Wizdocs reports generator: Formula Tip sheet

A compilation of basic and advanced formulas to operate the Wizdocs Reports Generator

Updated August 2017



Wizdocs Reports Generator Formulas

# Queries[[1]](#footnote-1)

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| ReportItems | Used to insert records based on specific records selected via checkmarks or views. Will insert only these records into the table. | |  | | --- | | **Column name** | | Column value | |
| GetView | Used to insert records based on a view from within DealManager. Note that all records in the view will be inserted even though only certain records may have been selected from generation screen. | GetView("LIST NAME"; "VIEW NAME") |
| GetUniqueItems | Used to only produce unique items within a list based on the unique item in a particular column. Useful in linked lists such as the Flagged Issues list within DealManager when you want to only include a single record but pull in linked records via a related list. | GetUniqueItems(ReportItems;"COLUMN NAME ")  In the above example, the query starts with the records produced via the ReportsItems function (which produces records selected via checkmarks or views) then produces only the first record based on the COLUMN NAME if there are two or more records matching the value of the COLUMN NAME. |
| Find Items within GETUNIQUEITEMS Operator | Locate items in a list based on the value of one or more columns. Display all records matching that criteria. This formula variation is used when inside a GETUNIQUEITEMS row parameter. | FindItems("LIST NAME"; "COLUMN NAME"; COLUMN NAME)  Note that you repeat the name of the column again as the third variable in the expression. Also note that in this variation the third variable is not surrounded by quotation marks. |
| FindItems | Locate items in a list based on the value of one or more columns. Display all records matching that criteria.  Place this formula via a rich text content control around a row where you want the records inserted. Within the row columns insert columns. | FindItems("LIST NAME"; "COLUMN NAME"; "Text that you are searching") |
| MultiString Queries[[2]](#footnote-2) | String multiple GetView and FindItems parameters together to create a multi variable query. Additional parameters string together as "equal" conditions and combined with "and" operators. This feature makes it easy to create complex filters within the Word template without having to first create a DealManager/SharePoint view. | GetView("TestList"; "All Items"; "TestBool"; true)  In the above example, all records contained in the "All Items" view of the "TestList" list are pulled. Then only records where the column "TestBool are equal to "true" are produced.  GetView("TestList"; "All Items"; "Fruit"; "Apples")  Additional choice parameter may also be specified. In the above example, all records contained in the "All Items" view of the "TestList" list are pulled. Then only records where the column "Fruit" are equal to "Apples" are produced.  Unlimited numbers of additional parameters may be strung together.  The same multistring query may also be used in "FindItems" queries.  FindItems("TestList"; "TestChoice"; "Group1") – same as before (single equal condition)  FindItems("TestList"; "TestChoice"; "Group1"; "TestBool"; "0") – two conditions.  Formula with dates example:  GetView("TestList"; "All Items"; "ExpirationDate"; Today()) – dynamic date used  FindItems("TestList"; "ExpirationDate"; Today()) – dynamic date used  FindItems("TestList"; "ExpirationDate"; "2016-01-10") – static date used. |

# Basic Columns

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| Basic Colum | Drop in a column using the display name or the internal name of the column. If the display name contains special characters then use the internal name or enclose the display name of the column using brackets [ ] | [ColumnName] |
| Lookup Column | See below. |  |
| Me | Retrieves currently logged in full name. | [Me] |
|  | ListTitle, ListUrl, ItemUrl. |  |
| ListTitle | Inserts the name of the list from which the record is located. | ListTitle |
| ListUrl | Inserts the full URL of the list which the record is located. | ListUrl |
| ItemUrl | Inserts the direct URL of the item | ItemUrl |
| SiteUrl | Inserts the URL of the DealManager room. | SiteUrl |
| SiteTitle | Inserts the title of DealManager room (as displayed at the top of the application window). | SiteTitle |
| ViewItemPropertiesUrl | Inserts address of the "display" page of the record. May be used with the Hyperlink formula (below). | ViewItemPropertiesUrl |
| EditItemPropertiesUrl | Inserts address of the "edit" page of the record. May be used with the Hyperlink formula (below). | EditItemPropertiesUrl |
| Hyperlink[[3]](#footnote-3) | Used to insert hyperlinks and embedded descriptions within Word reports. | Hyperlink(url; description)  Hyperlink(clientURL; clientURLdescription)  *! Note that you should use a Rich Text Content Control when using the Hyperlink formula.*  Note that DealManager/SharePoint column names should be substituted for "url" and optionally "description."  Either value may be substituted for static values by surrounding the text with quotation marks.  Use with the "ViewItemPropertiesUrl" and "EditItemPropertiesUrl" variables to insert links to the edit and display pages for each record.  Hyperlink(EditItemPropertiesUrl; "Click to edit record")  *!* *Note that before you use the "hyperlink" formula in a Word template, you must manually insert a hyperlink (any will do) so that the "hyperlink" style within Word is added to your template.* |

