How Do I Create Dynamic Formulas in Excel Templates?

You can use Excel's dynamic formulas - formulas that produce data on the fly - in Report Designer Office Edition (*the Designer*) templates.

The Designer carries functions across cells, and it knows when to include new cells and when to change range numbers. Here we will walk though a simple example. This tutorial begins with a blank template connected to the Northwind XML data source.

ℹ This tutorial uses an XML data source, and the examples shown here use the sample data source Northwind. You can follow along with this tutorial by using the Northwind XML file that ships with the Designer. Or, you can follow these steps with your organization's own data sources, keeping in mind your specific choices (data source location, server name, database name, data groups, etc.) will be different from what's listed here.

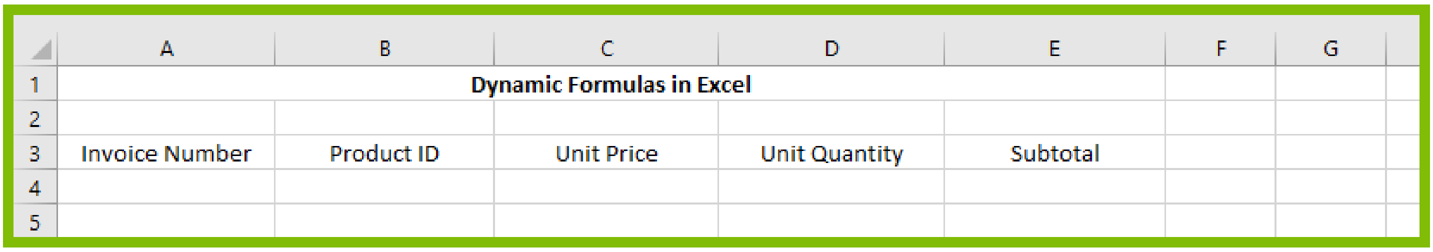
Create a Table

Before we can begin placing Tags in the template, we need to know where they will go. In this example, we'll be creating a table of products and information about those products. This table will have columns for the invoice number, the product ID, the product's unit price, the product's quantity, and a column subtotaling the cost of each product ordered.

Beginning with a new Excel workbook, choose ten cells in which you will create a 5x2 table. We choose this size because it gives us one column for each data category. In the first row we will place the data titles, and in the second row we will place the Tags. You do not need to know how many rows of data will be output when you run the report; the Designer takes care of expanding that for you automatically.

We recommend that you do not use the table command in Excel. Instead, use cells only for table construction.

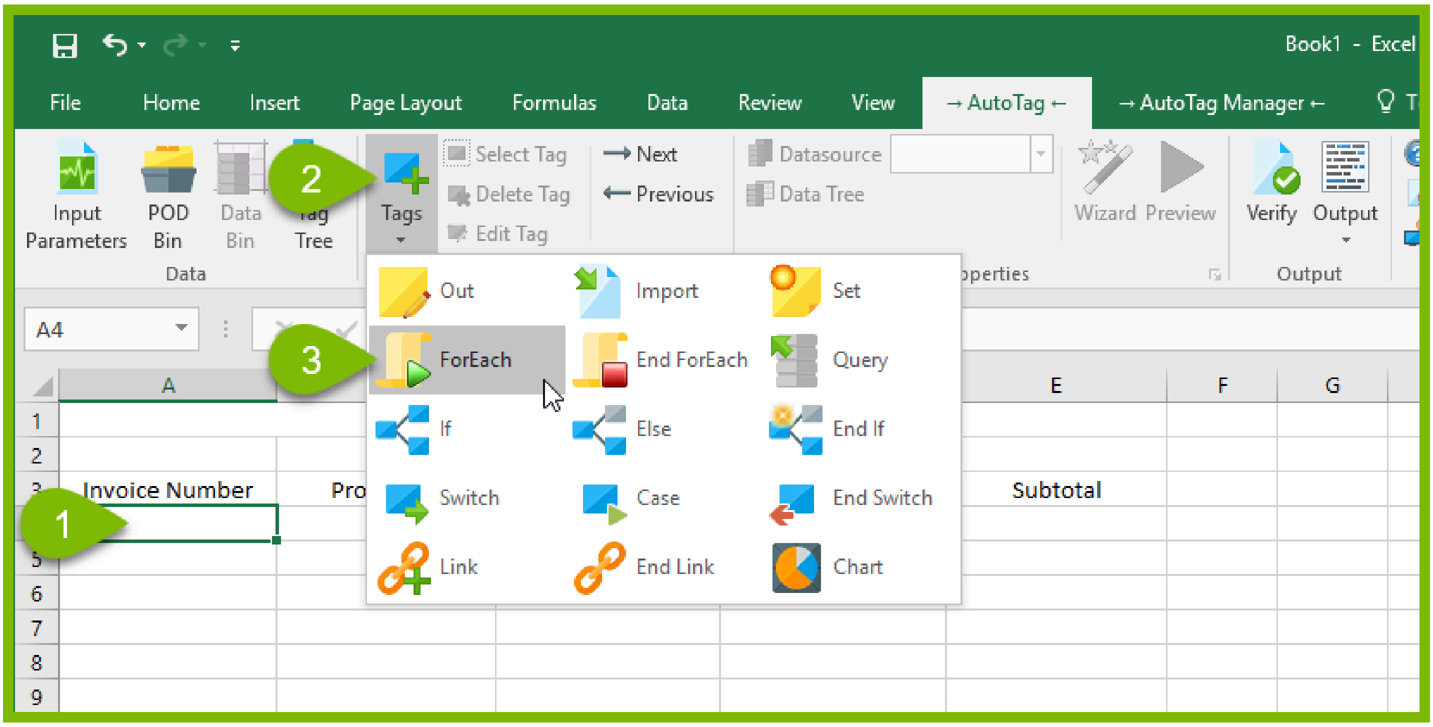
In the table's first row, enter the column headings:



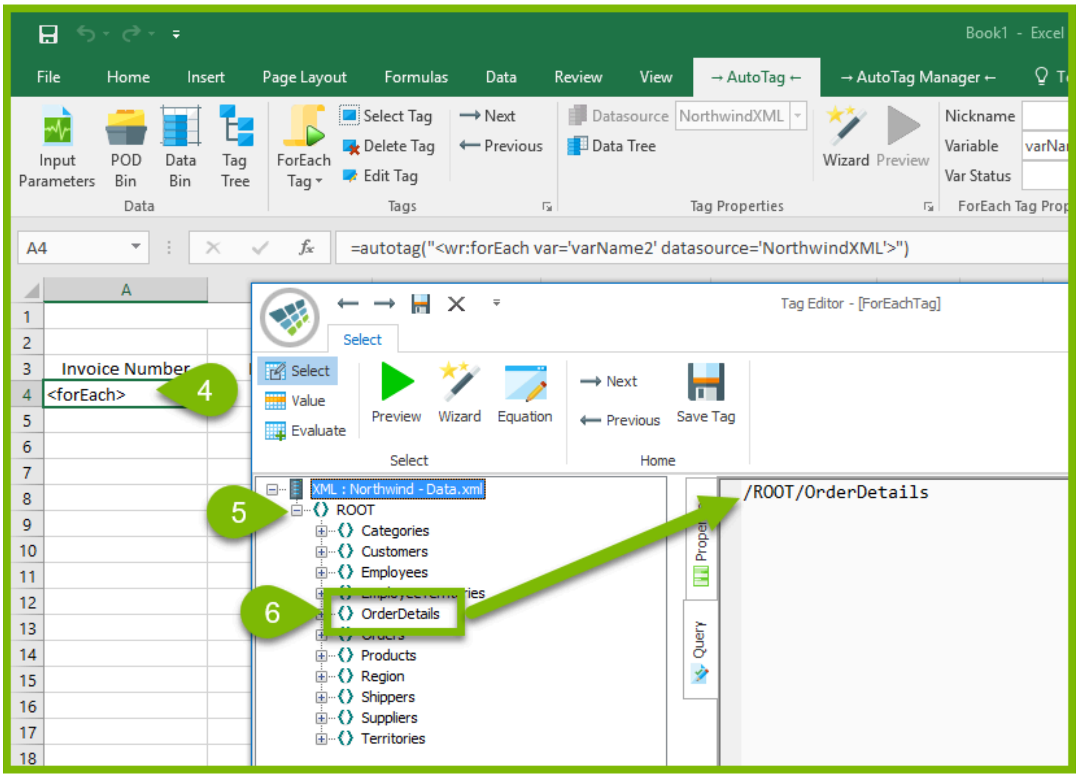
	A	B	C	D	E	F	G
1	Dynamic Formulas in Excel						
2							
3	Invoice Number	Product ID	Unit Price	Unit Quantity	Subtotal		
4							
5							

Create a ForEach Tag

1. Click on the first cell in what will be the second row of the table (the cell A4 in the image above).
2. Click on the Tags button.
3. Select the ForEach Tag.



