How Do I Use Input Parameters?

This article explains how to create a variable input parameter inside of the Designer, and how to build a drop-down list of valid values. When users run the report, they will either choose one value from a drop-down list or enter a value in the input box.

Sample Templates

You can use Windward's Sample templates to practice using variables.

📄 Select_Variable-All_Data_Sources_Template.docx

📄 Select_Variable-JSON_Example.docx

📄 Select_Variable-OData_Example.docx

📄 Select_Variable-SQL_Example.docx

📄 Select_Variable-XML_Example.docx

The SQL sample template uses Windward's public MS SQL database to show an example of using variable values in a select statement. This sample shows a list of Orders that a user can select from a drop-down menu or enter directly in a field from the Northwind database. Examples for XML, JSON and OData are listed on further pages in the Select Variable-All Data Sources Template sample, as well as individually above.

Defining and Using a Variable

Step 1 - Create a Variable

1. First you will need to create your variable in order to use it in your template. This is done by selecting the Windward tab and clicking on the *Input Parameters* button which opens the *Input Parameters* (variables) window.
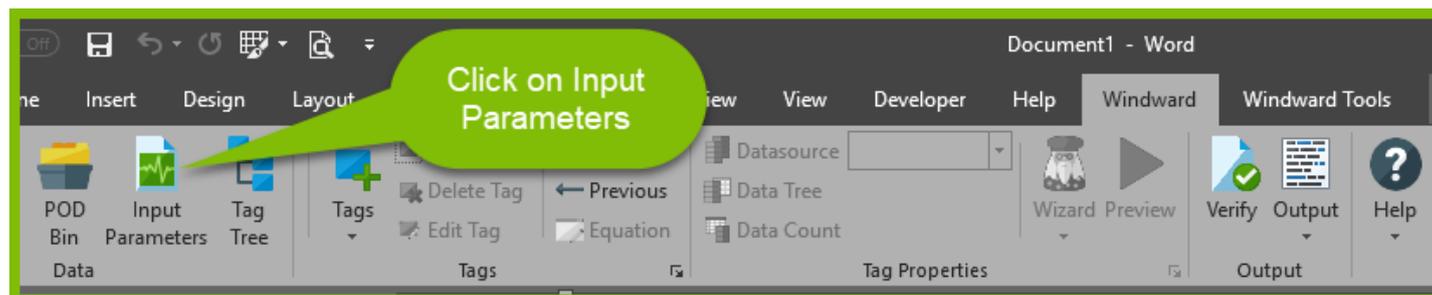
2. Next, a variable needs to be created by clicking on the *Add* icon and entering a name for the variable; in this example we'll use *Order*. Once created, your new variable Order will appear as a tab as shown in the image below. Each new variable will have a separate tab.
3. Once the variable is added, you need to specify the type of information that it will receive from your data source. For this example we will choose *Select* as the variable type. This will allow us to retrieve a list of orders from the data source, and allow a user to select one order from the drop-down box when the report is run. For a complete list of variable types refer to the section [Variable Type](#).
4. Then enter 10537 as a *Default* value to be presented to the user *first,* each time the report is run. This is a value that is known to exist in the data source. It also provides a good sample set of data to demonstrate in the report. Try using a few values to see what they return before settling on a default value.
    1. The *Default* value must also be defined for the *type* of information it returns. In this case we use an *Integer* value because it is a number.
    2. A default value also allows you to work with your variable in other tags in your document, especially using a *Select Statement Wizard*, as it will provide a value for your variable and returning meaningful results in the preview window.

5. Lastly, you should give your users a prompt for what type of information to enter, and how to enter it for the variable when the report is run. Enter the following text in the *Description:* field - *Please enter an order number.*
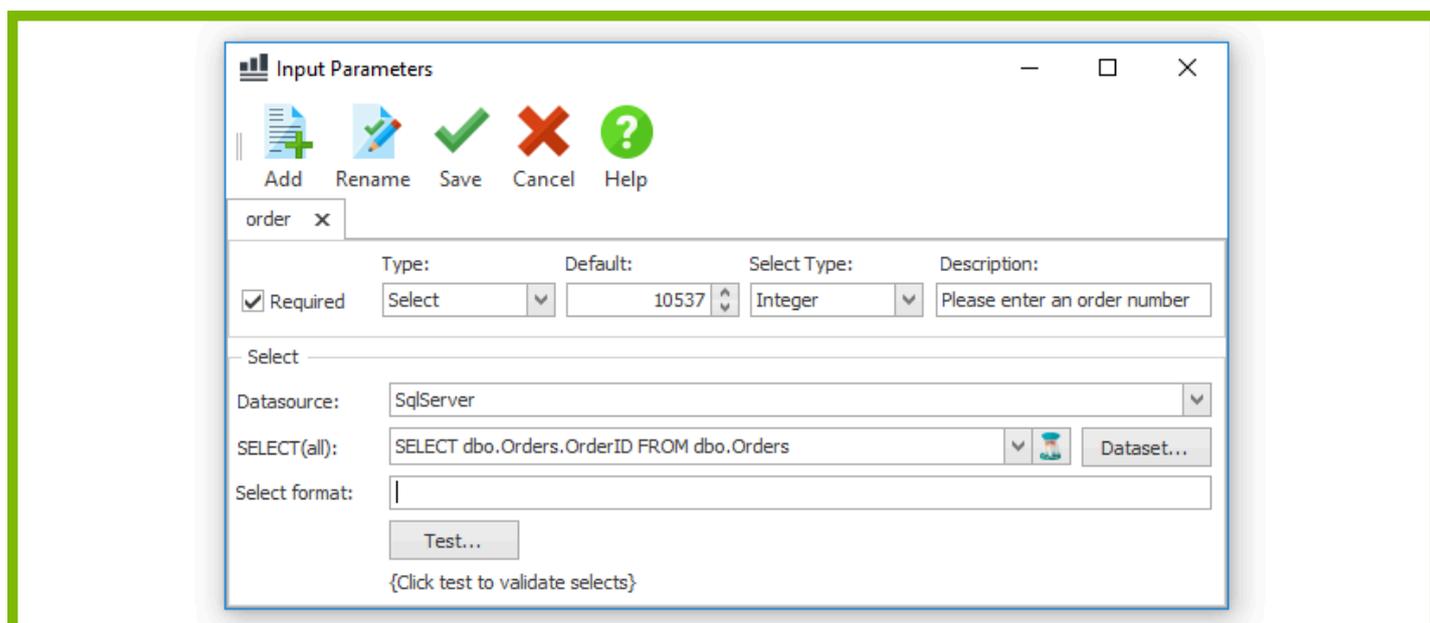
6. Then click *Save* to finish creating your new select variable *order* with a *Default* value of *10537*. Proceed to Step 2 below to build an interactive drop-down box of orders, or skip to Step 3 to test your variable.

