

Did you know that you can use **Regular Expressions** in the ArcMap and the ArcGIS Pro Field Calculators?

## What are Regular Expressions, you ask?

Regular Expressions are a really cool way to search through text fields to find very specific bits of text. And, if you want, to change that text. Use them to clean up your attribute tables and in label expressions.

For example, suppose you need to search through a column that has well names in it, and you need to get rid of any hyphen (“-”) that is at the end of the line, but not in any other location. Or, suppose you need to change road names so that they don’t include the “ST” at the end (but you don’t want to risk messing up a name like FIRST ST).

This brief tip will show you how to use Regular Expressions in ArcMap and ArcGIS Pro.

## First, a primer on regular expressions:

Regular expressions can be accessed in Python using the re module (available in both Python 2.7 for ArcMap and Python 3 for ArcGIS Pro).

```
import re
```

The regular expression module gives you a set of character matching patterns and a few functions, like sub (substitute) that area really handy.

Here are a few of the most commonly used regular expression patterns:

Regular Expression Pattern	Explanation
^	matches the beginning of the line
\$	matches the end of the line
[0-9]	matches any digit
[A-Z]	matches any uppercase letter
* and +	repeaters, which partner with the character before them (+ is for one or more, * is for zero or more)

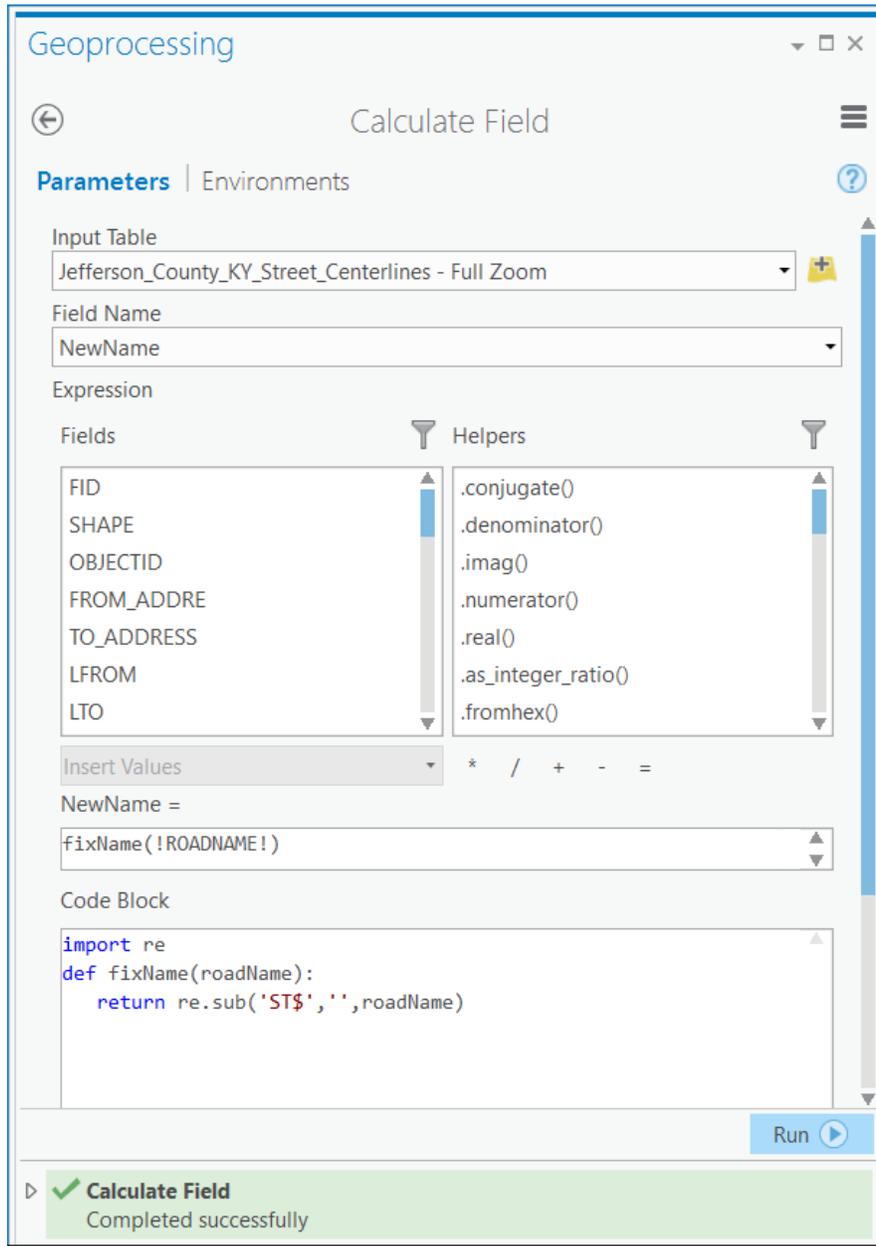
So....some examples:

^Main	matches the word Main, only if it’s at the beginning of the line
^Main\$	matches the beginning of the line, followed by the word Main, followed immediately by the end of the line (Main is the only thing on the line)
^- [0-9]	matches any line that starts with a negative number
^[A-Za-z]+\$	matches a line that contains only upper and lower case letters, nothing else
^-9+\$	matches a line that has -9, -99, -999 (or any number of 9’s really) but nothing else

Cool, huh??

## So how do you use regular expressions in ArcMap or ArcGIS Pro?

Use a Code Block in the calculator tool. Here's an example that would look at a ROADNAME column and replace any occurrence of 'ST' at the end of a line with nothing ("):



Let's break this down:

1. Open the field calculator on the field that you want to change. You may want to add a new field to the table, and open the calculator on that field (to make sure not to mess up the original field while you are practicing).
2. If you are in ArcMap, set the Expression Type to Python. In ArcGIS Pro, this is assumed.

3. In ArcMap, click the Show Code Block button. In ArcGIS Pro, scroll down to the Code Block section of the Calculator window.
4. In the Code Block, you will create a little subroutine. Since it will use the regular expressions module from Python, you'll need to type, as the first line:

```
import re
```

5. Now the subroutine...it has to be defined (give it a name and create a variable for it to use while processing each road name). In the example below, the new subroutine is named **fixName**. The new variable is named **RoadName**. Remember that Python is case sensitive, and that indention matters. `import re`

```
def fixName(roadName):  
    return re.sub(' ST$', '', roadName)
```

The `re.sub` function has 3 arguments. The first is the regular expression that you want to search for ('`ST$`'). The second is the text you want to replace the matched text with (in this example, we are replacing the matched text with an empty string, ''). The third argument is the string you want to search through. In this example, the string we want to search through is the string held in the variable `roadName` (the current `roadName` read from the table). The regular expression and the replace string get surrounded by single quotes.

```
re.sub(<regular expression>,<replace string>,<string to search>)  
re.sub(' ST$', '', roadName)
```

6. Now you are almost there. You have to tell ArcMap/ArcGIS Pro which value to write back into the attribute table by setting the right side of the `[COLUMN] = expression`. Put this in the return field:

```
fixName(!ROADNAME!)
```

7. Now run your expression!

Questions? Drop us an e-mail any time: [info@TeachMeGIS.com](mailto:info@TeachMeGIS.com).