4. Double-click on the A4 cell, which now contains a ForEach Tag, to open the Tag Editor.
5. In the Data Pane on the left side of the Tag Editor, expand the ROOT node.
6. Drag and drop the "OrderDetails" node from the Data Pane to the Query Pane on the the right.



A4 =autotag("<wr:forEach var='varName2' datasource='NorthwindXML'>")

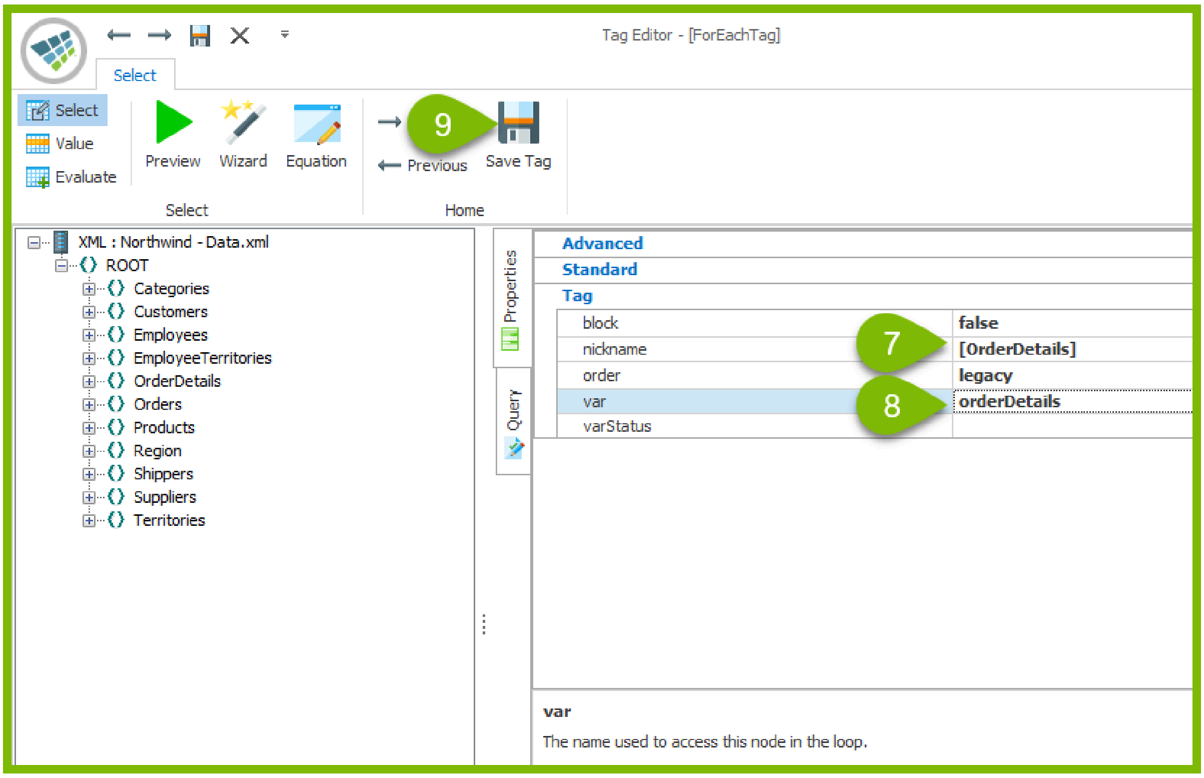
	A
3	Invoice Number
4	<forEach>

Datasource: NorthwindXML
Variable: varNa

XML : Northwind - Data.xml
- ROOT
  - Categories
  - Customers
  - Employees
  - EmployeeTerritories
  - OrderDetails
  - Orders
  - Products
  - Region
  - Shippers
  - Suppliers
  - Territories

```
/ROOT/OrderDetails
```

7. Give the ForEach Tag a descriptive nickname.
8. Give the ForEach Tag a descriptive variable name.
9. Save the Tag.



XML : Northwind - Data.xml
- ROOT
  - Categories
  - Customers
  - Employees
  - EmployeeTerritories
  - OrderDetails
  - Orders
  - Products
  - Region
  - Shippers
  - Suppliers
  - Territories

Advanced
Standard
Tag

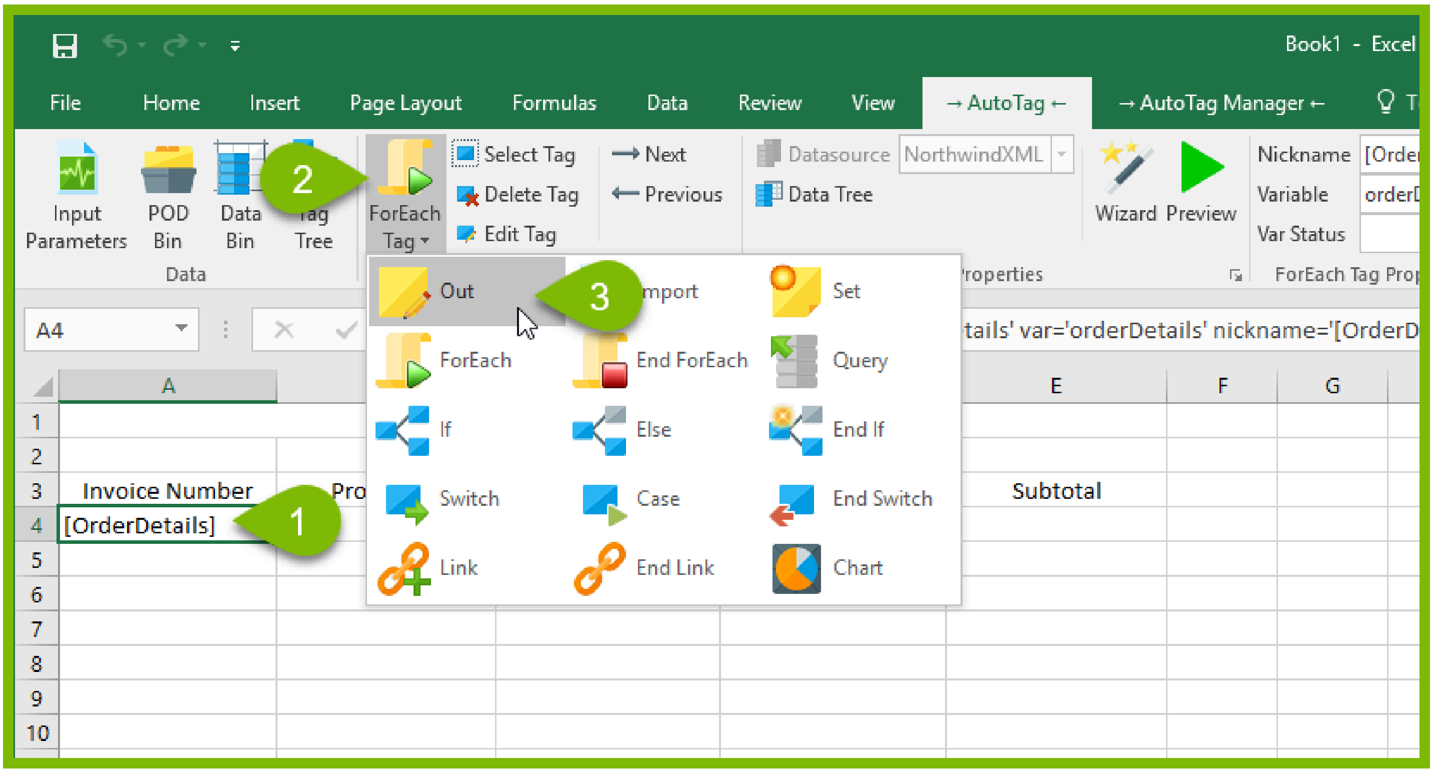
block	false
nickname	[OrderDetails]
order	legacy
var	orderDetails
varStatus	

var
The name used to access this node in the loop.

