



**Spreadsheet functions**

# **How and when to use spreadsheet functions**

# Overview

**01. Data overview**

**02. Mixed cell reference**

**03. Linking sheets**

**04. The LOOKUP function**

**05. The SORT function**

**06. The FILTER and UNIQUE functions**

# Data overview

To investigate how we can use spreadsheet functions in data analytics, we will use two datasets; one on food crops and the other on their subsidies.

## 1. Average retail market prices of selected food crops 2014–2018

A Kenyan dataset containing prices for selected food crops during the months of March and September for the years 2014–2018.

## 2. Food crops subsidies 2014–2018

A dataset indicating the subsidies the Kenyan government provided for all food crops during the months of March and September for the years 2014–2018.



### Dataset 1

	A	B	C	D	E	F	G	H	I	J	K
1		2014		2015		2016		2017		2018	
2	Crop	Mar	Sept	Mar	Sept	Mar	Sept	Mar	Sept	Mar	Sept
3	Maize	38.24	35.47	33.19	33.77	33.92	35.10	48.02	43.86	41.32	30.87
4	Beans	77.16	74.67	77.56	77.08	76.74	74.36	93.96	87.46	88.10	70.86
5	Finger Millet	78.90	79.29	83.71	88.86	84.03	84.62	108.59	105.20	107.69	89.73
6	Sorghum	54.07	54.01	55.51	53.60	54.36	52.58	72.65	64.85	73.41	54.68
7	Potatoes	31.20	30.33	34.46	34.11	39.56	38.91	55.96	30.67	41.54	55.51
8	Cabbages	24.67	24.75	38.86	22.17	25.71	31.73	37.54	29.79	32.87	26.28
9	Tomatoes	58.70	68.11	68.09	55.03	70.23	52.60	73.84	79.82	65.29	63.76
10	Bananas	42.50	42.46	37.26	37.46	37.36	41.82	49.18	50.68	45.57	50.81
11											



### Dataset 2

	A	B	C	D	E	F	G	H	I	J
1		2014		2015		2016		2017		2018
2	March	September	March	September	March	September	March	September	March	September
3	10.00%	8.00%	11.00%	11.00%	15.00%	22.00%	20.00%	0.00%	0.00%	25.00%
4										

# Mixed cell reference

A **mixed cell reference** is a combination of relative and absolute references. Part of a mixed reference is fixed, either the column or the row, and the other is relative.

- **\$B3 locked column** – The column does not change. For example, replicating a cell reference with a locked column from cell **D3** to cell **E10** will indicate a change in row number but not column letter.
- **B\$3 locked row** – The row does not change. For example, replicating a cell reference with a locked row from cell **F3** to cell **G10** will indicate a change in column letter but not row number.

	A	B	C	D	E	F	G
1	Original average retail prices			Mixed reference replication			
2	Crop	March	September	Locked column		Locked row	
3	Maize	38.24	35.47	=\$B3	=\$B3	=B\$3	=C\$3
4	Beans	77.16	74.67	=\$B4	=\$B4	=B\$3	=C\$3
5	Finger Millet	78.90	79.29	=\$B5	=\$B5	=B\$3	=C\$3
6	Sorghum	54.07	54.01	=\$B6	=\$B6	=B\$3	=C\$3
7	Potatoes	31.20	30.33	=\$B7	=\$B7	=B\$3	=C\$3
8	Cabbages	24.67	24.75	=\$B8	=\$B8	=B\$3	=C\$3
9	Tomatoes	58.70	68.11	=\$B9	=\$B9	=B\$3	=C\$3
10	Bananas	42.50	42.46	=\$B10	=\$B10	=B\$3	=C\$3
	+ ≡	2014 ▼	2015 ▼	2016 ▼	2017 ▼	2018 ▼	Subsidies ▼

