

Aaron Marks
Geog 485
Lesson 3

Loops/Conditionals/Re-writing Code

Summary

This lesson dealt with implementing loops and conditional statements into our previous customization code. The use of loops simplifies our code by removing redundant lines of code of similar actions. Loops are also useful doing the same thing to multiple parts of the project, such as expanding all of the data frames in the map. The number of data frames in each map varies yet these loops will still work if setup correctly. Conditionals allow for one thing or another to happen depending on the current state of something, whether it is itself or another object.

The two exercises were relatively easy after writing out what had to be done step by step and following some of the practice examples for correct syntax. I could have written the second part differently by combining a few lines but sometimes simple isn't always best as it becomes harder to follow to someone else reading the code.

Lesson 2

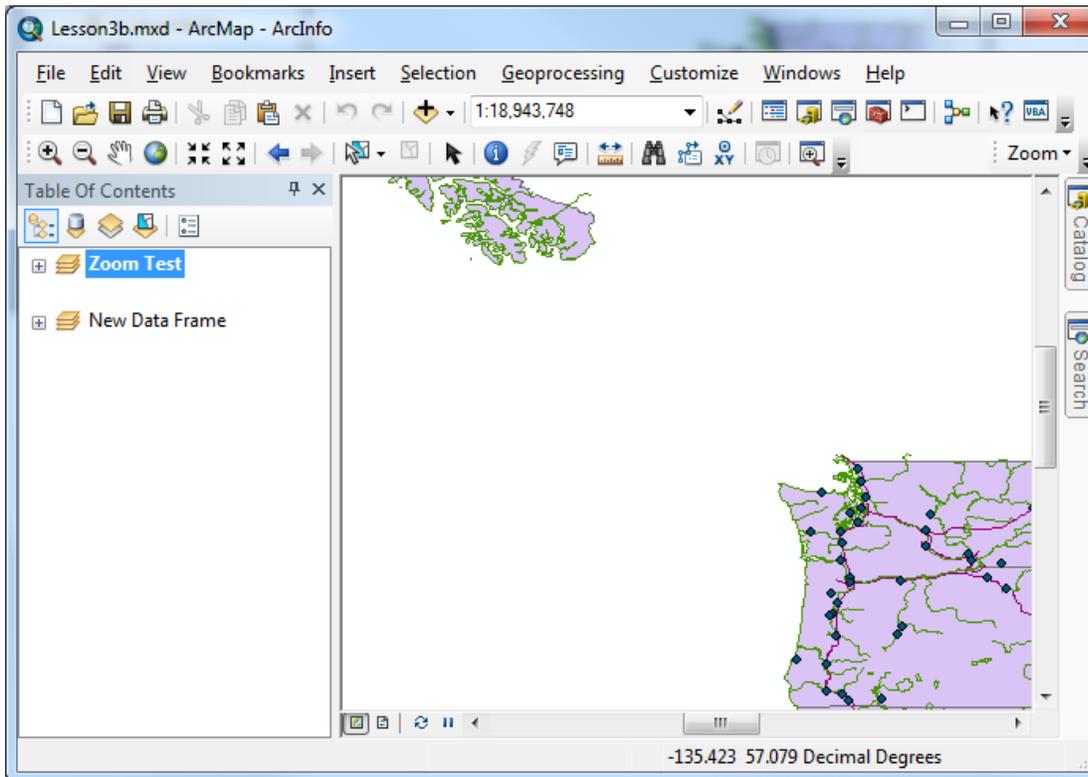
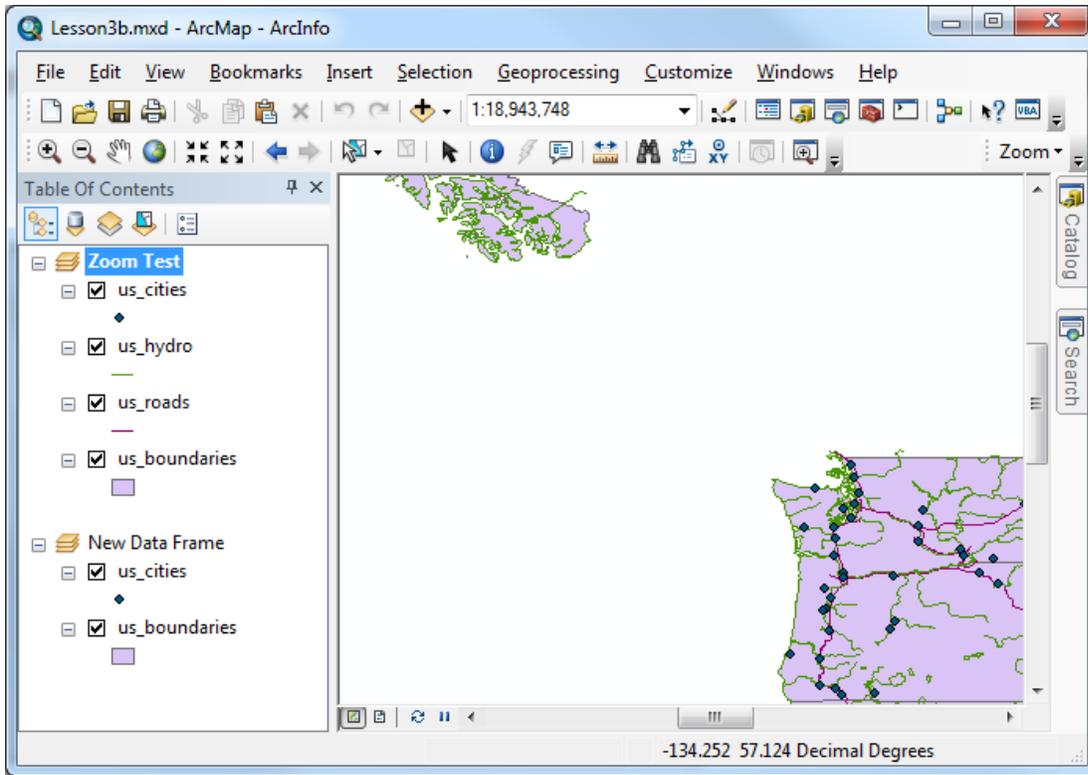
The first step to creating the function was to include the code that each button contained, which was everything except for the X and Y extend modifiers. These would simply be passed onto the function when calling to it. No return is needed here.

Adding the scale option required the current scale, which would be stored, as well as the scale after zooming. This requires at least one variable for storing the old scale. Two could be used if you store the current scale in its own but it isn't required. The scale was found in the **Imap** interface and reported through the applications message method. This also took place in the function as it happens with every button.

I had some trouble with the message bar syntax initially because I forgot to set it to **Application.StatusBar** which I found out through some Google searches. The other part of the syntax was the **pStatusBar.Message(0)** part, I had left off the **(0)**. I'm assuming there can be multiple messages. The message only remains up for a split second but it does show up. On the other hand recoding my buttons to use the function was pretty easy and sure looks a lot easier to follow.

As mentioned in the last lab I could alter my program to have the buttons scale based on the current view as opposed to the full extent view. Perhaps that is what was wanted but there were no specifics in the instructions.

Screen Captures



ArcMap

Enter a String

OK

Cancel

Patemo

First Three Upper...

Paterno -> PAT

OK