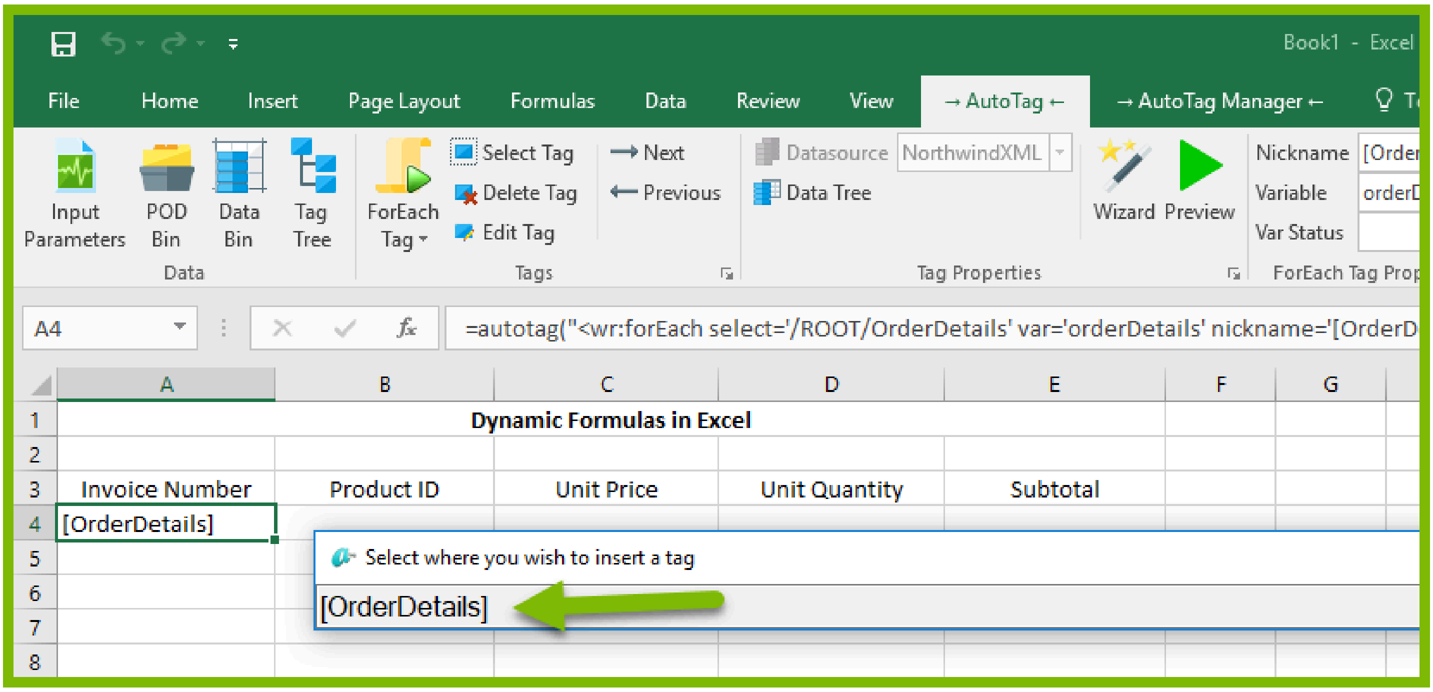
Create the First Out Tag

Now it's time to place the data subgroups to output into the individual cells.

1. Click on the cell that holds the ForEach Tag.
2. Click on the ForEach Tag button (the Tags button changes based on what Tag has been inserted).
3. Select the Out Tag.



A prompt appears, asking you where you'd like to place the Out Tag. Place your cursor just after the ForEach Tag's nickname ("[OrderDetails]") and click. The Designer places the Out Tag after the ForEach Tag in the same cell.



A4: =autotag("<wr:forEach select='/ROOT/OrderDetails' var='orderDetails' nickname='[OrderD

Nickname: [Order

Variable: orderD

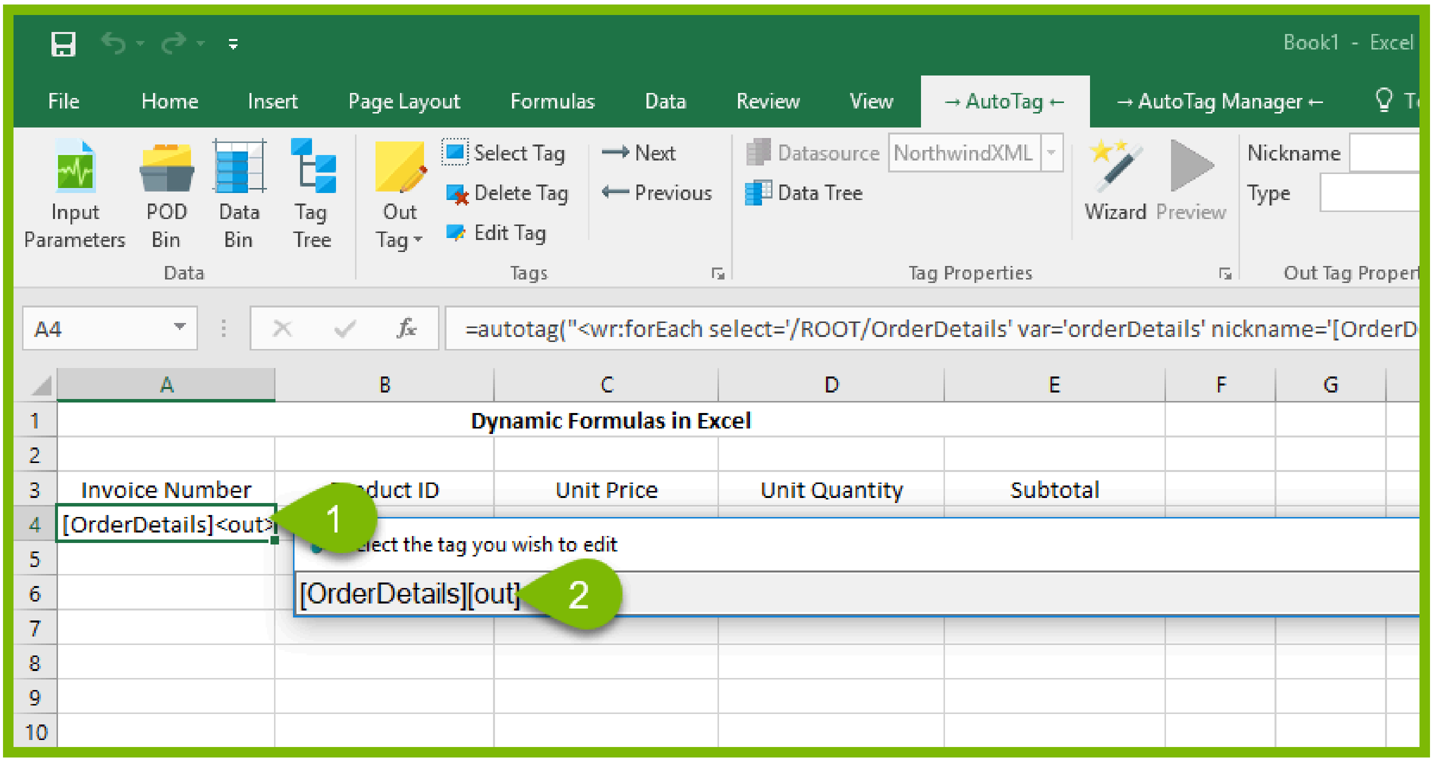
	A	B	C	D	E
1			Dynamic Formulas in Excel		
2					
3	Invoice Number	Product ID	Unit Price	Unit Quantity	Subtotal
4	[OrderDetails]				

Select where you wish to insert a tag

[OrderDetails]

Attach a Data Field to Your Out Tag

1. Double-click on the cell where you inserted your ForEach and Out Tags. A prompt will appear asking you to select which Tag to edit.
2. Click on the Out Tag to bring up the Tag Editor.



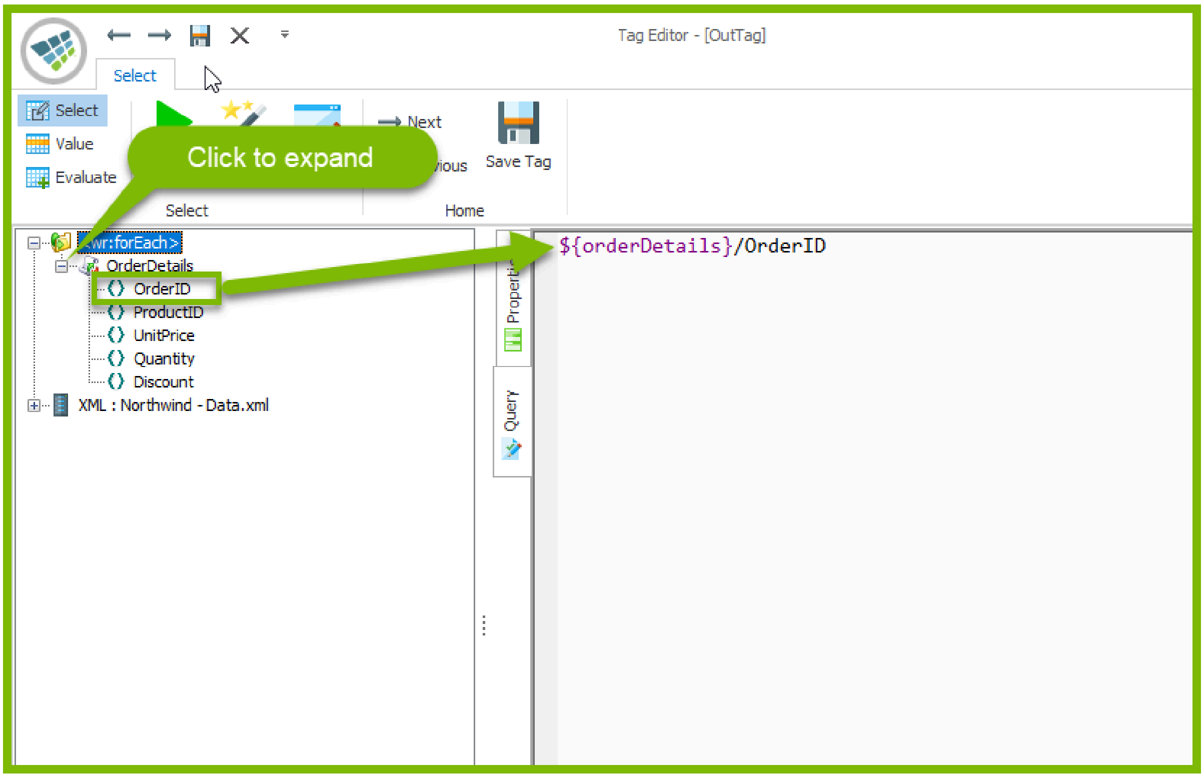
A4: =autotag("<wr:forEach select='/ROOT/OrderDetails' var='orderDetails' nickname='[OrderD