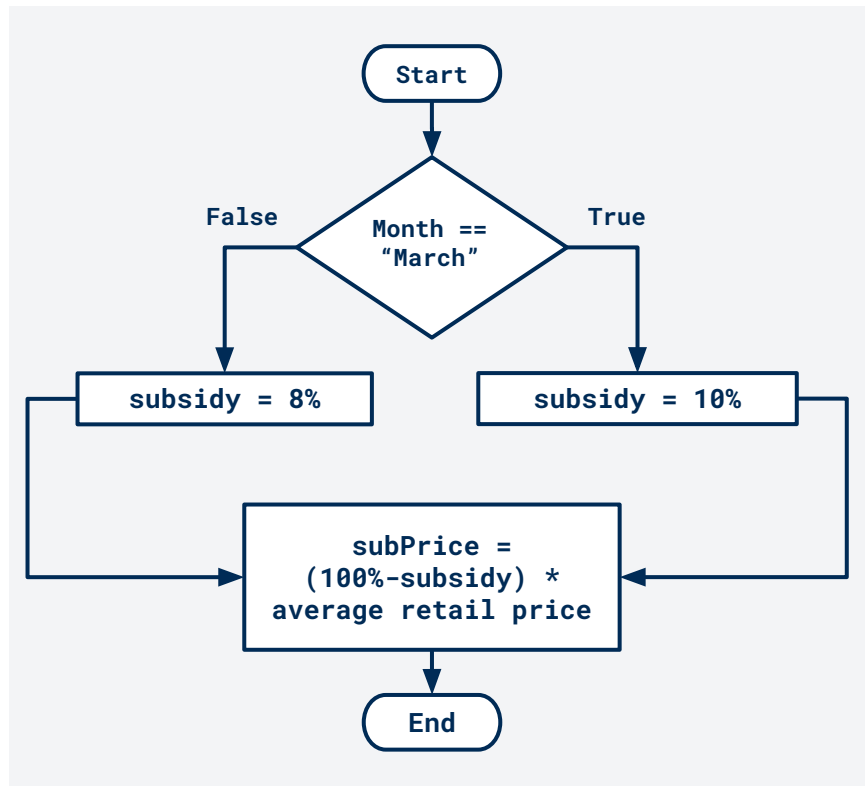
**Mixed references** are useful when the same calculations need to be used across rows and columns.

# Mixed cell reference

## Example use:

- To calculate a **food crop's subsidized price**, we will use the following formula:  
$$\text{subPrice} = (100\% - \text{subsidy}) * \text{average retail price}$$
- We only have two months' data to concentrate on for the year 2014, which are **March** with a subsidy of **10%** and **September** with a subsidy of **8%**.

```
If month == "March" then
- subsidy = 10%
Else
- subsidy = 8%
subPrice = (100%-subsidy) * average retail price
```



- We can convert this into a one-size-fits-all formula that populates all values of the **Subsidized average retail prices** table for the year 2014.

# Mixed cell reference

## Example use:

- We will use a **mixed reference** for the **subsidy percentage** value such that the **row number remains fixed** but the column number can change based on the month's subsidy.
- The **food crop's original price** will be best captured using the **relative reference** since it changes based on the food crop as well as the month.

Original average retail prices				Subsidised average retail prices		
	Crop	March	September	Crop	March	September
3	Maize	38.24	35.47	Maize	34.42	
4	Beans	77.16	74.67	Beans		
5	Finger Millet	78.90	79.29	Finger Millet		
6	Sorghum	54.07	54.01	Sorghum		
7	Potatoes	31.20	30.33	Potatoes		
8	Cabbages	24.67	24.75	Cabbages		
9	Tomatoes	58.70	68.11	Tomatoes		
10	Bananas	42.50	42.46	Bananas		
11						
12	Subsidies in percentage					
13	March	September				
14	10.0%	8.0%				

**01.** Enter `= (100% - A$14) * B3` on cell F3.

**02.** Replicate the formula to the other rows and column by dragging the fill handle down and then across to September.

We could also have used a **conditional statement**. What spreadsheet function would we use?

**Alternative solution**

# Linking sheets

**Linking sheets** is a technique that **uses cell referencing** to link a cell or range of cells **from one sheet to another** using relative, absolute, or mixed referencing.

01. Enter an **equal sign**, type in the **name of the sheet** you'd like to reference in **single quotes**, and type in an exclamation mark.
02. Type in the **relevant cell reference** then press **ENTER**.

01.

	A	B	C	D	E	F	G
1		2014		2015		2016	
2	Crop	Mar	Sept	Mar	Sept	Mar	Sept
3	Maize	= '2014' !					
4	Beans						

Reference sheet

2014-2018 subsidized 2014 2015

02.

**Master sheet:**  
2014-2018 subsidised

	A	B	C
1		2014	
2	Crop	Mar	Sept
3	Maize	38.24	
4	Beans		

2014-2018 sub

**Linked sheet:**  
2014

	A	B
1	Original average retail	
2	Crop	March
3	Maize	38.24
4	Beans	77.16

2014 2015

# Linking sheets

## Example use:

We can create a **master view** of the subsidized retail prices for 2014 to 2018 by **consolidating** information from the respective worksheets into one.

