# ADVANCING WITH SPREADSHEETS

Taking your Data-Analysis Further with String Functions and Pivot Tables

Tom Hohenstein

BU Libraries

# STRING FUNCTIONS

# LEFT

`LEFT(text, [number of characters])`

Starts with the first character on the left and returns the number characters we specify

Example `LEFT(A1, 2)`

| | A | B |
|---|---|---|
| 1 | Tom | =LEFT(A1,2) returns "To" |
| 2 | Bob | =LEFT(A2,1) returns "B" |

# RIGHT

```
RIGHT(text, [number of characters])
```

Starts with the first character on the right and returns the number characters we specify

Example `RIGHT(A1, 2)`

| | A | B |
|---|---|---|
| 1 | Tom | =RIGHT(A1,2) returns "om" |
| 2 | Bob | =RIGHT(A2,1) returns "b" |

# MID

```
MID(text, start_num, num_chars)
```

Starts at the location you specify and returns the number of characters you specify

Example `MID(A1, 2, 1)`

| | A | B |
|---|---|---|
| 1 | Tom | =MID(A1,2,1) returns "o" |
| 2 | Bob | =RIGHT(A2,2,2) returns "ob" |

# SEARCH

`SEARCH(find text, within_text, [start_num])`

Searches for one text string within another text string and returns the starting position of the searched string.

Example `SEARCH("o", A1)`

| | A | B |
|---|---|---|
| 1 | Tom | =SEARCH("o",A1) returns 2 |
| 2 | Bob | =SEARCH("ob",A2) returns 2 |

# SEARCH + MID

You can combine SEARCH and MID to find the start positions for characters in a text string. For example:

```
=MID(A3, SEARCH(",", A3),100)
```

| | A | B |
|---|---|---|
| 1 | St. Paul, MN | =MID(A1, SEARCH(",", A1),100) returns ", MN" |

# SEARCH + MID (CONTINUED)

Let's clean that up with some math

```
=MID(A3, SEARCH(",", A3)+2,100)
```

| | A | B |
|---|---|---|
| 1 | St. Paul, MN | =MID(A1, SEARCH(",", A1)+2, 100) returns "MN" |

# LEN

`LEN(text)`

Returns the number of characters in a text string.

Example `=LEN(A1)`

| | A | B |
|---|---|---|
| 1 | Tom | =LEN(A1) returns 3 |
| 2 | Bob | =LEN(A2) returns 3 |

# LEN + LEFT

LEN + LEFT (or RIGHT) can be used to grab parts of a text string. For example city and state information within the same cell.

Example `=LEFT(A1,LEN(A1)-3)`

| | A | B |
|---|---|---|
| 1 | St. Paul MN | =LEFT(A1,LEN(A1)-3) returns St. Paul |

# OTHER TEXT FUNCTIONS

# SUBSTITUTE

```
SUBSTITUTE(text, old text, new_text, [instance number])
```

Replaces text, specific text, in a string.

```
SUBSTITUTE(A1, "Tom", "Thomas")
```

| | A | B |
|---|---|---|
| 1 | Tom Jones | =SUBSTITUTE(A1,"Tom","Thomas") returns Thomas Jones |

# EXACT

```
EXACT(text_1, text_2)
```

Checks to see if two strings are identical. Useful for checking two lists.

```
EXACT(A1, B1)
```

| | A | B | C |
|---|---|---|---|
| 1 | MN | MN | =EXACT(A1,B2) returns TRUE |
| 2 | MN | MA | =EXACT(A2,B2) returns FALSE |

# REPT

```
REPT(text, number to repeat)
```

Repeats a text a specified number of times.

```
REPT("!", 1000)
```

| | A | B |
|---|---|---|
| 1 | 5 | =REPT("!",A1) returns !!!!! |
| 2 | 3 | =REPT("!",B1) returns !!! |

# PROPER

```
PROPER(text)
```

Alters the capitalization of text so that the first letter of any string is capitalized and any character that follows a non-letter character.

```
PROPER(A1)
```

| | A | B |
|---|---|---|
| 1 | jack h | =PROPER(A1) returns Jack H |
| 2 | d7d | =PROPER(B1) returns D7D |

# UPPER

`UPPER(text)`

Converts all text to uppercase.