	A	B	C	D	E
1			Dynamic Formulas in Excel		
2					
3	Invoice Number	duct ID	Unit Price	Unit Quantity	Subtotal
4	[OrderDetails]<out>				

ect the tag you wish to edit

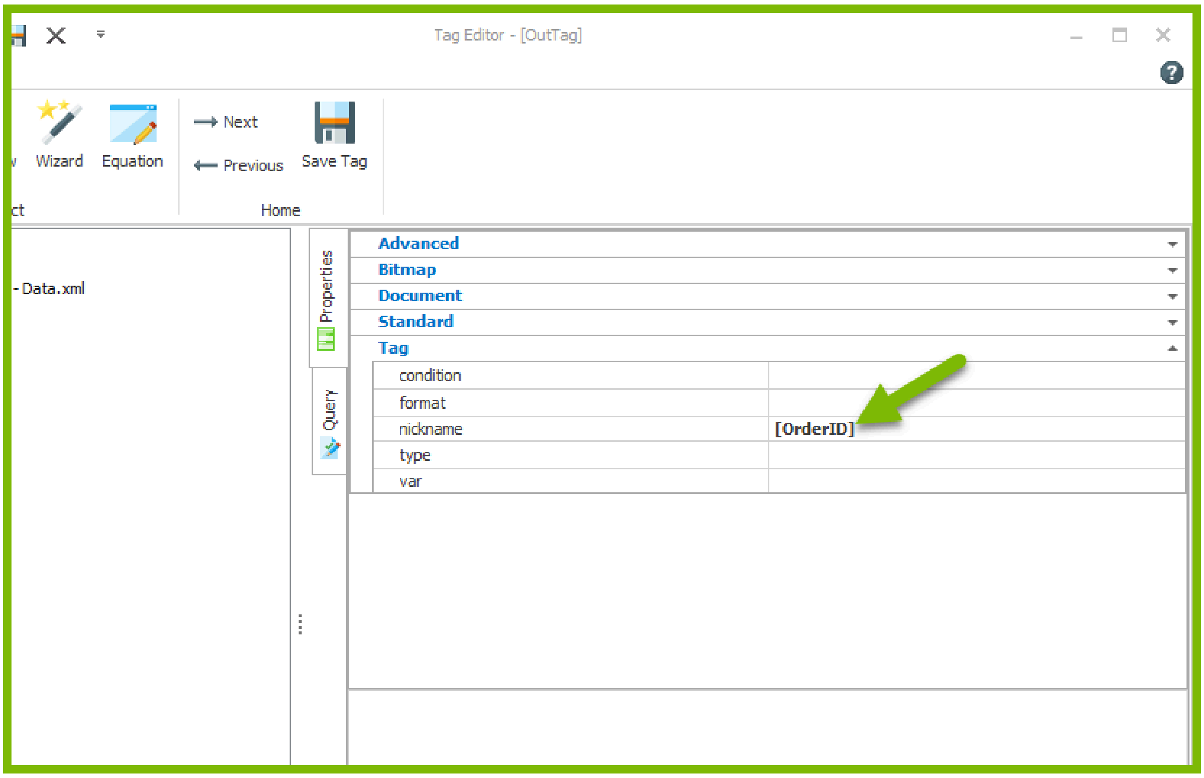
[OrderDetails][out]

In the Tag Editor, expand the ForEach Tag variable you created above, "OrderDetails", by clicking on the + sign next to it. The "OrderID" group contains the invoice number, so you will drag the OrderID data subgroup from the OrderDetails variable into the Query Pane.



```
${orderDetails}/OrderID
```

Give the Out Tag a descriptive nickname, and save the Tag.



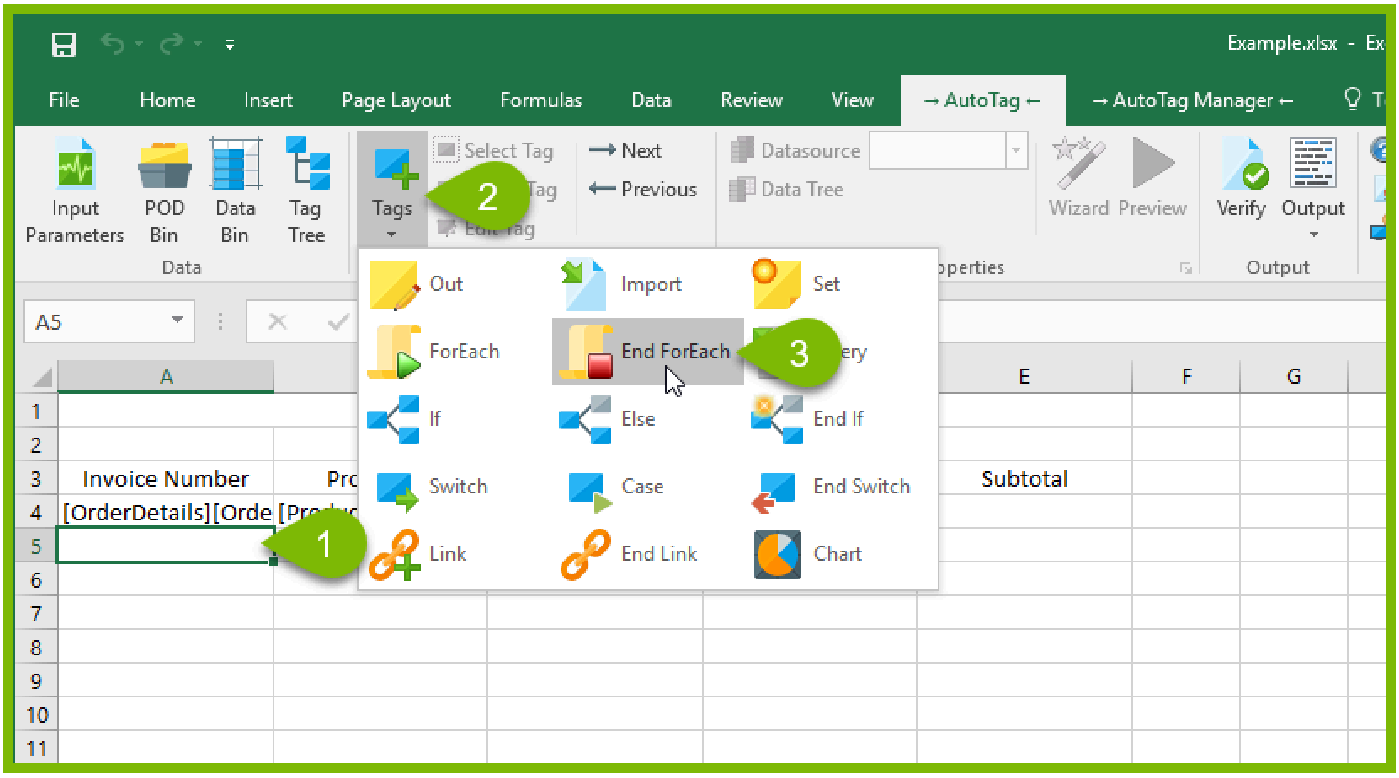
condition	
format	
nickname	[OrderID]
type	
var	

Create Additional Out Tags

Using the procedure above, create Out Tags for the three remaining cells. However, each of these Out Tags will be in its own cell, so you will not see a prompt asking you where to insert the Tag as before. The "ProductID" subgroup goes into the Query Pane of the Tag in the "Product ID" column; the UnitPrice subgroup goes into the Query Pane of the Tag in the "Unit Price" column; and the "Quantity" subgroup goes into the Query Pane of the Tag in the "Unit Quantity" column.