**01.** Enter the formula `= '2014' !B3` on cell B3 of the master sheet to reference data from the 2014 sheet.

**02.** Replicate the formula to other rows and to the **Sept** column for 2014 by dragging the fill handle down then across.

**03.** Repeat for years 2015 to 2018.

Linking sheets has an **advantage** over copy-pasting in that **changes** made on the original sheet **automatically reflect** on the referencing sheet.

01.

02.

	A	B	C
1			
2	Crop	Mar	Sept
3	Maize	38.24	
4	Beans		
5	Finger Millet		
6	Sorghum		
7	Potatoes		
8	Cabbages		
9	Tomatoes		
10	Bananas		

	A	B	C
1			
2	Crop	Mar	Sept
3	Maize	38.24	35.47
4	Beans	77.16	74.67
5	Finger Millet	78.90	79.29
6	Sorghum	54.07	54.01
7	Potatoes	31.20	30.33
8	Cabbages	24.67	24.75
9	Tomatoes	58.70	68.11
10	Bananas	42.50	42.46

03.

	A	B	C	D	E	F	G	H	I	J
1										
2	Crop	Mar	Sept	Mar	Sept	Mar	Sept	Mar	Sept	Mar
3	Maize	38.24	35.47	37.26	37.46	33.92	35.10	48.02	43.86	41.32
4	Beans	77.16	74.67	77.56	77.08	76.74	74.36	93.96	87.46	88.10
5	Finger Millet	78.90	79.29	38.86	22.17	84.03	84.62	108.59	105.20	107.69
6	Sorghum	54.07	54.01	83.71	88.86	54.36	52.58	72.65	64.85	73.41
7	Potatoes	31.20	30.33	33.19	33.77	39.56	38.91	55.96	30.67	41.54
8	Cabbages	24.67	24.75	34.46	34.11	25.71	31.73	37.54	29.79	32.87
9	Tomatoes	58.70	68.11	55.51	53.60	70.23	52.60	73.84	79.82	65.29
10	Bananas	42.50	42.46	68.09	55.03	37.36	41.82	49.18	50.68	45.57
11										
12										
13										



# The LOOKUP function

A **LOOKUP** function is used to search for a value in a range of cells and returns a corresponding value from a specific column or row.

```
=LOOKUP(search_key, search_range|search_result_array, [result_range])
```

- **search\_key** – The value to search for in the row or column.
- **search\_range** – The row or column to look through for the search with a second argument **result\_range**.
- **search\_result\_array** – A combination of **search\_range** and **result\_range** where the first row or column is searched and a value is returned from the last row or column in the array.
- **result\_range** – [OPTIONAL] The range from which to return a result.

# The LOOKUP function

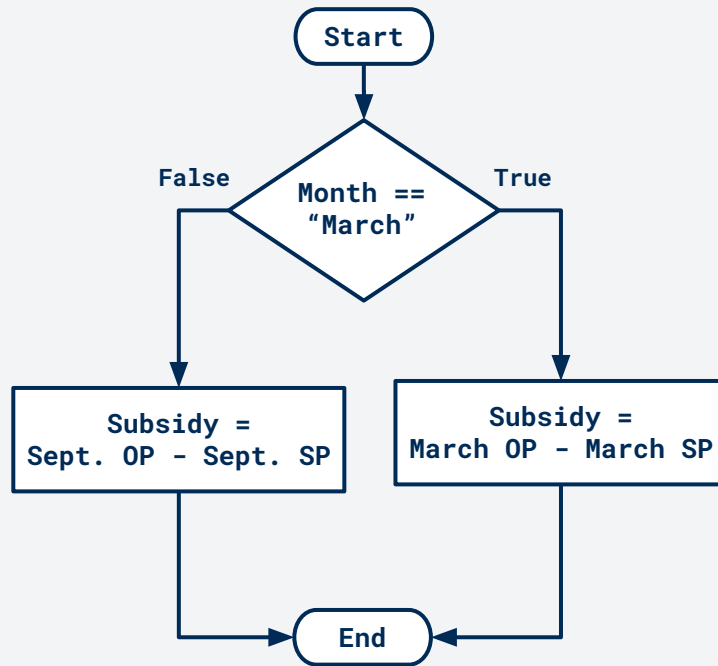
## Example use:

- Create a table that shows the **amount of money** that was **subsidized** for each food crop in **2014**.
- We only have two months' data to concentrate on for the year 2014; March and September.