Step 2 - Building Validated List of Values from Data Source (Optional)

When asking your end users to enter data that is required for your template to run, you should ensure that what data they enter will not create an error while running your template. As a best practice, you can optionally create a list of values retrieved from your data source for the users to choose from while running your template. This can be a drop-down box, an input field, or even a date picker to select a date from a calendar. In the example below you will build a drop-down box of 50 orders from your data source.

To list all of the orderID's from the orders in your data source, you would have a select statement that looks something like the following in the *SELECT(all)* field. In order to create this data selection statement, we will use the integrated Designer data selection wizard directly from the *Input Parameters* window.

```
SELECT Orders.orderID FROM Orders
```

1. While still in the *Input Parameters* interface, click on the data selection *Wizard* icon to open the SQL Wizard to assist in creating the data selection statement from the data source.
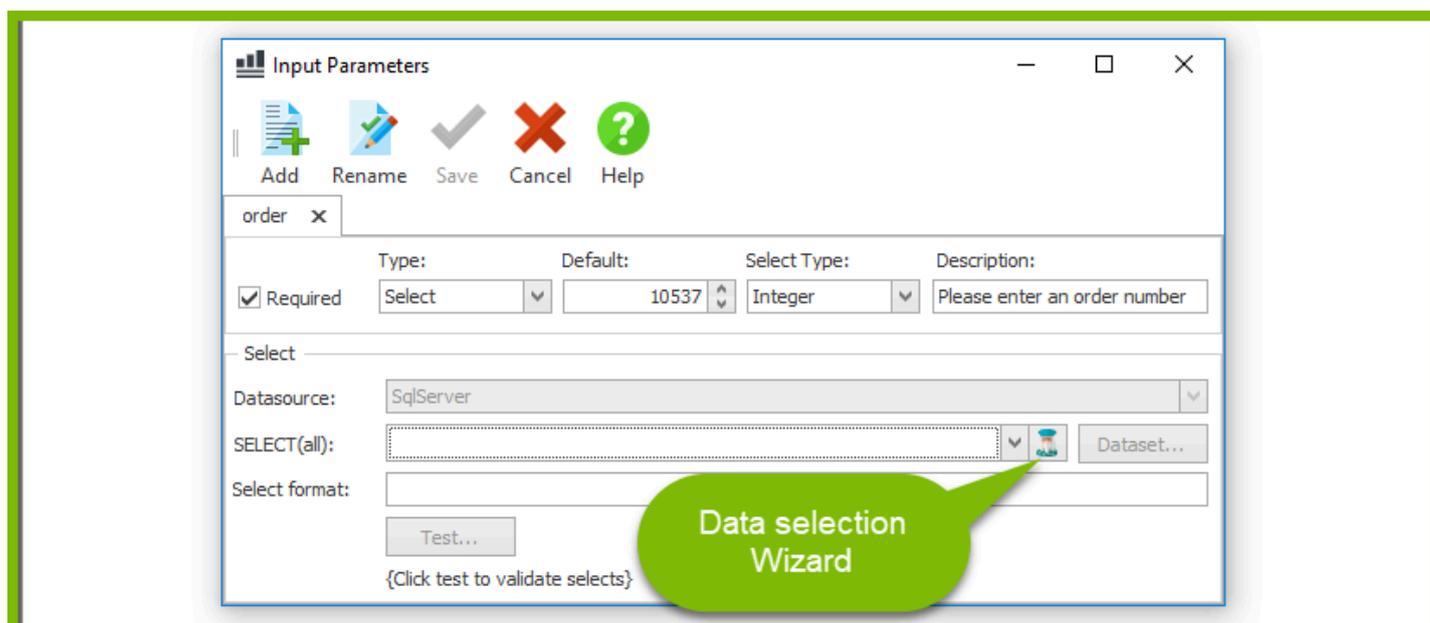
2. In the [SQL Select Wizard](#),

