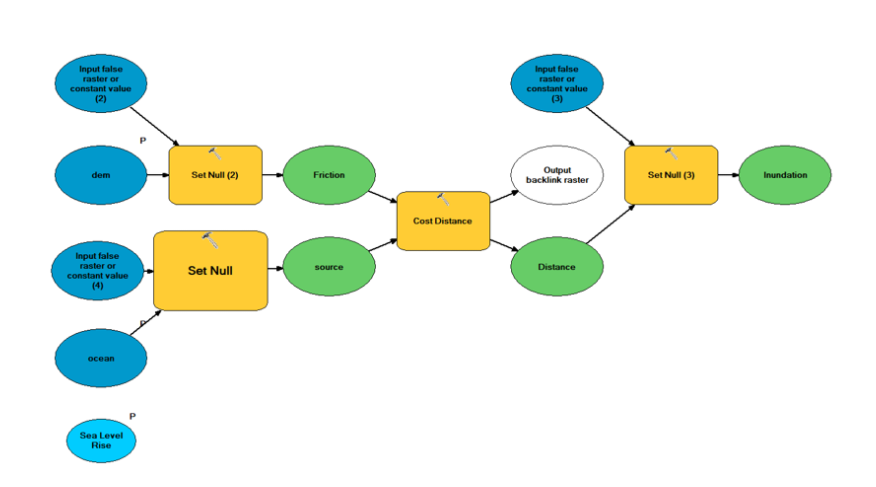
## **Saving and testing your whole model**

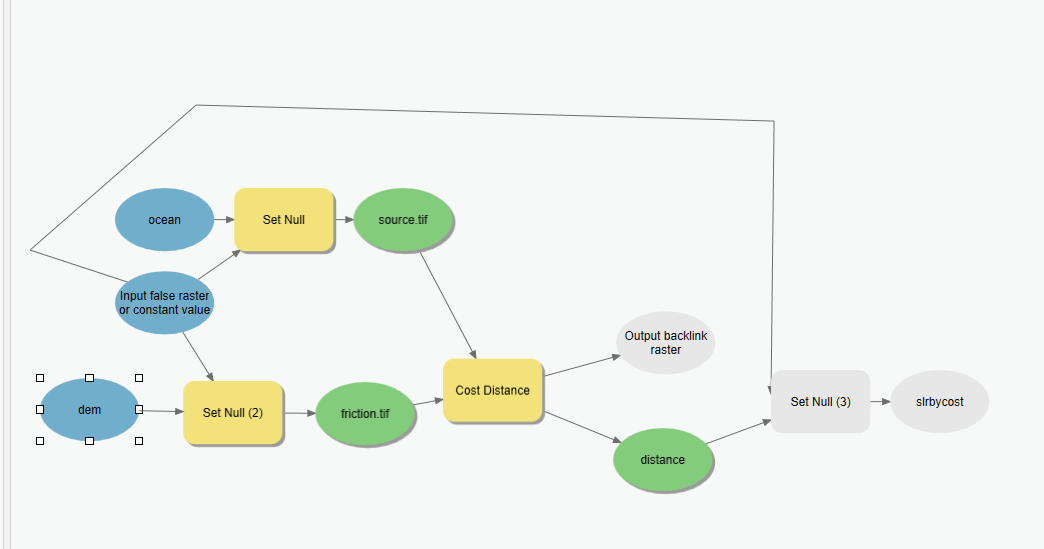
* Save your model.
* While you’re building your model, you can check if each tool (so far, we created 5 tools; Set Null, Set Null(2), Region Group, Zonal Statistics, and Raster Calculator for this model) you created works fine by right-clicking and running it (select Run). This time, let’s check if whole your model runs in one go.
* Close the model.
* After all tools are correctly established in the model, delete all of the output files (delteslr, regions, zonemax, diff, inundation) from your output folder.
* Run your model using the appropriate inputs.



## **Now recreate the cost distance approach**

Create a new model using the cost distance approach to delineating sea level rise. Your final model should look something like the following two images:





Answer question 2-5.