Let a food crop's *original price* be represented by **OP** and its *subsidized price* be represented by **SP**:

```
If month == "March" then
- Subsidy = March OP - March SP
Else
- Subsidy = Sept. OP - Sept. SP
```

- We will use two **LOOKUP** functions to calculate the price difference. One will search for a crop's original price and the other for its subsidized price.





# The alternative LOOKUP functions

**VLOOKUP** and **HLOOKUP** are used to retrieve **specific** data from a spreadsheet. **VLOOKUP** only searches in a **column** (vertically) while **HLOOKUP** only searches in a **row** (horizontally).

```
=VLOOKUP(search_key, range, index, [is_sorted])
```

```
=HLOOKUP(search_key, range, index, [is_sorted])
```

- **search\_key** – The value to search for.
- **range** – The upper and lower values to consider for the search.
- **index** – The index of the column (VLOOKUP) or row (HLOOKUP) with the return value of the range. The first column (VLOOKUP) or row (HLOOKUP) in **range** is numbered 1. The index must be a positive integer.
- **[is\_sorted]** – [OPTIONAL – TRUE by default]
  - **FALSE** = Exact match. This is recommended.
  - **TRUE or omitted** = Approximate match.  
*Important:* Before you use **TRUE**, sort your search key in ascending order. Otherwise, you may get a wrong return value.

# The alternative LOOKUP functions

## Example use:

- A** Compare the **price of maize** in **March and September** 2015

Since March and September values for maize are in columns, we will use two **VLOOKUPs** for this problem. Our **range** will consider **three columns**:

- column index 1** contains the name of the crops which will be used to locate the row containing data on maize prices;
- column index 2** contains the return value for March prices; and
- column index 3** contains the return value for September prices.

- B** Compare the **price of maize and beans** in **March** 2015

For the second problem, we will use two **HLOOKUPs** since the values of maize and beans are in rows. Our **range** will consider **nine rows** but we will concentrate on three:

- row index 1** contains the name of the months which will be used to locate the column containing data on **March** prices;
- row index 2** contains the return value for maize prices; and
- row index 3** contains the return value for beans prices.

# The alternative LOOKUP functions

A

01.

Enter the formula  
`=VLOOKUP(F2,A3:C10,2,FALSE)` on cell F3.

02.

Enter the formula  
`=VLOOKUP(F2,A3:C10,3,FALSE)` on cell F4.

B

03.

Enter the formula  
`=HLOOKUP(F6,B2:C10,2,FALSE)` on cell F7.

04.

Enter the formula  
`=HLOOKUP(F6,B2:C10,3,FALSE)` on cell F8.

## Example use:

F3		=VLOOKUP(F2,A3:C10,2,FALSE)	01.
F4		=VLOOKUP(F2,A3:C10,3,FALSE)	02.
F7		=HLOOKUP(F6,B2:C10,2,FALSE)	03.
F8		=HLOOKUP(F6,B2:C10,3,FALSE)	04.

	A	B	C	D	E	F
1	Original average retail prices					
2	Crop	March	September		VLOOKUP	Maize
3	Maize	37.26	37.46		March 01.	37.26
4	Beans	77.56	77.08		Septem 02.	37.46
5	Finger Millet	38.86	22.17			
6	Sorghum	83.71	88.86		HLOOKUP	March
7	Potatoes	33.19	33.77		Maize 03.	37.26
8	Cabbages	34.46	34.11		Beans 04.	77.56
9	Tomatoes	55.51	53.60			
10	Bananas	68.09	55.03			
	+	2014	2015	2016	2017	2018

# The SORT function

**SORT** is used to sort the rows of a given range by the values in one or more columns.

```
=SORT(range, sort_column, is_ascending, [sort_column2, ...], [is_ascending2, ...])
```

- **range** – The data to be sorted.
- **sort\_column** – The index of the column in **range** or a range outside of **range** containing the values by which to sort.
  - A range specified as a **sort\_column** must be a single column with the same number of rows as **range**.
- **is\_ascending** – **TRUE** or **FALSE** indicating whether to sort **sort\_column** in ascending order. **FALSE** sorts in descending order.
- **sort\_column2, is\_ascending2 ...** – [OPTIONAL] Additional columns and sort order flags beyond the first, in order of precedence.