1. Drag the OrderID node into the Columns pane denoted by *marker #1* in the image below.
2. You'll notice that a list of Order IDs from your data set appear in the right preview pane.
3. (Optional) Check the *Distinct* box in order to prevent duplicate values from being returned.
4. The bottom pane shows the actual SQL query statement that you created by dragging and dropping!
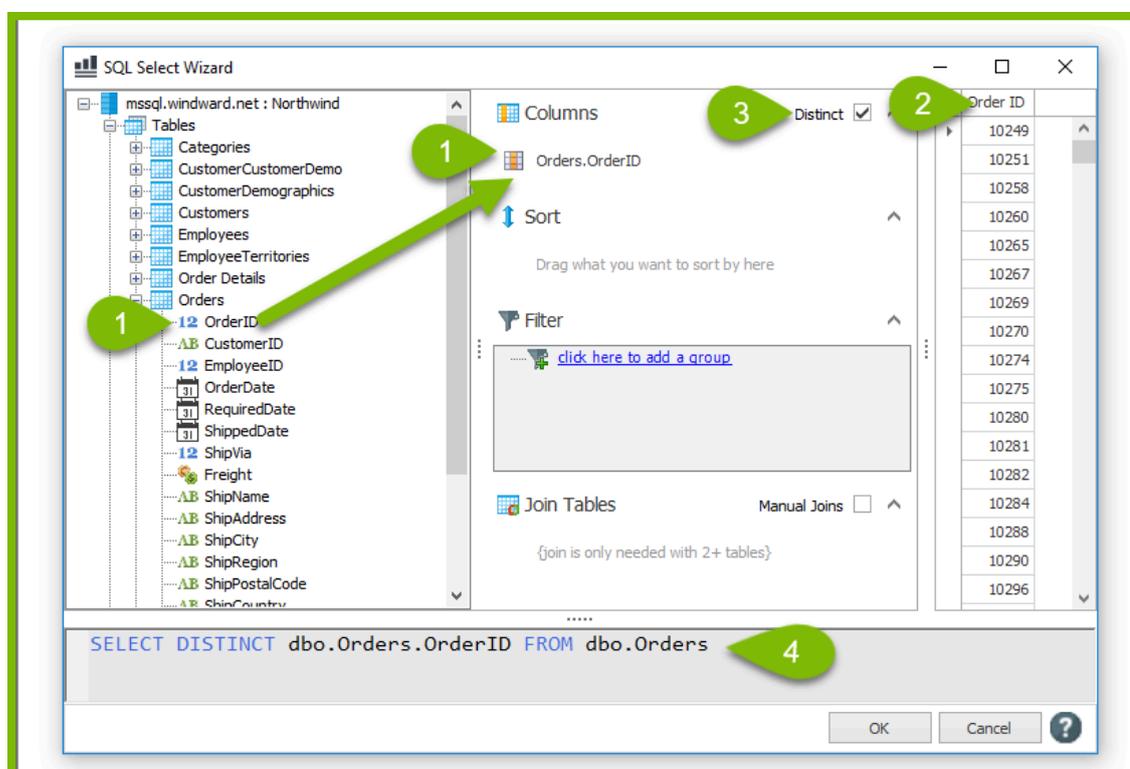
3. However, in order to demonstrate creating a drop-down box of orders in your data source you will need to limit how many orders are returned. In order to do this we have modified

the SQL statement by adding the text 'TOP 50' after the SELECT text in order to limit the returned orders list to only 50 orders. Your SQL statement from Step 2.1 should now look like the one listed below.

```
SELECT TOP 50 Orders.orderID FROM Orders
```
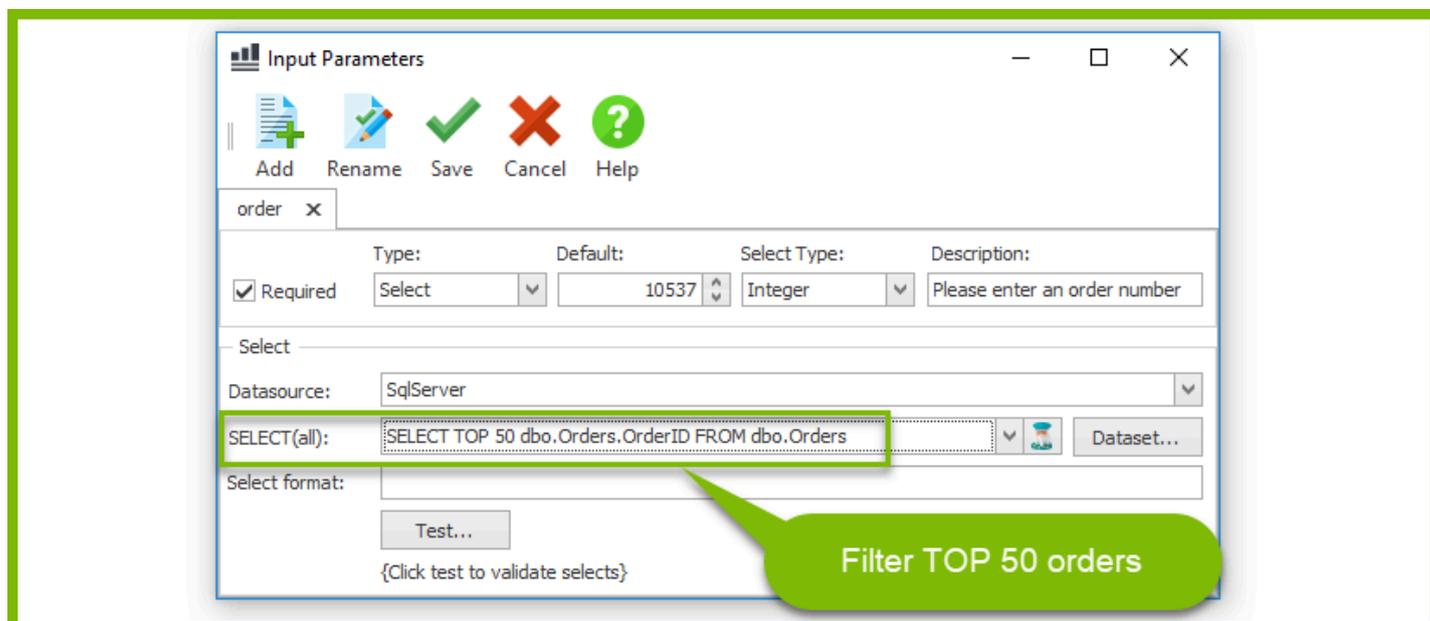

