Editing Tables with the Field Calculator

Tracy.Rijken@TeachMeGIS.com

Presentation Overview

- Getting to Know the Field Calculator
- Simple Calculations
- Function Calculations
- Advanced Calculations
Getting to Know the Field Calculator

What is it and why do we love it?

Field Calculator

Use the Field Calculator to modify records in a table
Field Calculator Expression

- Calculate values for a column
- Build an expression to calculate the value
  - Same value
  - Combination of values
- Choose parser (language)
  - VB Script
  - Python

RoyaltyRate! * !GasProduct!

Some Things to Know

Modifying Data
- Need write access to data
- Existing field
- In or out of edit session
  - Undo-able inside

Selection
- Only modifies selected records
- Modifies all records if none selected
- No checkbox option in 10.0

Results
- Geoprocessing ➔ Results
  - Error messages
  - Re-run
Fields

- Field order
  - Layer Properties order by default
  - Or sort from within Field Calculator

Simple Calculations
Static Values

Value type must match destination column type
- String
  - Quotes
  - Follow parser rules
- Number
  - No quotes

“Inner Corridor Technologies”

Copy Values

Source type must match destination column type

Fields
- !<fieldname>! in Python
- [<fieldname>] in VB Script

!field_code!

TRICK: Make the selection first

TRICK: Use to order fields
Combine Values

- Merge multiple columns
- Combine attribute values and static values
- Data types matter
- Syntax depends on parser:
  - + for Python
  - & for VB Script

\[ \text{!Township!} + "N-" + \text{!Range!} + "W" \]

Combination Tricks

<table>
<thead>
<tr>
<th>EXPRESSION</th>
<th>RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{!PointX!} + &quot;, &quot; + \text{!PointY!}</td>
<td>223.43, 5356.5</td>
</tr>
<tr>
<td>\text{!County_Name!} + &quot; County, &quot; + \text{!State!}</td>
<td>Harris County, TX</td>
</tr>
<tr>
<td>\text{!Company!} + &quot;\n&quot; + &quot;!Address!&quot;</td>
<td>TeachMeGIS 3000 Wilcrest</td>
</tr>
<tr>
<td>\text{[Company]} &amp; \text{vbNewLine} &amp; &quot;![Address]&quot;</td>
<td>TeachMeGIS 3000 Wilcrest</td>
</tr>
</tbody>
</table>
Function Calculations

Number & String

Number Functions

Basic math
- + - / *

Built-in Functions
- round()
- abs()
- max()
- min()

round( !PROD!, 1 )

TRICK
Use to convert units
Math Functions

Math module
- math.pow()
- math.sqrt()
- math.tan()
- math.ceil()
- math.trunc()

```
math.pow( !PROD!, 2 )
```

Number Tricks

<table>
<thead>
<tr>
<th>EXPRESSION</th>
<th>RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>!Elevation_ft! * 0.3048</td>
<td>Feet → Meters</td>
</tr>
<tr>
<td>!Elevation_m! * 3.2808399</td>
<td>Meters → Feet</td>
</tr>
<tr>
<td>!Area_km2! * 247.105381</td>
<td>km² → Acres</td>
</tr>
<tr>
<td>!Area_acres! * 0.00404685642</td>
<td>Acres → km²</td>
</tr>
<tr>
<td>round(!PROD! * !Royalty!, 2)</td>
<td>$</td>
</tr>
<tr>
<td>!Part1! / !Total!</td>
<td>%</td>
</tr>
</tbody>
</table>
String Functions

- **Built-in Functions**
  - `.upper()`
  - `.lower()`
  - `.title()`

- The field name goes **before** the `.function()`

```
!Company!.upper()
```

---

String Slicing

- Slice returns a **substring**:  
  - 2 inputs (start, end)  
  - All inputs are 0-based  
  - Negative inputs count backwards from end  
  - Non-inclusive

```
!API!![0:7]
```
Find and Replace
- Replaces the first string with the second string

REMEMBER:
- The field name goes before the .function()

```python
!CO_NAME!.replace("Inter", "Inner")
```

---

Type Conversion
- Convert values to correct data type
  - str()
  - int()
  - float()

```python
float(!GasSales!)
```

---

© 2012 TeachMeGIS. All rights reserved.
Revised 05/12
SCENARIO:
– Have a string formatted like: nnnn ssss ttt vvvv xxx yyyy

TASK:
– Format in new field as: ttt vvvv xxx yyyy

Example: String Functions

**WELL_NAME:**
‘1200 RAAA SUA HOLY BEA CALI’

**NEW WELL_NAME FIELD:**
‘SUA HOLY BEA CALI’
Example: String Functions

**SOLUTION:** 
" "join(!WELL_NAME!.split()[2:]),

<table>
<thead>
<tr>
<th>WELL_NAME</th>
<th>Column with original data</th>
<th>‘1200 RAAA SUA HOLY BEA CALI’</th>
</tr>
</thead>
<tbody>
<tr>
<td>!WELL_NAME!.split()[2:]</td>
<td>Removes the first 2 values from the list</td>
<td>‘SUA’, ‘HOLY’, ‘BEA’, ‘CALI’</td>
</tr>
<tr>
<td>&quot; &quot;.join(&lt;list&gt;)</td>
<td>Creates a string from the values in &lt;list&gt; separated by a space</td>
<td>‘SUA HOLY BEA CALI’</td>
</tr>
</tbody>
</table>

Advanced Calculations

*Show Codeblock*
Show Codeblock Calculations

- Write function in Pre-Logic Script Code section
- Call function (with inputs) in Expression section
- Returned value is written to the table

Sequential Numbers

TASK:
- Populate a column with sequential numbers

```
numVal = 0
def SequentialNums(numStart):
global numVal
if (numVal < numStart):
    numVal = numStart
else:
    numVal = numVal + 1
return numVal
```

SequentialNums(1000)
**Pick Non-Null Value**

**TASK:**
- Populate a column with either API or UWI (whichever exists)

```python
def PickNonNullString(val1, val2):
    valOut = val2
    if val1:
        valOut = val1
    return valOut
```

---

**Thanks for Attending**

Any questions?

Tracy.Rijken@TeachMeGIS.com