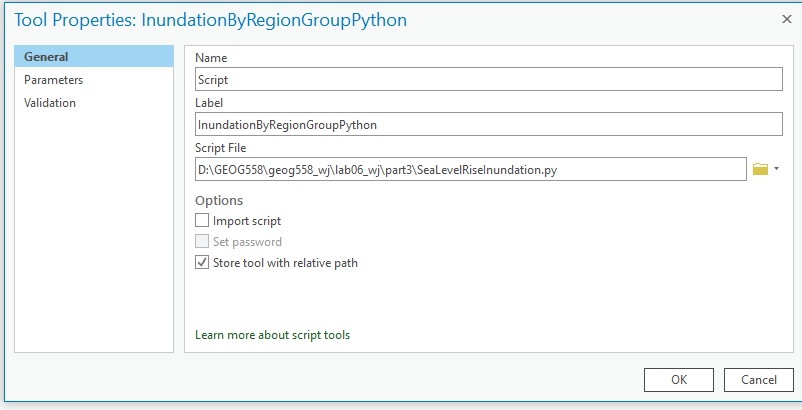
# ****Part 3: Sea Level Rise Inundation Delineation with Python Scripting****

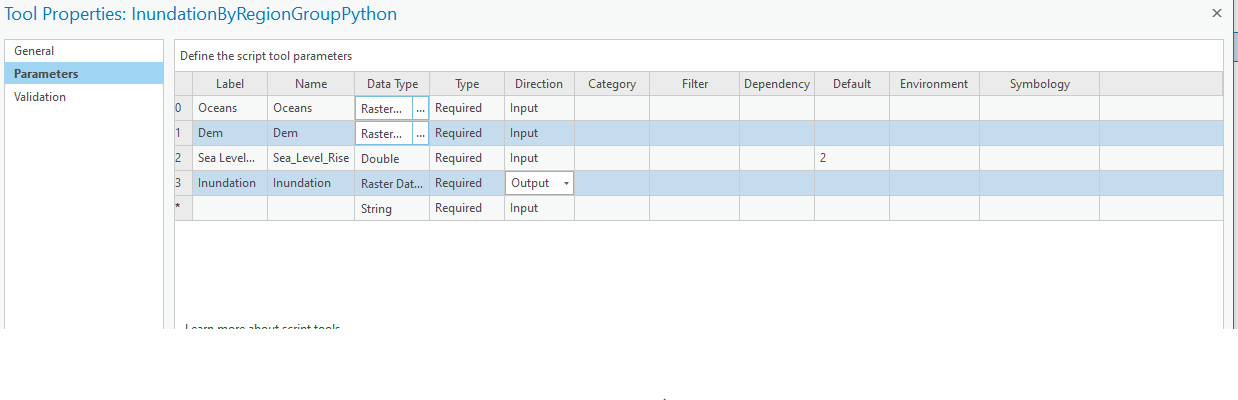
In this part of the lab we will be using ArcGIS Python Command Window to do the same analysis we did in the first part. The main purpose is to show how such a model/analysis can be implemented by using a python script tool. We, therefore, will practice how to create ArcGIS Python tool using a python script.

## **Creating a script tool**

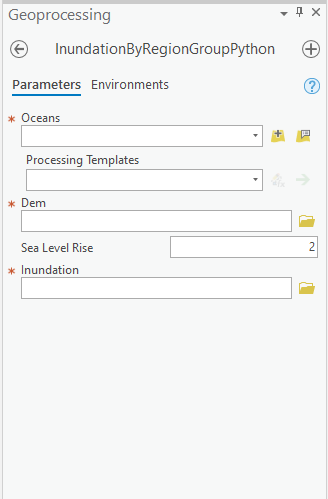
* Within ArcGIS, you can create a script tool inside a toolbox. A script tool is like any other tool - it can be opened and executed from the tool dialog box, used in models and the **Python** window, and called from other scripts and script tools.
* To create a script tool, you need three things; a script, a custom toolbox, and a precise definition of the parameters of your script.

## **Creating a custom toolbox**

* You can create your own toolbox called a custom toolbox and add tools and toolsets to it. Script and model tools that you create should be stored in a custom toolbox.
* Open ArcCatalog. In the Catalog Tree, navigate to the folder where you want the toolbox to be created.
* Right-click the folder and click **New > Toolbox**. Give the toolbox a new name InundationToolbox.tbx.
* To create a script tool, right-click your custom toolbox and click **New > Script**. This opens the **New Script window** that takes you step by step through the process of creating a script tool.
* Give a new script a Label InundationByRegionGroupPython. (It is optional to change both label and name).
* Add a script file by specifying the location of the file under part3 folder. SeaLevelRiseInundation.py.
* 
* Then, add parameters by typing the name and selecting the data type as described in the table below. As for the property, use the default options unless specifically mentioned in the table below.



* After completing the steps, your toolbox will contain a new script tool. You can always modify properties (such as parameter names and data types) of this script tool by right-clicking the script tool and choosing **Properties**.
* Double click the script file you just added. The *InundationByRegionGroupPython* tool dialog will appear and it should look like this image below.