> ℹ️ *If less than 50 values are returned we present the user with a drop down list with all 50 (or less) options from the database. If the number of values returned are greater than 50, the end user is presented with a textbox instead, and they will then have to type in a value for the variable, which is checked if the value exists in the data source.*

Step 3 - Testing the Created Variable

Now that the variable is created, the last step is to test it against a real value to ensure that it will work when you run your report.

1. Clicking the *Test...* button will open the *Test variable select* window.
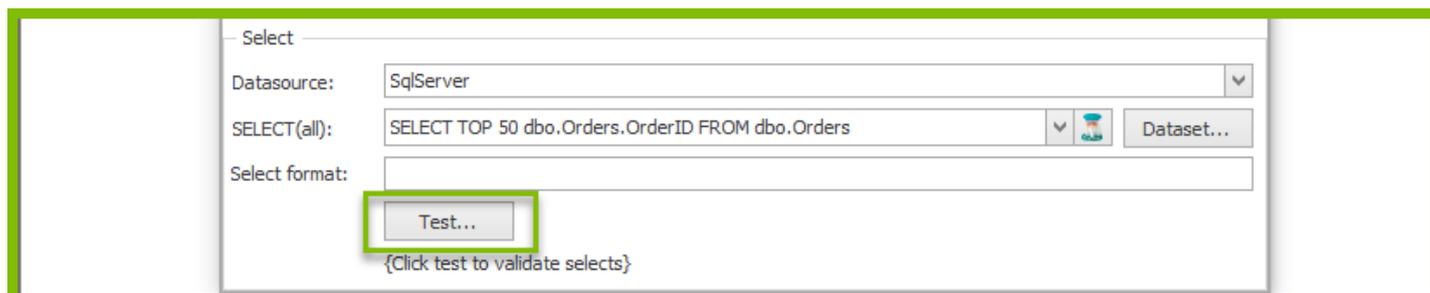
2. Here you can clearly see the *default value (Value to select: 10537)* that you provided earlier is listed as the first value to choose in the list. You can either enter a new value or use the default value.
3. Clicking *Test* will send the value of this variable (10537) to the data source to test if the value exists in the data source or not. If all goes well you will see that order 10537 exists in the data source and all tests were successful. If the order does not exist in the data source, then the output in the window will give you more information to help you troubleshoot your error.
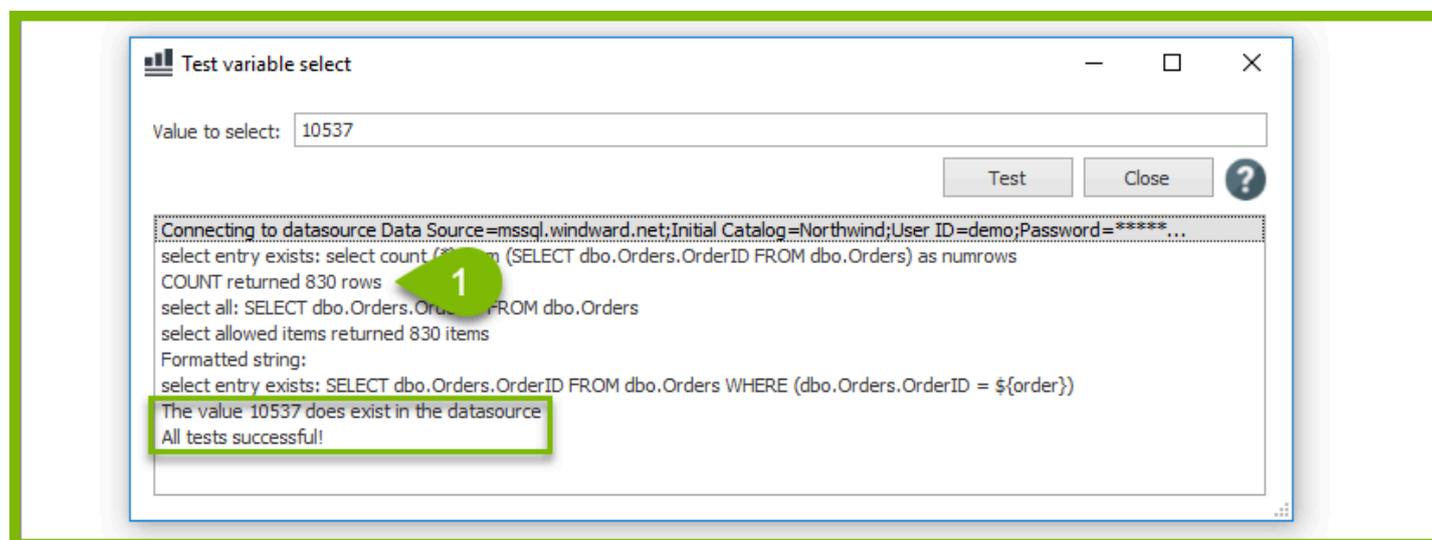