# The SORT function

## Example use:

**Reorder** the food crops in **descending order** of the March prices for the year 2015.

- We will select the names of the crops plus their prices (March and September) for our **range** because we want to make sure that all crops move with their respective prices during the sort.
- The **sort\_column** will be the range of prices for March.
- Since we want to sort in descending order, we will use **FALSE** as our **is\_ascending** value.

**01.** Enter the formula `=SORT(A3:C10,2,FALSE)` on cell E3.

**02.** Press **Enter**.

E3	=SORT(A3:C10,2,FALSE)						
	A	B	C	D	E	F	G
1	Original average retail prices				Sorted average retail prices		
2	Crop	March	September		Crop	March	September
3	Maize	37.26	37.46		Sorghum	83.71	88.86
4	Beans	77.56	77.08		Beans	77.56	77.08
5	Finger Millet	38.86	22.17		Bananas	68.09	55.03
6	Sorghum	83.71	88.86		Tomatoes	55.51	53.60
7	Potatoes	33.19	33.77		Finger Millet	38.86	22.17
8	Cabbages	34.46	34.11		Maize	37.26	37.46
9	Tomatoes	55.51	53.60		Cabbages	34.46	34.11
10	Bananas	68.09	55.03		Potatoes	33.19	33.77
+ ≡ 2014 2015 2016 2017 2018 Su							



# The alternative SORT function

**SORTN** returns the **first n** items in a dataset after sorting it.

```
=SORTN(range, [n], [display_ties_mode], [sort_column], [is_ascending],  
[sort_column2, ...], [is_ascending2, ...])
```

- **range** – The data to be sorted.
- **[n]** – The number of items to return.
- **[sort\_column]** – [OPTIONAL] The column containing the values to sort by.
- **[is\_ascending]** – [OPTIONAL] Indicates if **sort\_column** is ascending or descending.
- **[sort\_column2, ...], [is\_ascending2, ...]** – [OPTIONAL] Additional columns to sort.
- **[display\_ties\_mode]** – [OPTIONAL] A number representing how to display ties:
  - 0: Only show the first n rows.
  - 1: Show first n rows, plus any additional rows that are identical to the nth row.
  - 2: Show first n rows with no duplicates.
  - 3: Show at most the first n unique rows, but show every duplicate of these rows.

# The alternative SORT function

## Example use:

Create a list of the **four priciest** crops, in **descending** order, for September 2015.

- Our solution will look a lot like that of **SORT**.
- However, we will have to make sure we change the function to **SORTN**, specify that we only want **4** items to be returned, and change the **sort\_column** to September price range.
- We will also add **2** as our **display\_ties\_mode** just in case our dataset has duplicates.

**01.** Enter the formula **=SORTN(A3:C10,4,2,C3:C10, FALSE)** on cell E4.

**02.** Press **Enter**.

E4	=SORTN(A3:C10,4,2, C3:C10, FALSE)							
	A	B	C	D	E	F	G	
1	Original average retail prices							
2	Crop	March	September		Sorted average retail prices			
3	Maize	37.26	37.46		Crop	March	September	
4	Beans	77.56	77.08		Sorghum	83.71	88.86	
5	Finger Millet	38.86	22.17		Beans	77.56	77.08	
6	Sorghum	83.71	88.86		Bananas	68.09	55.03	
7	Potatoes	33.19	33.77		Tomatoes	55.51	53.60	
8	Cabbages	34.46	34.11					
9	Tomatoes	55.51	53.60					
10	Bananas	68.09	55.03					
+		≡	2014	2015	2016	2017	2018	Subs

# The FILTER and UNIQUE functions

**FILTER** returns only **rows or columns** from the source range that **meet** the specified filtering **conditions** while **UNIQUE** **discards duplicate** rows and only **returns unique** rows in the provided source range.