# Math Formulas

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| Count | Count the number of items appearing in a view. Useful when creating a summary report or displaying statistics. Create a view in DealManager with desired filters. Then, use the "Count" feature to display the results in any report. | Count(GetView("List Name"; "View Name")) |
| Basic Math: Add, Subtract, Divide, Multiply | Use math operators to add, subtract, divide or multiple any series of COUNT formulas. Useful for generating statistics and analyzing information. You may string together several math operators. Common Operators  * + Addition. * - Subtraction. * \* Multiplication. * / Division * & Concatenation – Used for combining strings together. Example: "The" & " table" Result: "The table" | 100\*Count(GetView("List Name"; "View Name"))  In the above example, the COUNT of the view "View Name" is being multiplied by 100 via the multiply "\*" operator.  You may use the formulas to manipulate the numerical data.  For *Concatenation*, you may string together the text output of two fields and combine them into one content control. As an example, you could combine “First name” and “Last name” fields to appear in one content control. |
| Round | This will round the resulting number to the nearest decimal place specified within the ROUND operator. Useful for displaying percentages or in connection with the basic math operators to avoid number fractions. | =Round(100\*Count(GetView("Diligence Summaries"; "Change in Control Issues"))/ Count(GetView("Diligence Summaries"; "All Summaries"));0)  Note that the ROUND operator surrounds the expression via parenthesis. The "0" at the end of the ROUND expression designates the number place used in rounding. The "0" indicates that we desire rounding to the nearest whole number. |
|  |  |  |

# Date Functions

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| Now() | Retrieves current date and time. |  |
| Today() | Retrieves current date. |  |
| FormatDate(date, format) | Converts date to text representation using specified format ("d", "t", "yy", "yyyy", "MMMM", "MM", "dd",). It also can be used to extract part of a date. | FormatDate(Today, "yyyy")  This will result in 2012. |
| Today + 1. | Calculating tomorrow's date: Today + 1. |  |
| FormatDate(Now, "t"). | Current time: FormatDate(Now, "t"). |  |

# Lookup Columns

The Wizdocs Reports Generator has the ability to access any linked list via a lookup column. This allows for the retrieval of information contained in related list or information kept in a centralized list.

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| LookupColumn:[ColumnName] | Will grab the column value from a related list via a lookup column. | DocLink:Title  LookupColumn:[ColumnName] |
| Lookup Column Stringed | You may string several lookup columns together to grab a column value to a related list linked to yet another related list. | LinkedA:LinkedB:Column1 |
| IfEmpty | Used to insert default text when a value is empty. For example, you could insert the text "N/A". | IfEmpty(COLUMNNAME;"Text entered when the value is empty.") |

# IfEmpty Default Text

The Wizdocs Reports Generator has the ability to access any linked list via a lookup column. This allows for the retrieval of information contained in related list or information kept in a centralized list.

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| IfEmpty | Used to insert default text when a value is empty. For example, you could insert the text "N/A". | IfEmpty(COLUMNNAME;"Text entered when the value is empty.") |

# Deleting Sections

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| Delete if column empty | Used to delete column or section if section if value is empty. | IfEmpty(COLUMNNAME; DeleteControl()) |
| Delete when criteria is met | Used to delete section if column name equals a certain value. For example if Governing Law = California.  May also be used with additional operators or combined with strings via NOT, OR, and AND. | DeleteControlIf(COLUMNNAME = "Value")  DeleteControlIf( NOT(COLUMNNAME = "Value") )  *Note in the above example where NOT is used.*  DeleteControlIf( AND (COLUMNNAME = "Value");(COLUMNNAME = "Value") )  *Note in the above example two strings are linked together with AND.* |
| Multi Delete based on a string of forumlas | You can delete a section based on a series of related or unrelated formulas. This is particularly useful if you are deleting a section based on whether a series of field values are blank. | DeleteControlIf( AND( IfEmpty(ColumnToCheck1; True; False)=True; IfEmpty(ColumnToCheck2; True; False)=True) ) |

# Dynamic Filtering Using Views

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| Use a view to dynamically add content in Find Commands | There are cases where a record view will contain a sub lookup. For example, if you are using a table to lookup records which are linked via a connected or related table. In some cases, a more complex filter and sort will be needed that can be achieved using the “Find” command via our Word formula.  Instead, you may create a view from within DealManager. When creating a view, you may insert the “find” filter criteria using curly brackets “{“ to enclose the value for which the filtered values should be compared against. See the example screenshot. |  |
|  |  |  |