* Input *ocean* & *dem* for the first two input layers. Use the default value for Sea Level Rise. Save *Inundation* in your part3 folder as an output layer. Click OK to run the tool. Answer question 6-9.

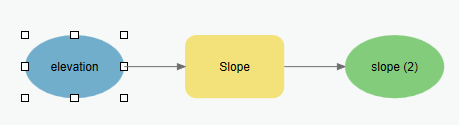
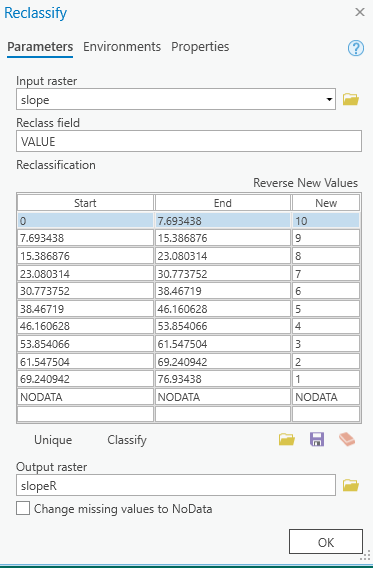
# **Part 4: School Sighting (Site Suitability Analysis)**

## **Creating a Toolbox and a Model**

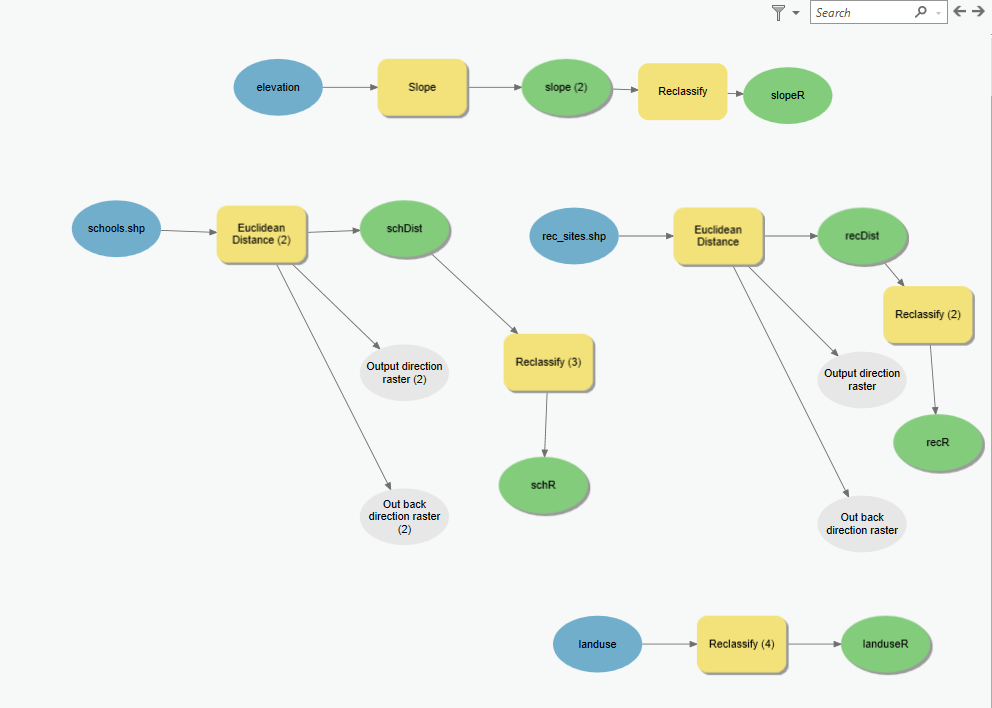
* Create a new toolbox (Lab8Tools), and a new model (SchoolSiting) in that toolbox.
  + R-click on your lab08/Part4 folder and create a new toolbox.
  + Name the new toolbox Lab8Tools.tbx.
  + R-click on the toolbox and create a new model.
  + Rename the new Model to SchoolSiting (in propertities)*.*
* Set up the Model Properties by going to **Model | Model Properties**
* Under the **Environments tab**, make the **Extent** the same as the landuse raster in your data folder, and the **Cell Size** the same as elevation.
* Click OK on the Environment Settings.

## **Setting up the Workflow**

You will create a workflow diagram in the Model window that will show all the steps and operations/tools needed to get the suitable school sites. We start the process by dragging tools from geoprocessing Tools into the Model window to perform a specific operation.