Close the ForEach Tag Loop

1. Click on the empty cell under your first ForEach Tag (A5).
2. Click on the Tags button.
3. Select the End ForEach button. This will close your ForEach loop in your table.

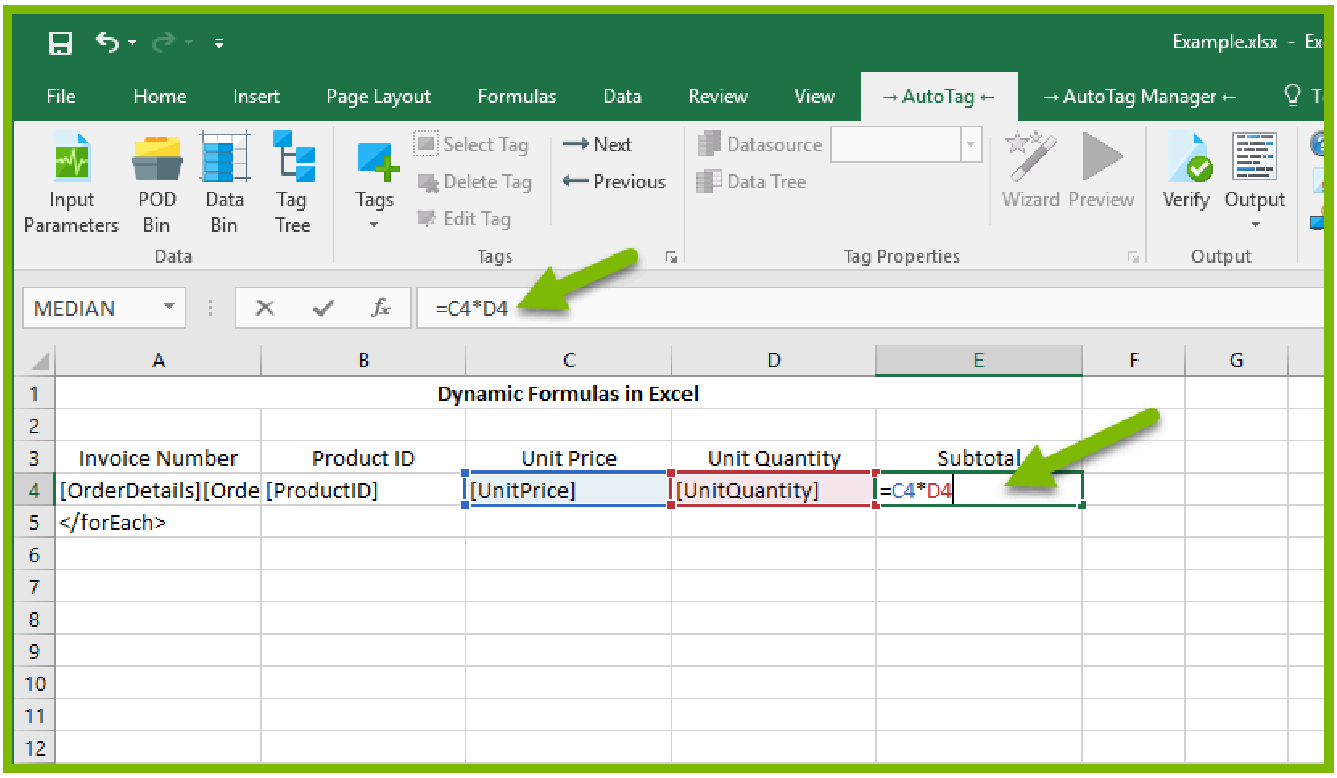


Create a Subtotal

Here's where we get down to business – create a subtotal using Excel's built-in dynamic formula for multiplication.

Now we want to know how much is being spent on each item. In other words, we want to multiply the unit price by the unit quantity for each item, and that will give us our subtotal.

Click on the cell below the cell labeled "Subtotal". Enter the formula Excel uses for multiplication: an equal sign, the location of the first cell to be multiplied, an asterisk, then the location of the second cell to be multiplied. Your template will look like this (note the formula in the formula bar):



MEDIAN

=C4*D4

	A	B	C	D	E	F	G
1			Dynamic Formulas in Excel				
2							
3	Invoice Number	Product ID	Unit Price	Unit Quantity	Subtotal		
4	[OrderDetails][Orde	[ProductID]	[UnitPrice]	[UnitQuantity]	=C4*D4		
5	</forEach>						
6							
7							
8							
9							
10							
11							
12							

The power in using this formula is the Designer will automatically create the additional Subtotal cells to match the data. Instead of just one subtotal, we will see a subtotal for each row generated by the ForEach Tag. Even if our data source contains hundreds of orders, we only have to input the formula once.

When the Designer changes references in an Excel formula, it does not parse the formula to determine what parts are references. It just looks for XX123, where the letters are A – ZZ and then numbers.

So if you have a custom macro named "A2", as in "A2(B1, C3)", the Designer will think the "A2" is a cell reference, not the name of a macro. It's unlikely you'll name a macro with one or two letters followed by a number; but if you do, the Designer will break because it will change the macro if it's moved/replicated in the final report.

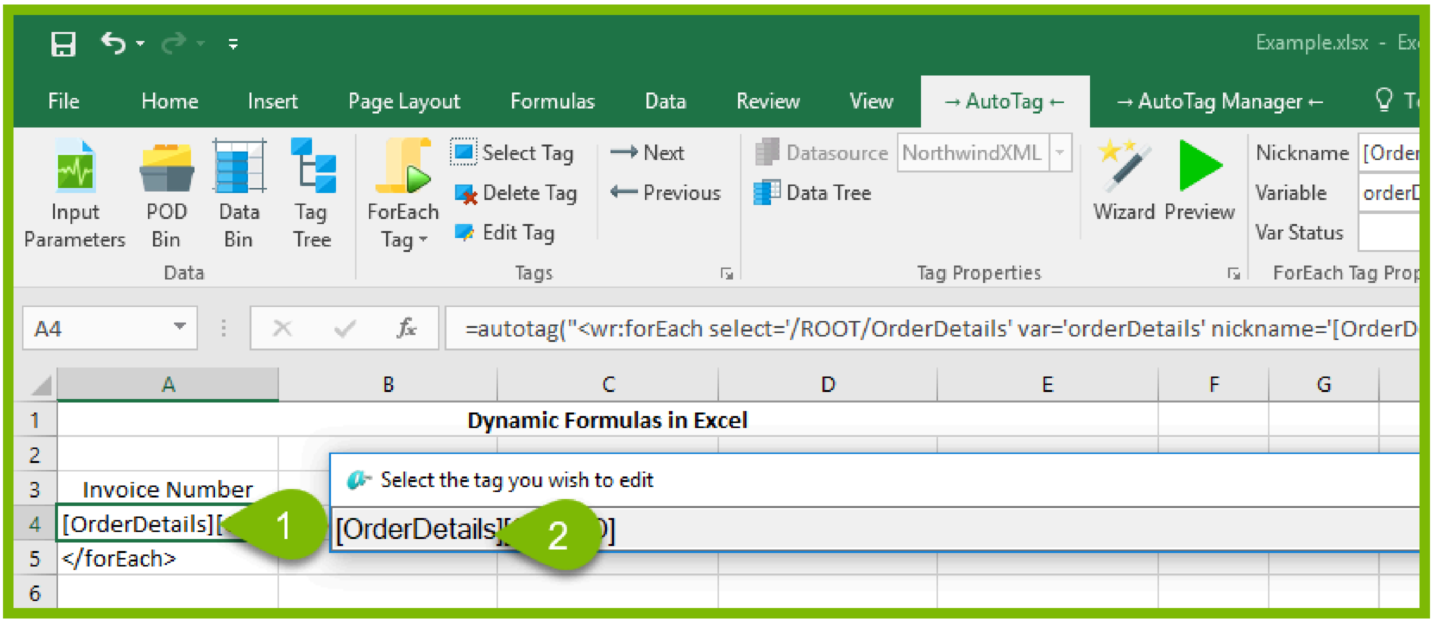
Filter the Table (Optional)

It isn't necessary to add filters to our table in order to demonstrate the formula features in Excel, but because we are working with a large set of data, we will do so now for two reasons:

- Our data source is quite large, and this will make illustrating the next formula much simpler
- The formula we are going to create will sum items, and using a filter now will allow us to demonstrate a very practical application: summing items and coming up with a total cost for a particular invoice in our system

Returning to our example Template:

1. Double-click on the cell containing the ForEach Tag (A4).
2. In the "Select the tag you wish to edit" prompt, click on the ForEach Tag.



