Empty strings (null values)

- Empty strings **should not be used** within calculations. A blank custom function named "Null" should be created and used for empty values (*see note below*)

Within solution code this increases code readability and distinguishes literal values and escaped text from empty values.

```cpp
Case {
    If ( PatternCount ( Table::fieldName ; Tab & Tab )  1;
        "Awesome use of indentation!";
        Null
    );
    If ( PatternCount ( Table::fieldName ; Tab )  1 ;
        "Good code should be easy to read";
        Null
    );
    Null
} // Note the above function uses another custom function
// which simply returns a tab character in order to improve
// readability
```

```cpp
Case {
    If ( PatternCount ( Table::fieldName ; "        " & "        " )  1;
        "Awesome use of indentation!";
        ""
    );
    If ( PatternCount ( Table::fieldName ; "        " )  1 ;
        "Good code should be easy to read";
        ""
    );
    ""
}
```

- Here's a series of test cases for clarification - the subtle differences in FM between Null, reserved word False and "" (literal empty string):

<table>
<thead>
<tr>
<th>Expression</th>
<th>Result</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null</td>
<td></td>
<td>Result is empty.</td>
</tr>
<tr>
<td>GetAsBoolean ( Null )</td>
<td>0</td>
<td>GetAsBoolean returns 0/1 for expression results.</td>
</tr>
<tr>
<td>not Null</td>
<td>1</td>
<td>The boolean inverse of Null, or 1.</td>
</tr>
<tr>
<td>Null = 0</td>
<td>0</td>
<td>False: This is boolean false because FM does not evaluate empty to reserved word False (0).</td>
</tr>
<tr>
<td>GetAsBoolean ( Null ) = 0</td>
<td>1</td>
<td>True: The explicit FM boolean interpretation of Null is 0.</td>
</tr>
<tr>
<td>GetAsBoolean ( Null ) = False</td>
<td>1</td>
<td>True: A different adaptation - FM reserved words True and False are evaluated to the boolean 0/1 values, so 0 = 0.</td>
</tr>
<tr>
<td>Null = False</td>
<td>0</td>
<td>False: This is the most interesting case of all. Null is technically empty and False evaluates to boolean 0, so they are not equal.</td>
</tr>
<tr>
<td>Null = &quot;&quot;</td>
<td>1</td>
<td>True: The literal empty string &quot;&quot; evaluates the same in FM as the empty return of Null.</td>
</tr>
</tbody>
</table>

**Exception Note**

Code within a custom function can and should use double quotes for empty values "". This decreases custom function dependencies on the "null" function - making them more portable for copy-pasting.