* Set up your screen so you can see both Geoprocessing Tools Window and the model window at the same time.
* Add the slope tool by finding it under Tools (**Spatial Analyst Tools | Surface**) and then dragging it onto the model window.
* Double click on the slope tool. The slope dialog will open, and will look familiar.
* Set the input to the elevation raster (from your data folder), and set the output to slope and make sure it is saved in the output folder in the Lab 8 folder. Leave other parameters to their defaults.
* Click Run. You will see the diagram objects become colored. You now have a very simple model.
* 
* Next, add the Euclidean (straight line) distance tool (**Spatial Analyst Tools | Distance | Euclidean Distance**) to create a distance layer from *rec\_sites*.
* Set the Input to *rec\_sites* (from your data folder), the Output to *recDist* in your output folder, and the Cell Size to 30. You don’t need a Maximum distance or a direction Raster.
* Click OK. Notice that an empty diagram object is there for direction raster. Since it is blank, it will be ignored, so we will leave it alone.
* Add and set up another Euclidean distance tool to calculate distance from schools (input should be school.shp, call the output schDist).
* Save your Model (**ModelBuilder | Save**).
* Click the **Run** button. To make sure that these three layers are calculated correctly and saved in your output folder.
* Add a Reclassify tool (**Spatial Analyst Tools | Reclassify**) to the Model window to reclassify the slope raster.
* Double click on the **Reclassify tool** in the Model window. Select *slope* as the input layer.
  + This time, select the input layer from the dropdown in this dialog window, which shows the layers that are part of the model (We do this so that it will perform this next step from what we calculated in the previous one. In other words, it maintains a flow within the model).
* In order to create the new classes for the slope layer, click on the **Classify button**, number of classes is 10.
* It is preferable that the new school site be located on relatively flat ground. You will reclassify the slope layer, giving a value of 10 to the most suitable slopes (those with the lowest angle of slope) and 1 to the least suitable slopes (those with the steepest angle of slope). Click on the **Reverse New Values** button so the lowest slopes get a value of ten and the steepest slopes a value of 1.
* Name the output raster layer as *slopeR*.
* 
* Add a new **Reclassify tool** and double click on it.
* Select the *RecDist* layer as the input raster from the dropdown.
* Reclassify distance to recreation sites raster (*recDist*) into 10 classes with new values from 1 to 10 using Equal Interval classification method. The school should be located near recreational facilities. You will reclassify this dataset, giving a value of 10 to areas closest to recreation sites (the most suitable locations), giving a value of 1 to areas far from recreation sites (the least suitable locations), and ranking the values in between.
* Name the output raster layer as *recR* and click OK.
* Add a third **Reclassify tool** and double click on it.
* Select the *SchDist* layer as the input raster from the dropdown.
* Reclassify distance to schools raster (*schD*) into 10 classes with new values from 1 to 10 using Equal Interval classification method. It is necessary to locate the new school away from existing schools in order to avoid encroaching on their catchment areas. You will reclassify the distance to schools layer, giving a value of 10 to areas away from existing schools (the most suitable locations), giving a value of 1 to areas near existing schools (least suitable locations), and ranking the values in between.
* Name the output raster layer *schR* and click OK.
* Save your Model.
* Add and set up a fourth **Reclassify tool** in order to reclassify the *landuse* raster.
  + Select ‘LANDUSE’ as the ‘Reclass field’ using the dropdown menu.
  + Use the following table to assign the new values to each of the land use categories, click unique:

|  |  |
| --- | --- |
| **Landuse type** | **Score** |
| Agriculture | 10 |
| Barren land | 6 |
| Brush/Transitional | 5 |
| Built up | 3 |
| Forest | 4 |
| Water | NoData |
| Wetlands | NoData |

* Call the output *landuseR*.
* Click OK and Save your Model.
* After all four reclassify tools are correctly established in the model, click **Run** again and check your output folder to make sure everything was calculated correctly.
* 

## **Calculating a Suitability Raster**

* With all of our standardized layers, we are ready to calculate our suitability raster. Add a **Raster Calculator** Tool (Spatial Analyst Tools | Raster Calculator) to the Model window.
* Double click on this tool to set up the parameters.
  + Create an expression to weight each factor using the table below.

|  |  |
| --- | --- |
| **Layer** | **Weight** |
| Reclass of distance to recreation sites (recR) | 0.5 |
| Reclass of distance to schools (schR) | 0.25 |
| Reclass of landuse (landuseR) | 0.125 |
| Reclass of slope (slopeR) | 0.125 |

\* Check your expression against the image below. Save it to your lab#8 part4 folder.