A4

=autotag("<wr:forEach select='/ROOT/OrderDetails' var='orderDetails' nickname='[OrderD

Dynamic Formulas in Excel

Invoice Number

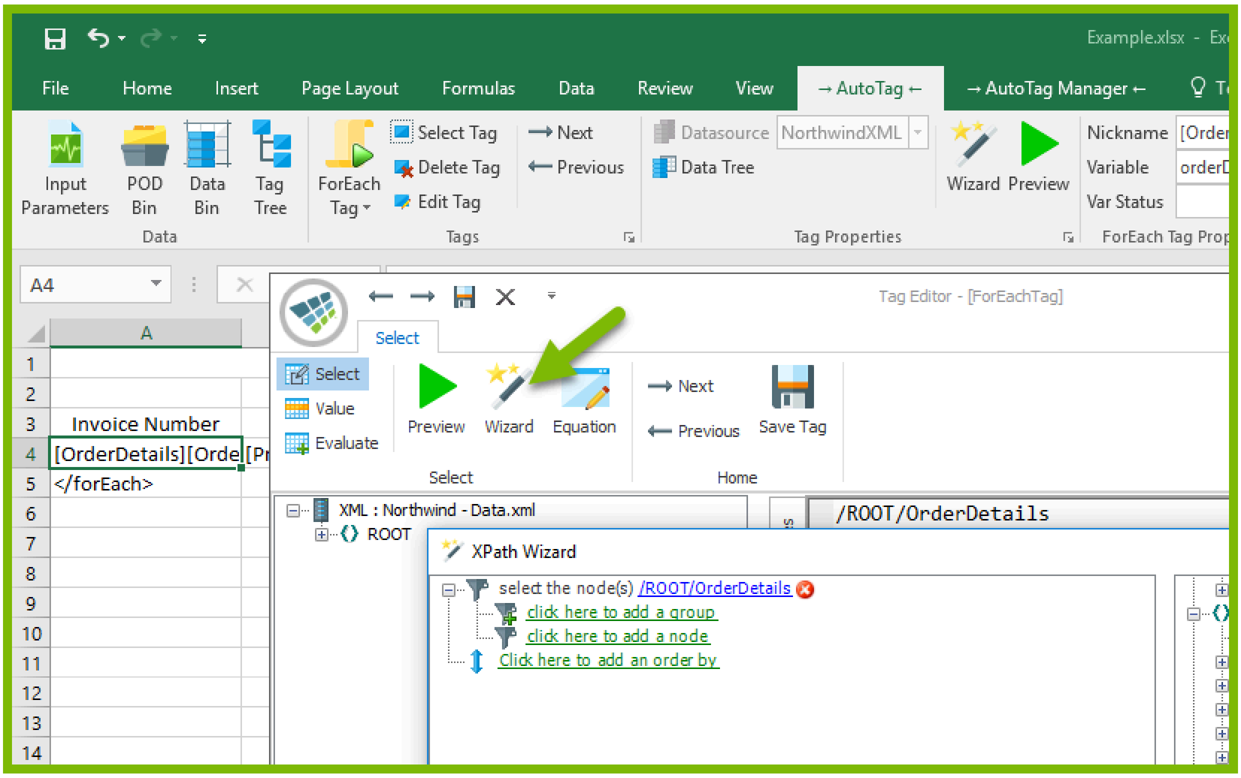
[OrderDetails][

</forEach>

Select the tag you wish to edit

[OrderDetails]

In the Tag Editor, click on the Wizard icon, which brings up the XPath Wizard.



A4

Invoice Number

[OrderDetails][Orde

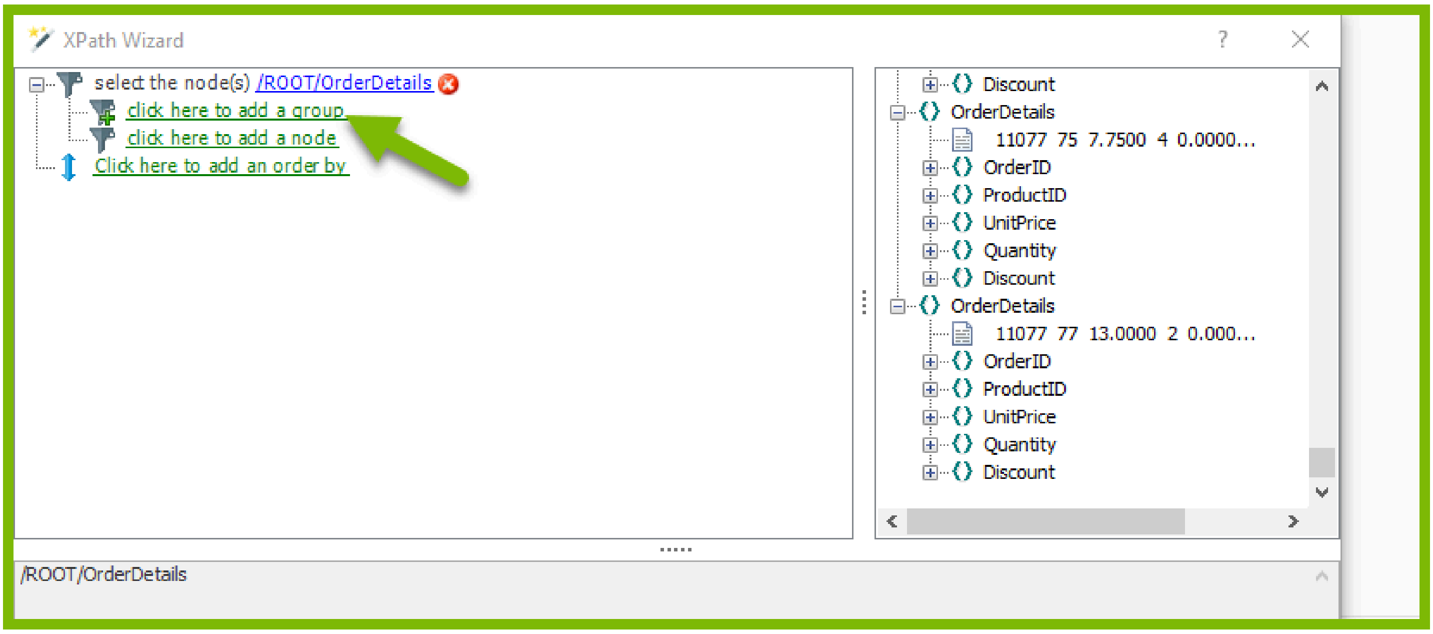
</forEach>

/ROOT/OrderDetails

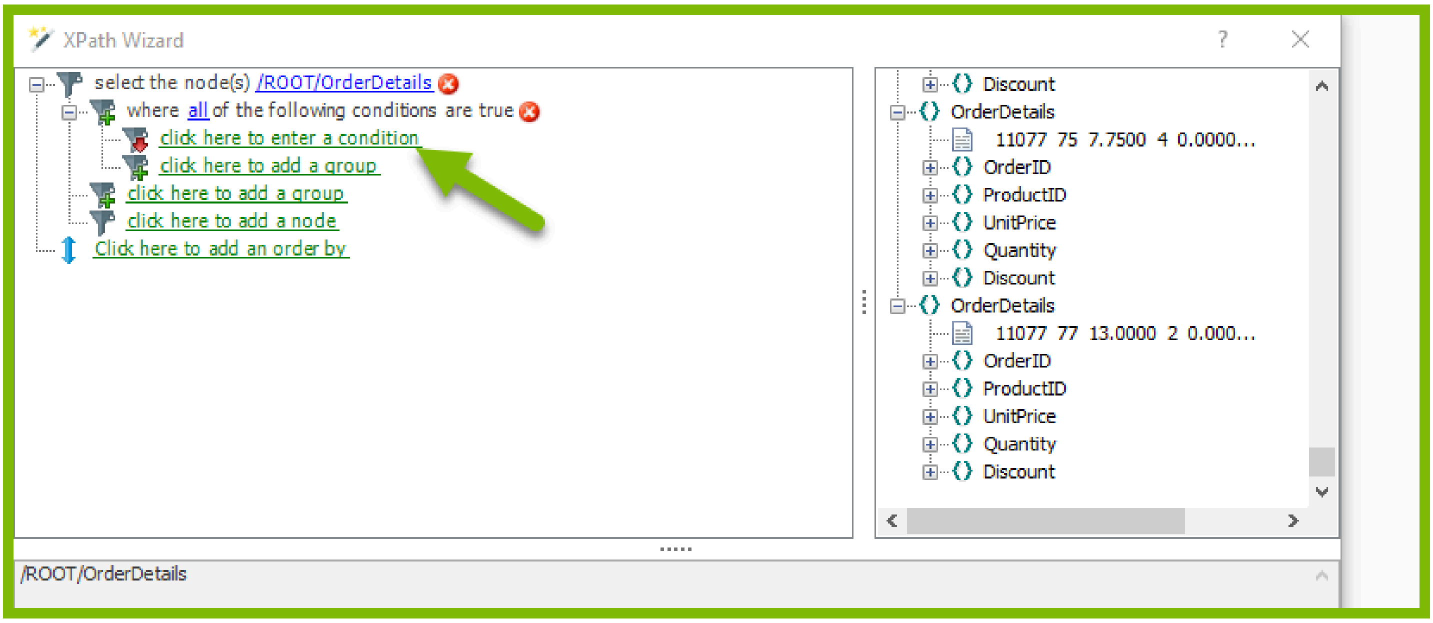
select the node(s) /ROOT/OrderDetails

In this example, since we are using an XML data source, we will use the XPath Wizard. But for other data source types, e.g. SQL or JSON, there are other Wizards.