> **ⓘ** *The number of data items returned is also listed. In this case there are 830 orders in the data source (marker #1 image above). This means that without the previous TOP 50 limiter in your select, you would receive a normal input field to enter your data. Limiting the values returned to below 50 allows a drop-down box to be created where the end user can easily choose from a list of orders.*

4. *Close* the *Test variable select* window. Then click *Save* to finish creating your new select variable *order* listing the top 50 orders in a drop-down box with a *Default* value of *10537*.

Step 4 - Running the Template and Using Your Variable

1. Click Output and choose an appropriate Output format to run your report.
2. The *Run a Report* window will pop up, and prompt you to choose your order number from either the drop-down list (if you followed Step 2), or enter a number (between 10249 and 11061), or use the *Default* value *10537*.
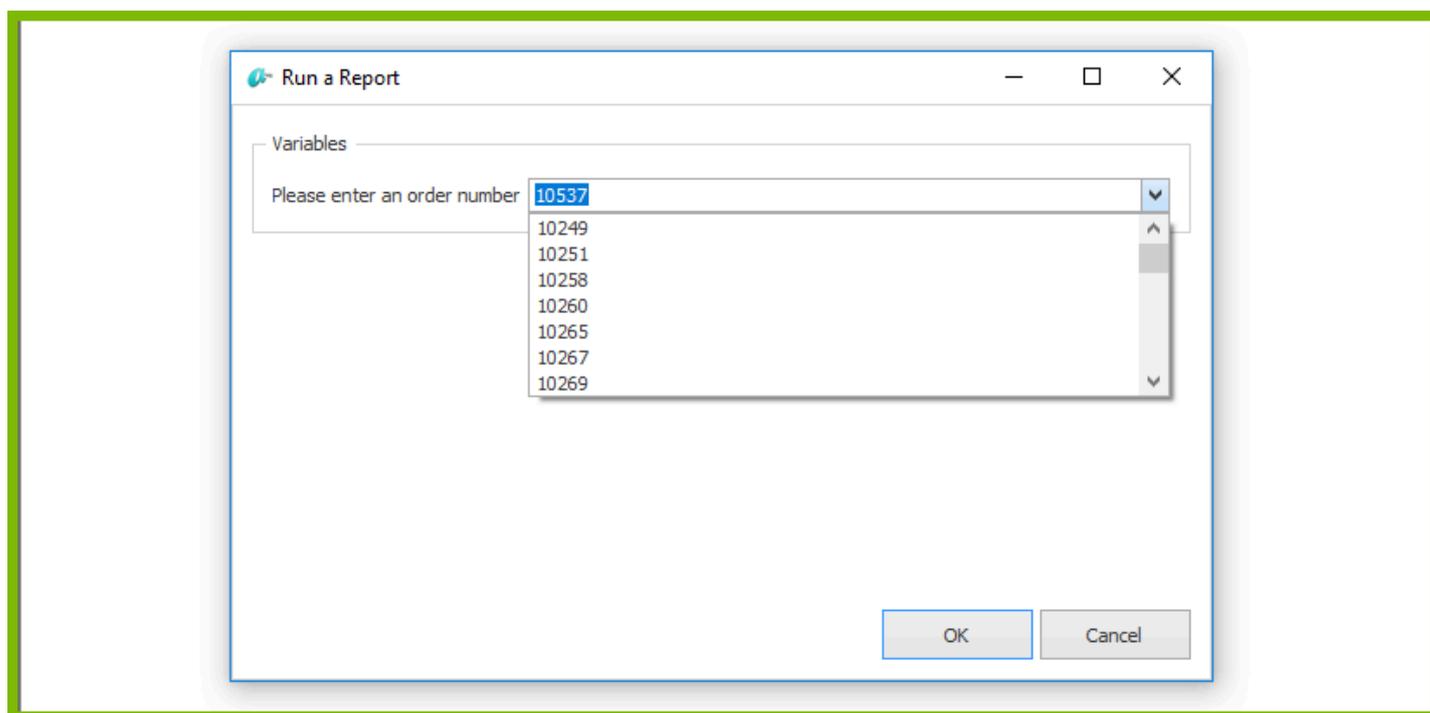
3. Your finished invoice should be printed as shown below for the order number you selected (10537 in this case).

Order Number: 10537

Richter Supermarkt
Michael Holz
Grenzacherweg 237
Genève, 1203
Switzerland

Hello Michael Holz,

Thanks for your order!