`=FILTER(range, condition1, [condition2, ...])`

`=UNIQUE(range)`

- **range** – The data to be filtered.
- **condition1** – A row or column containing **TRUE** or **FALSE** values corresponding to the first column or row of **range**.
- **condition2** – [OPTIONAL] Additional rows or columns containing boolean values.

fx		=FILTER(A3:C13, A3:A13="Maize")										
fx		=UNIQUE(A3:C13)										
Original average retail prices			Filtered average retail prices			Unique average retail prices						
Crop	March	September	Crop	March	September	Crop	March	September				
Maize	37.26	37.46	Maize	37.26	37.46	Maize	37.26	37.46				
Cabbages	34.46	34.11				Cabbages	34.46	34.11				
Beans	77.56	77.08	Maize	37.26	37.46	Beans	77.56	77.08				
Finger Millet	38.86	22.17				Finger Millet	38.86	22.17				
Sorghum	83.71	88.86				Sorghum	83.71	88.86				
Maize	37.26	37.46				Potatoes	33.19	33.77				
Potatoes	33.19	33.77				Tomatoes	55.51	53.60				
Cabbages	34.46	34.11				Bananas	68.09	55.03				
Tomatoes	55.51	53.60										
Bananas	68.09	55.03										
Tomatoes	55.51	53.60										

# The FILTER and UNIQUE functions

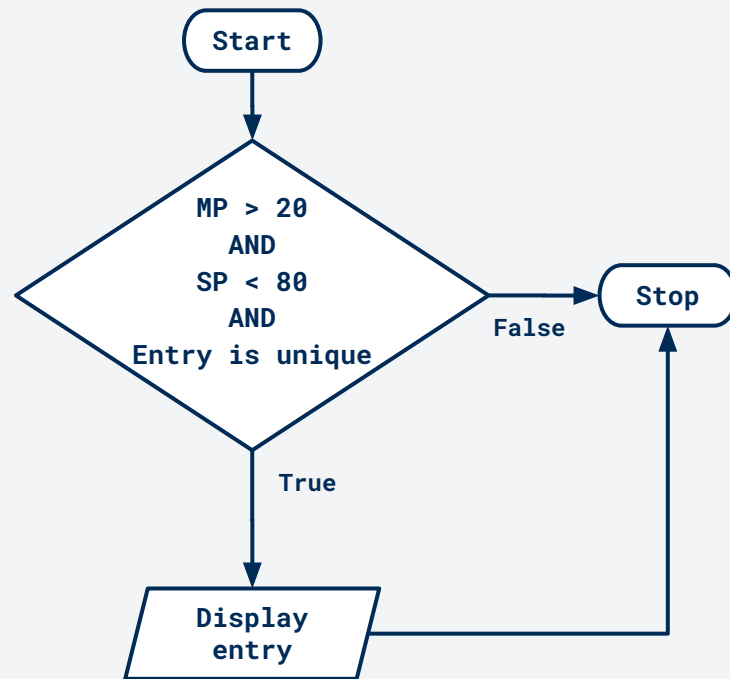
## Example use:

**Create a table** that lists food crops whose prices were **greater than** KSh 20 in March and **less than** KSh 80 in September of 2015.

Make sure that this table has **no duplicate** entries.

Let a *March price* be represented by **MP** and *September price* by **SP**:

If **MP > 20 AND SP < 80 AND entry is unique**  
- Display **entry**



# The FILTER and UNIQUE functions

E4	=UNIQUE(FILTER(A3:C13,B3:B13>20,C3:C13<80))							
	A	B	C	D	E	F	G	H
1	Original average retail prices							
2	Crop	March	September		Filtered unique average retail prices			
3	Maize	37.26	37.46		Crop	March	September	
4	Cabbages	34.46	34.11		Maize	37.26	37.46	
5	Beans	77.56	77.08		Cabbages	34.46	34.11	
6	Finger Millet	38.86	22.17		Beans	77.56	77.08	
7	Sorghum	83.71	88.86		Finger Millet	38.86	22.17	
8	Maize	37.26	37.46		Potatoes	33.19	33.77	
9	Potatoes	33.19	33.77		Tomatoes	55.51	53.60	
10	Cabbages	34.46	34.11		Bananas	68.09	55.03	
11	Tomatoes	55.51	53.60					
12	Bananas	68.09	55.03					
13	Tomatoes	55.51	53.60					
14								
+    2014    2015    2016    2017    2018    Subsid								

## Example use:

- We will use the **FILTER** function to list all food crop entries with prices that are **greater than** KSh 20 in March and **less than** KSh 80 in September.
- We will then wrap the **FILTER** function with a **UNIQUE** function to make sure that no duplicate entries are included in our list.

- Enter the formula  
`=UNIQUE(FILTER(A3:C13, B3:B13>20, C3:C13<80))` in cell E4.
- Press **Enter**.