In the Wizard Conditions Pane, click on the button click here to add a group.

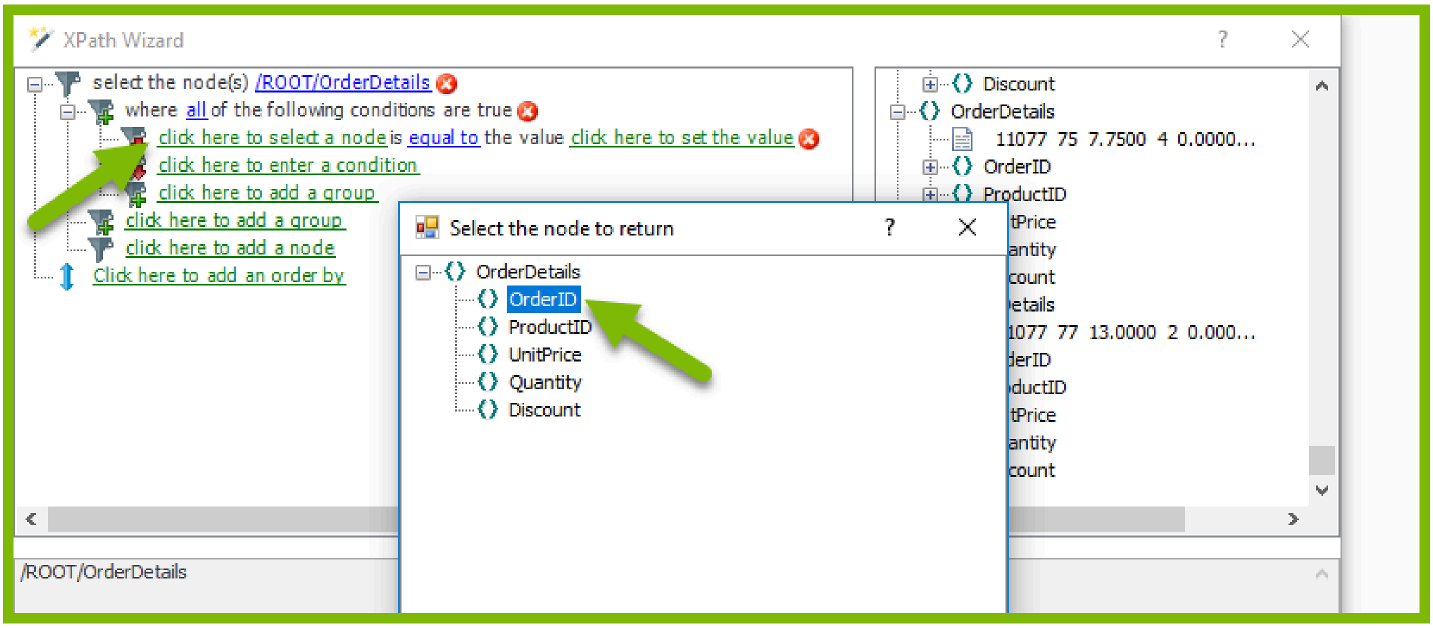


Click on the button click here to enter a condition.



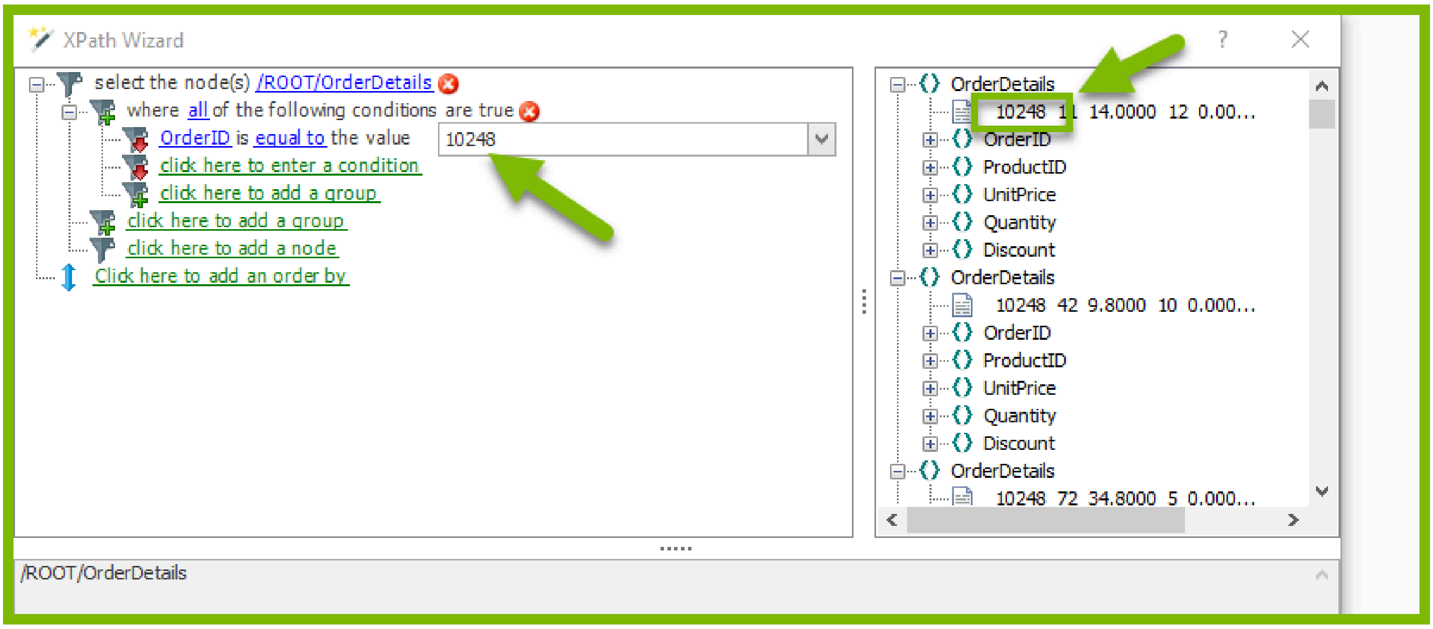
Click on the button click here to select a node, then select the subnode "OrderID". (Remember, OrderID is the invoice number.)

By default, the comparison is set to equal to, so you don't need to select a comparison.