You have ordered the following:

Product	Quantity	Price	Sub Total
Gorgonzola Telino	30	$12.50	$375.00
Manjimup Dried Apples	6	$53.00	$318.00
Escargots de Bourgogne	20	$13.25	$265.00
Mozzarella di Giovanni	21	$34.80	$730.80
Röd Kaviar	9	$15.00	$135.00
Total Price			$1,823.80

Advanced

Windward Tags That Can Use Variables

Each of the Tags below will allow you to use a variable you have created in your template. The variable can be used to output its value (Out Tag or Import Tag), compare its value and make a decision (If and Case Tag), or filter data returned based on a value (ForEach, Query, Out, If, Set, Case or Chart Tag).

Out	Query	Link
ForEach	Set	Chart
If	Case	Import

Tags That Can Set var Property

Windward Tags that can assign their returned-output value to a Tag-assigned-variable using the *var* property.

ForEach	Query	Switch
Out	Set	

Different Ways to Use Variables

When creating a variable in Designer, creating a *default value* is very important. This value will be used when your variable is evaluated in other Tags in order to return a preview of results. The default value will also appear first when the report is run in either a drop-down list, or a data field.

When a variable is created, Windward is not aware of the type of data it will return. The variable can also be assigned to a select statement which will return a list of values that the end users can select when they Output. Since the form is generated dynamically at runtime, it can prompt the end user in two different ways:

- Drop-down List of Values
  - If the list of values returned is 50 values or less, a drop-down box is filled with these values enabling the end user to easily select which value they want to assign to the variable.

- Data Field

- The second type of form is a blank input field in which the end user will need to manually enter a value for the select variable. This form appears if the list of values returned is greater than 50. Once a value is entered, it is quickly checked against the data source to ensure it is a valid value in order to generate the report. If the value entered is not valid an error message is displayed in the variable form and the end user is asked to enter a new value.
- Date Picker
    - The user will be presented with a calendar style date picker in order to intuitively select a date to run the report. The date picker will only appear when the type *Date* is selected.

## How to Reference a Variable in Other Tags

Once a variable has been created in the Designer, it can be referenced in other Tags, Wizards and selects by its variable name via the following syntax:

```
${VARIABLE_NAME}
```

where *VARIABLE_NAME* is the text value assigned in the *Input Parameters* window, or in any Windward Tag where the property *var* is found. The variable name will then appear in the data tree where it can be added by a drag/drop action in the *select window* of the *Tag Editor*. It can also be used in the Select Wizard for filter operations.

> ℹ️ *Keep in mind that other variable values can sometimes appear in the data tree depending on where your Tag is located in the document. For example, if your Tag is inside two ForEach Tags then you would see both variable names referring to each ForEach Tag as well as your variable name.*

Variable (Define Parameters) Interface
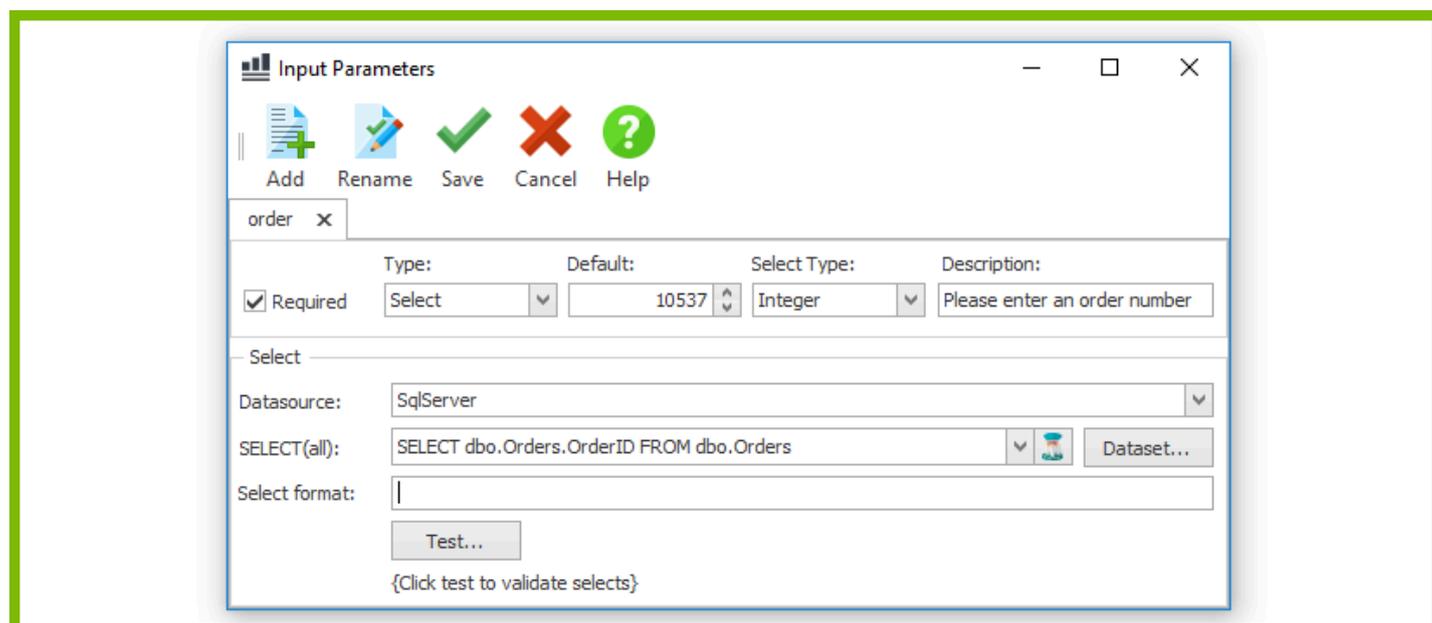
- *Add* - adds a new variable by creating a tab in the *Input Parameters* window. Each new variable is created on a new tab.
- *Rename* - allows you to rename the currently selected variable. Renaming takes place in the tab itself.
- *Save* - saves current variables and their properties.
- *Cancel* - cancels current variable and property changes. This will close the *Input Parameters* window.
- *Help* - opens the help context window for descriptions of the variable interface.
- *Required* - this checkbox, when checked, will require a value to be entered by the user before the report can be run. This is applied on a per variable basis.
- *Default* - This is the value that is returned whenever your variable is used in the template. It is a best practice to set this value as you will then be able to preview results when your variable is used in other Tags and Tag selects statements. This must be a value that is known to exist in the data source in order to work properly.
- *Description* - Text entered in this field will appear when the user runs the report. This should be something that will assist the user in the type of information, and in what format to input it.