`UPPER(A1)`

| | A | B |
|---|---|---|
| 1 | jack h | =UPPER(A1) returns JACK H |
| 2 | d7d | =UPPER(B1) returns D7D |

# LOWER

```
LOWER(text)
```

Converts all text to lowercase.

```
LOWER(A1)
```

| | A | B |
|---|---|---|
| 1 | JaCk H | =LOWER(A1) returns jack h |
| 2 | D7D | =LOWER(B1) returns d7d |

# IF STATEMENTS

including SUMIF and COUNTIF

# BASIC STRUCTURE

```
IF(logical test, if true, if false)
```

IF is used to retun one value if the condition is true and another if the condition is false.

```
IF(A1>B1, "Great", "Bad")
```

| | A | B | C |
|---|---|---|---|
| 1 | 3 | 5 | =IF(A1>B1, "Great", "Bad") returns Bad |
| 2 | 6 | 1 | =IF(A2>B2, "Great", "Bad") returns Great |

# NESTING IF STATEMENTS

Excel allows you to nest IF statements inside an IF statement. This can become a very powerful tool (just be sure not to confuse yourself). An example

```
IF(logical test, if true, IF(logical test, if true, if false))
```

# NESTED EXAMPLE

```
IF(A1>59, IF(A1>89, "A", IF(A1>79, "B",
      IF(A1>65, "C", "D"))),"Failed"
```

| | A | B |
|---|---|---|
| 1 | 10 | ? |
| 2 | 60 | ? |
| 3 | 80 | ? |
| 3 | 90 | ? |

# SUMIF

`SUMIF(range, criteria, [sum_range])`

SUMIF allows you to sum totals given a specific criteria and is great for subtotals

Example: `SUMIF(A1:A3,A2,B1:B3)`

| | A | B | C |
|---|---|---|---|
| 1 | Tom | 5 | =SUMIF(A1:A3,A1,B1:B3) returns 9 |
| 2 | Al | 1 | =SUMIF(A1:A3,A2,B1:B3) returns 1 |
| 3 | Tom | 4 | =SUMIF(A1:A3,"Tom",B1:B3) returns 9 |

# COUNTIF

`COUNTIF(range, criteria)`

COUNTIF allows you to count instances of a specific criteria

Example: `COUNTIF(A1:A3,A2`

| | A | B | C |
|---|---|---|---|
| 1 | Tom | 5 | =COUNTIF(A1:A3,A1) returns 2 |
| 2 | Al | 1 | =COUNTIF(A1:A3,A2) returns 1 |
| 3 | Tom | 4 | =COUNTIF(A1:A3,"Tom") returns 2 |

# LOOKUPS

# VLOOKUP

```
VLOOKUP(lookup value, table array, col index num, [range lookup])
```

Let's break this vertical lookup down:

1. lookup value - the value you would like to look up (has to be the first column set in the table array).
2. table array - range of cells you want to search.
3. col index number - the column number with your return value
4. range lookup - TRUE or FALSE - determines if you want an exact or not

# VLOOKUP EXAMPLE

```
=VLOOKUP(A1,A1:C3,2,FALSE)
```

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Tom | 351 | 879 | ? |
| 2 | Al | 201 | 938 | ? |
| 3 | Jil | 765 | 832 | ? |

# HLOOKUP

```
HLOOKUP(lookup value, table array, col index num, [range lookup])
```

Let's break this horizontal lookup down:

1. lookup value - the value you would like to look up (has to be the first column set in the table array).
2. table array - range of cells you want to search.
3. col index number - the column number with your return value
4. range lookup - TRUE or FALSE - determines if you want an exact or not

# HLOOKUP EXAMPLE

```
=HLOOKUP(A1,A1:C3,2,FALSE)
```

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Tom | Al | Jil | ? |
| 2 | 33 | 201 | 938 | ? |
| 3 | 35 | 765 | 832 | ? |

# DATES

# YEAR

`YEAR(date)`

Returns the year of a given date.

Example: `YEAR(A1)`

| | A | B |
|---|---|---|
| 1 | 1/5/2015 | =YEAR(A1) returns 2015 |
| 2 | 3/2/82 | =YEAR(A2) returns 1982 |

# MONTH

`MONTH(date)`

Returns the month of a given date.