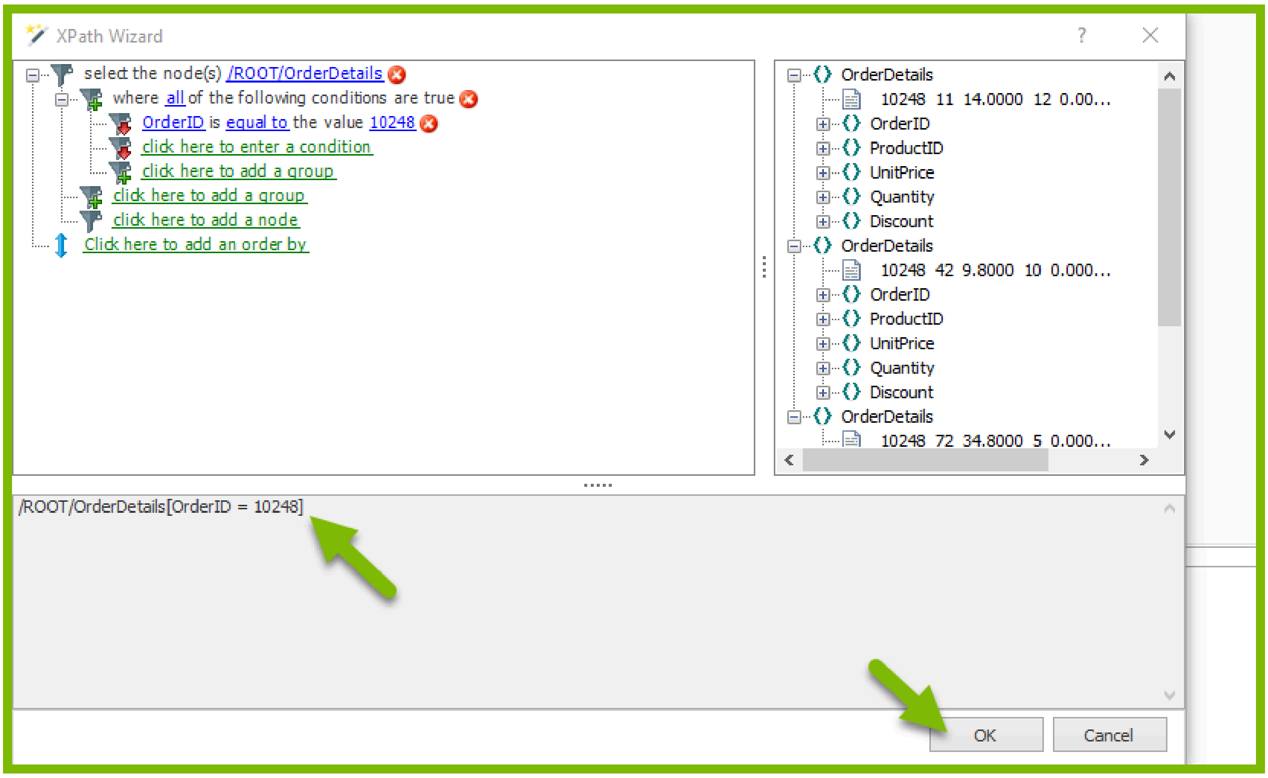


XPath Wizard
select the node(s) /ROOT/OrderDetails
where all of the following conditions are true
click here to select a node is equal to the value click here to set the value
click here to enter a condition
click here to add a group
click here to add a group
click here to add a node
Click here to add an order by
/ROOT/OrderDetails
Select the node to return
OrderDetails
OrderID
ProductID
UnitPrice
Quantity
Discount
Discount
OrderDetails
11077 75 7.7500 4 0.0000...
OrderID
ProductID
tPrice
antity
count
etails
1077 77 13.0000 2 0.000...
derID
ductID
tPrice
antity
count

Take a look at the various OrderID values listed in the Data Pane on the right. There's one for invoice number 10248, and that's the one we'll use in our example. Click on the button click here to set the value and in the text box, enter the number 10248.



After setting the OrderID value, you can see the entire select statement (the query that will be sent to the data source when you run the report) in the lower pane of the Wizard. Click on "OK" to close the Wizard.



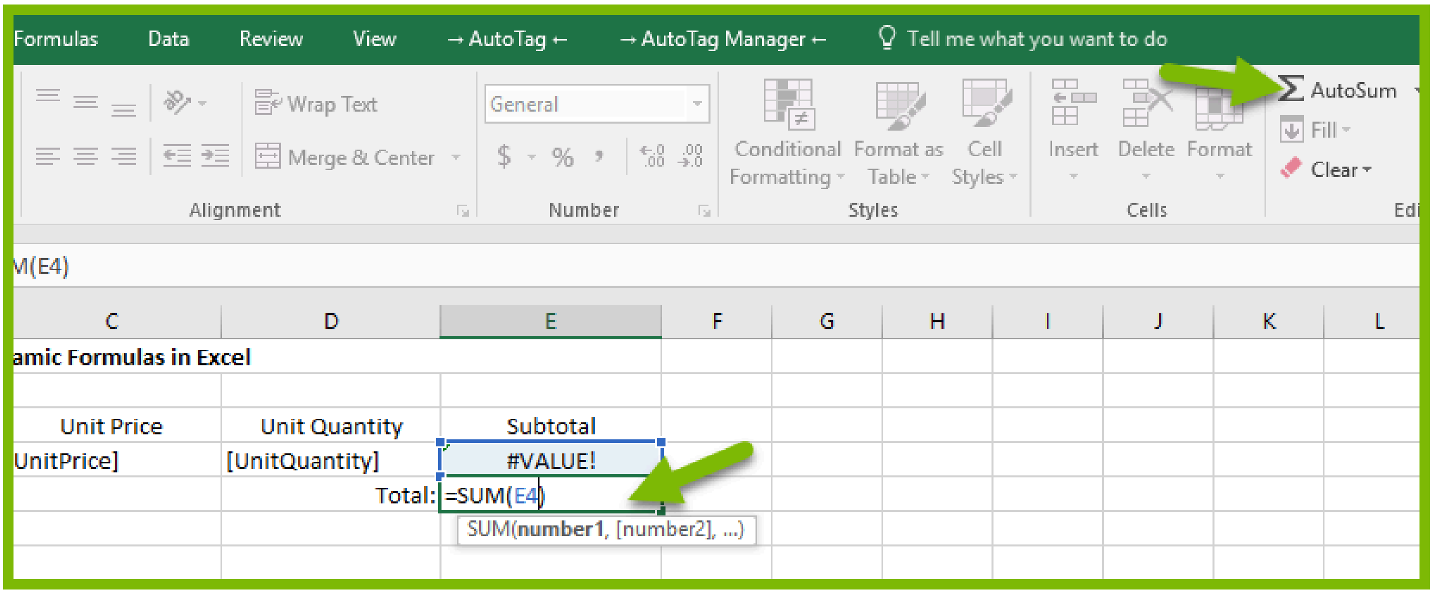
select the node(s) /ROOT/OrderDetails
where all of the following conditions are true
OrderID is equal to the value 10248
click here to enter a condition
click here to add a group
click here to add a group
click here to add a node
Click here to add an order by

```
OrderDetails
   10248  11  14.0000  12  0.00...
   OrderID
   ProductID
   UnitPrice
   Quantity
   Discount
OrderDetails
   10248  42  9.8000  10  0.000...
   OrderID
   ProductID
   UnitPrice
   Quantity
   Discount
OrderDetails
   10248  72  34.8000  5  0.000...
```

/ROOT/OrderDetails[OrderID = 10248]

Create a Total

Now let's use Excel's Autosum feature to create a total.
In our spreadsheet we'll create a total cell below the subtotal cell (it can be on the same row as your EndForEach Tag).
From Excel's Home tab, click on the AutoSum button. In the formula "=SUM()" put the location of the subtotal value, which in our example is E4, between the parentheses. Your template now looks like this:



M(E4)

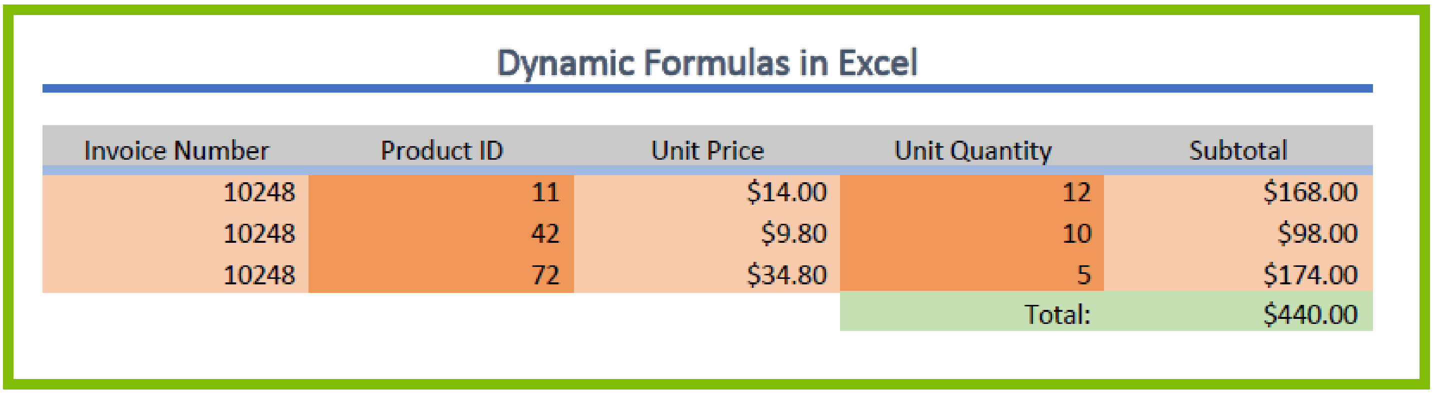
C	D	E	F	G	H	I	J	K	L
amic Formulas in Excel									
Unit Price	Unit Quantity	Subtotal							
UnitPrice]	[UnitQuantity]	#VALUE!							
	Total:	=SUM(E4)							

SUM(number1, [number2], ...)

Format the Table

Finally, use familiar Excel format commands, such as those in Excel's Home menu, to polish the table. Because columns 'C' and 'E' are currency, format the columns accordingly. Also, change the font, add colors, reposition text, place borders around cells, and otherwise enhance the look of the table.

Save your report template using Excel's Save command. Then, using the Output button in the Designer ribbon, click on the desired report format. In this example, we chose to view the report as a PDF file, and this is the result:



Dynamic Formulas in Excel

Invoice Number	Product ID	Unit Price	Unit Quantity	Subtotal
10248	11	$14.00	12	$168.00
10248	42	$9.80	10	$98.00
10248	72	$34.80	5	$174.00
			Total:	$440.00

Congratulations! You now have the basics of harnessing the power of Excel formulas in your report template. You may need to tweak your table dimensions a bit, as Excel does not let you auto-expand a cell as needed. For more information concerning XLSX output limitations please review Output Format Limitations.