Variable Types

Variables are input parameters entered by the user when the report is run. In order for Windward to understand what information you are entering you will need to tell it what type of information a user will enter. This needs to be assigned for the *Type:* field and the variable type can be any of the following.

> ℹ️ *Note: if the Type: field is set to 'Select' then you will need to set the Select Type: field as well.*
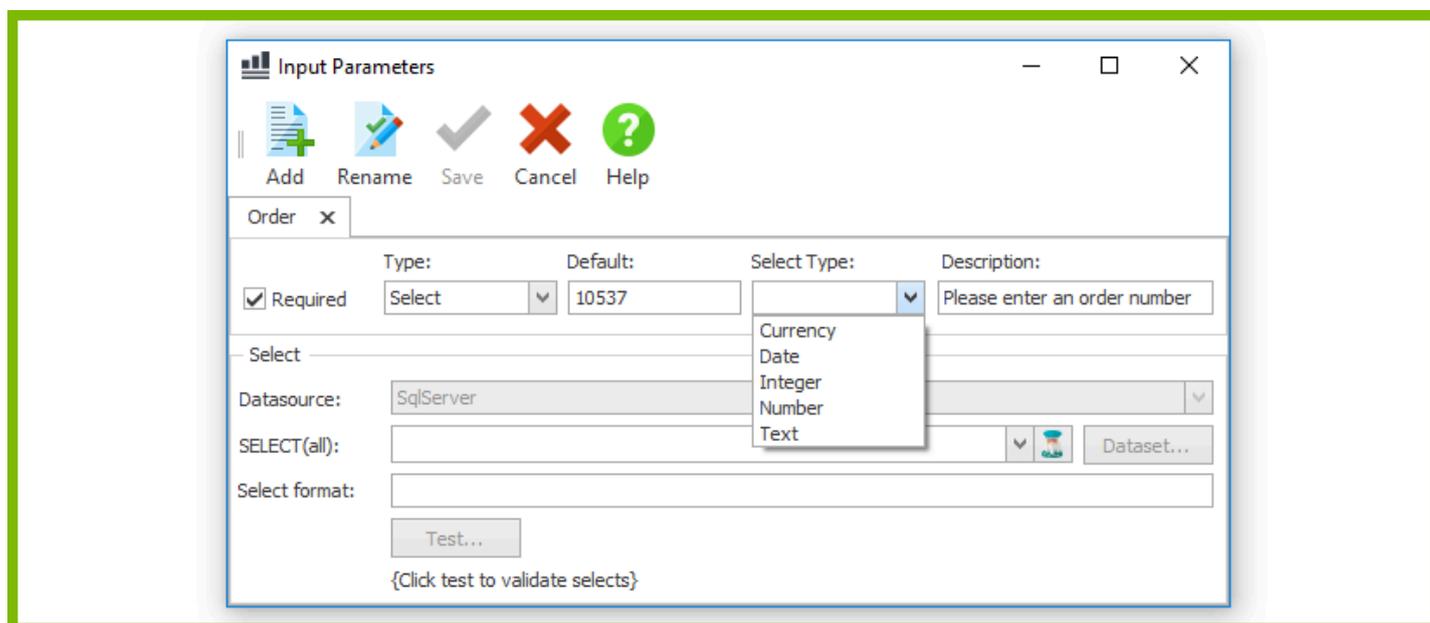

- *Currency* - the data returned is formatted as a currency type with periods and commas used as separators depending on your region (i.e., for US regions it would be X,XXX,XXX.XX).
- *Date* - When the *Type* is set to *Date* then a calendar date picker is presented to the user. The data returned is formatted as a date based on your region format settings, (i.e. a US formatted date would appear as MM/DD/YYYY).
  - *Offset* - this field only appears when type *Date* is selected. This allows you to set a default value for the date variable. It also allows you to set constantly updating date offsets.
    - *Specified Date* - this will convert the *Default:* field to a calendar date picker set to the current date. You can then specify a date to be used as the default date shown to the user when running the report.
    - *Today* - will always set the default date shown to the user as the current date.
    - *Start of week* - will always set the default date shown to the user as the date of the start of the current week.
    - *Start of month* - will always set the default date shown to the user as the date of the start of the current month.
    - *Start of quarter* - will always set the default date shown to the user as the date of the start of the current quarter.
    - *Start of year* - will always set the default date shown to the user as the date of the start of the current year.