Example: `MONTH(A1)`

| | A | B |
|---|---|---|
| 1 | 1/5/2015 | =MONTH(A1) returns 1 |
| 2 | 3/2/82 | =MONTH(A2) returns 3 |

# DAY

`DAY(date)`

Returns the day of a given date.

Example: `DAY(A1)`

| | A | B |
|---|---|---|
| 1 | 1/5/2015 | =DAY(A1) returns 5 |
| 2 | 3/2/82 | =DAY(A2) returns 2 |

# WEEKDAY

`WEEKDAY(date)`

Returns the day of the week of a given date as a number. 1 for Sunday, 2 for Monday, 3 for Tuesday, etc.

Example: `WEEKDAY(A1)`

| | A | B |
|---|---|---|
| 1 | 1/5/2015 | =WEEKDAY(A1) returns 2 (Monday) |
| 2 | 3/2/82 | =WEEKDAY(A2) returns 3 (Tuesday) |

# DATEVALUE

```
DATEVALUE(date as text)
```

Converts date string to Excel recognized date value. Useful when dates are stored in text format.

Example: `WEEKDAY("1/3/2015")`

| | A | B |
|---|---|---|
| 1 | 1/5/2015 | =WEEKDAY(A1) returns 42009 |
| 2 | 3/2/82 | =WEEKDAY(A2) returns 30012 |

# DATEDIF

`DATEDIF(start_date, end_date, unit)`

Determines the numbers of days, months, and years between two dates.

Example: `DATEDIF("3/5/1982", "1/5/2016", "Y")`

| | A | B | C |
|---|---|---|---|
| 1 | 3/5/82 | 1/5/16 | =DATEDIF(A1, B1, "Y") returns 33 |
| 2 | 3/5/82 | 1/5/16 | =DATEDIF(A2, B2, "D") returns 12359 |

# WEEKNUM

`WEEKNUM(date, [return_type])`

Returns the week number of a given date. We can use the return type to change the day of the week the week begins with (the default is Sunday).

Example: `WEEKNUM("1/10/15")`

| | A | B |
|---|---|---|
| 1 | 1/10/15 | =WEEKNUM(A1) returns 2 |
| 2 | 8/30/55 | =WEEKNUM(A2) returns 36 |

# TIMES

# HOUR

`HOUR(date_time)`

Given a formatted date time cell, returns the hour.

Example: `HOUR(1/10/15 12:45 AM)`

| | A | B |
|---|---|---|
| 1 | 1/10/15 1:45 AM | =HOUR(A1) returns 1 |
| 2 | 1/10/15 6:15 AM | =HOUR(A2) returns 6 |

# MINUTE

`MINUTE(date_time)`

Given a formatted date time cell, returns the minute.

Example: `MINUTE(1/10/15 12:45 AM)`

| | A | B |
|---|---|---|
| 1 | 1/10/15 1:45 AM | =MINUTE(A1) returns 45 |
| 2 | 1/10/15 6:15 AM | =MINUTE(A2) returns 15 |

# SECOND

`SECOND(date_time)`

Given a formatted date time cell, returns the second.

Example: `SECOND(1/10/15 12:45:18 AM)`

| | A | B |
|---|---|---|
| 1 | 1/10/15 1:45:52 AM | =SECOND(A1) returns 52 |
| 2 | 1/10/15 6:15:18 AM | =SECOND(A2) returns 18 |

# TIME

`TIME(hour, minute, second)`

Returns the decimal number for the given time, ranging from 0 to .99988426 (assuming cell was formatted as general)g.

Example: `TIME(HOUR(A1), MINUTE(A1), SECOND(A1))`

| | A | B |
|---|---|---|
| 1 | 1/10/15 1:45:52 AM | =TIME(HOUR(A1), MINUTE(A1), SECOND(A1)) returns .07351852 |

# ERRORS

# #DIV/0!

You're formula is probably trying to divide by zero

# #N/A

The formula you're using was not able to produce a valid response, data is missing, or required arguments are missing.

# #NAME?

Your formula is referring to a name that Excel doesn't understand. Good chance you have a typo in your workbook.

# #NUM!

Your formula contains numeric values that Excel doesn't understand.

# #REF!

Your formula contains an invalid reference to a cell (or other data).

# #VALUE!

Your formula is having issues with the data types you're using. Often caused by having text, or other restricted characters, in math formulas.

Thank you!

tah@bu.edu