- *Integer* - This type allows for both positive and negative numbers.
- *Number* - This type allows for only positive decimal numbers.
- *Text* - This type is a regular string of characters. This is typically used if you do not need to do calculations on the input data.
- *Select* - This type allows you to define a SQL select to retrieve a list of values from the data source that a user can select. If the items returned are less than 50, a drop-down box of data source values is provided, and if they are over 50 an input field is provided. The data entered is checked against the data source for validity.

> A select variable has two uses. When running a non ad-hoc report, it will list the possible values for the user to select one, to then use when running the report. This value is directly substituted into the select. The second use is for an ad-hoc report. In this case it is used to display a list of all values (or the first 10,000 if there are more than 10,000), and to perform conditional values in the selects when the report runs.
>
> The fields below in the Select section are *only* enabled when the *Type:* field is set to the value *Select*.
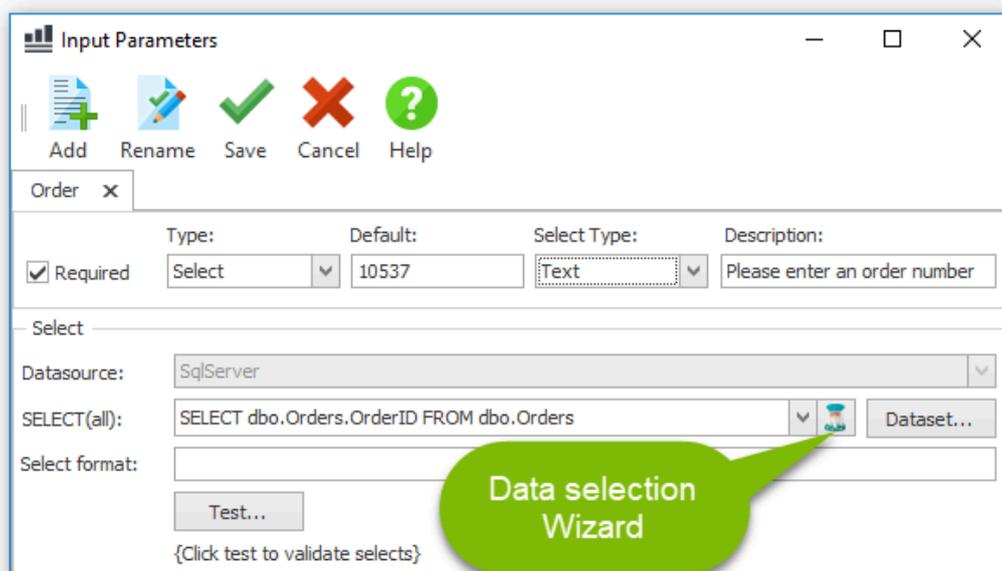
- *Datasource* - this defines which data source the select in *SELECT(all)* field will query in order to return a list of valid values for the user to choose from when running the report.
- *SELECT(all)* - the data selection query that will return the list of values your users will choose from to assign to the variable when the report is run. This can be written by hand or built with the data selection *Wizard* button at the far right of this field.
- *Wizard icon* - launches the data selection Wizard to graphically build the list of values your users will choose from to assign to the variable when the report is run.
- *Select format* - Enables you to define how each row of data returned will be displayed in a list to the user.
  - *<blank>* - If there is no value entered then Windward will use variable arguments in the order {0}, {1}, ... for each column of data returned.
  - *Value Example* - {0:N0} if the COUNT returned for the number of rows is 10249, then this would display that number as 10,249. The syntax here is the C# string.Format() syntax - refer to that for complex formatting.

- *Dataset...* - allows you to define which columns returned from the select should be used for variable value selection. Enable a column to be viewable by checking the box next to it.
  - *Set Key* - when two or more columns are returned for the select, it allows you to choose which column should act as the primary key. This is useful when you have multiple values that may be the same, and you need to specify a unique count of all values. The currently set key column will have a *p* icon next to it in the list.
  - *Save* - saves all changes made.
  - *Cancel* - closes the window and cancels all changes made.
  - *Help* - launches the Designer Ohana help page for this feature.
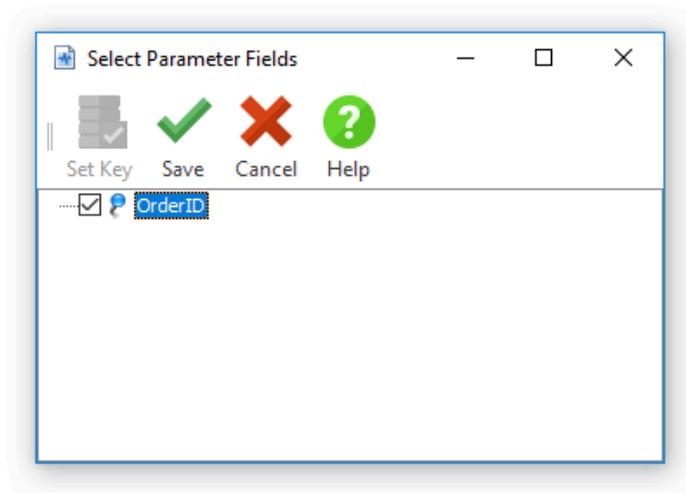
- *Test...* - this will launch the *Test variable select* window. It enables you to see the values, and default value returned from your built select, and to test a variable value to ensure your report will run properly. All errors and successes will be noted in the output window.