# Formatting

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| SubString(text, startPosition) | Retrieves a substring from text. The substring starts at a specified character position and has a specified length. First character is at 0 position. | Example: SubString("abcde", 1)  Result: bcde |
| SubString(text, startPosition, length) | Retrieves a substring from text. The substring starts at a specified character position and has a specified length. First character is at 0 position. | Example: SubString("abcde", 2, 1) Result: c |
| Lower(text) | Converts text to lowercase. | Example: Lower("The Table")  Result: the table |
| Upper(text) | Converts text to uppercase. | Example: Upper("The Table")  Result: THE TABLE |
| PadLeft(text, totalLength, symbol) | Right-aligns the characters in text, padding with spaces or symbol on the left for a specified total length. | Example: PadLeft("123", 5, "0")  Result: 00123 |
| PadLeft(text, totalLength) | Right-aligns the characters in text, padding with spaces or symbol on the left for a specified total length. | Example: PadLeft("123", 5)  Result: 123 |
| PadRight(text, totalLength, symbol) | Left-aligns the characters in text, padding with spaces or symbol on the left for a specified total length. | Example: PadRight("123", 5, "0")  Result: 00123 |
| PadRight(text, totalLength) | Left-aligns the characters in text, padding with spaces or symbol on the left for a specified total length. | Example: PadRight ("123", 5)  Result: 123 |

# Images

|  |  |  |
| --- | --- | --- |
|  | Description | Example |
| Image(imageUrl) | Finds an image by Url. Full or site relative Url can be specified. Image must be located in Wizdocs or SharePoint environment. | Example: Image("http://site/images/contoso.png") |

# Group By Functionality (introduced in Version 3.7.16.06081)

In prior versions, grouped reports could only be generated by referencing lists with a grouped view. Using this method, grouping were limited to two levels, did not provide for dynamic filtering (e.g. selecting a group of records via column filters), and could not access related list information for columns that were used to create the groups. This newly released version allows users to create grouped views without any of these limitations. See the example below with sample content controls and formulas.

1st level group (Country): 1st Group Category Title and Count

*[Note that in the you may now combine multiple formulas and column information in a single content control by using the "&" character. Also, the "GroupBy" data pull may now reference "ReportItems" which allows for pulling dynamically filtered reports. Instead of "ReportItems" you may also use any data pull formula method including "FindItems" and "GetView".]*

GroupTitle (Country): Click here to enter text.

GroupItems (Contracts grouped by Country): Click here to enter text.

Count (Contracts grouped by Country): Click here to enter text.

GroupTitle & Count: Click here to enter text.

FindItems Count (Customers by Country): Click here to enter text.

FindItems (Country Code from Customers by Country): Click or tap here to enter text.

FirstItem (Country Code from first Contract: Customer): Click here to enter text.

*["First Item" Returns record set containing only the first item from the SourceRecordSet or empty record set if SourceRecordSet is empty. This function is very useful if metadata associated with a particular lookup record, but do not want to reproduce such associated date over and over.]*

Count GroupBy (number of groups): Click here to enter text.

## *[Note that you may keep embedding multiple group layers be nesting another "GroupBy".]*

## **2nd level group (Customer): 2nd Group Nested Category**

GroupTitle (Customer): Click here to enter text.

GroupItems (Contracts grouped by Country and Customer): Click here to enter text.

Count (Contracts grouped by Country and Customer): Click here to enter text.

GroupTitle & Count: Click here to enter text.

FindItems Count (Customer by name): Click here to enter text.

FindItems (Customer email from Customer by name): Click or tap here to enter text.

FirstItem (Customer email from first Contract: Customer): Click here to enter text.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Contract File Name | Customer:Title | Customer:FirstName | Customer:Country |
|  | Click here to enter text. | Click here to enter text. | Click here to enter text. | Click here to enter text. |

*[Note that the above table row uses the "GroupItems" command. Encasing a section (in this case a row) with this command and then including fields within this content control section allows you to pull list data and repeat it for each record in the dataset specified at the very top of the data pull, in this case "ReportItems".]*

# First Item (introduced in Version 3.7.16.06081)

"FirstItem" allows you to pull the first record from a query (e.g. "ReportItems", "GetView", "FindItems"). This function is useful when you want to produce related metadata from a lookup list item within a grouped view.

The following example is a two level grouped view. The "2nd Group Nested Category" groups items by LegalTag - which is a lookup column that relates to the LegalTags list. In the LegalTags list, each LegalTag is associated with a description. In the example below, the "Description" text would like to be included. However, it only needs to be reproduced once (and not over and over each time the same LegalTag is repeated. Thus, the FirstItem function allows for the Description field to be reproduced one time.

1st Group Category Title and Count

2nd Group Nested Category

2nd Group Nested Category

Description - Utilizes "FirstItem"

|  |  |
| --- | --- |
| Item | Description of Issue |
|  | ​Click or tap here to enter text.  Section: Click or tap here to enter text. |

**FirstItem() function:**

FirstItem(SourceQuery)

The "SourceQuery" may be ReportItems, GetView, or FindItems data pull function.

**Examples**:

**Examples - Single Value:**Note these functions combine the query and the column into a single function. Thus, in the first example below, the LegalTag:Description field (which is a lookup list field that is pulling the description column information) is being produced for the first item pulled for the "ReportItems" record set.

FirstItem(ReportItems):LegalTag:Description

FirstItem(GroupItems): LegalTag:Description

FirstItem(FindItems("Diligence Summaries";"Content Type";"Material Agreement")):Legal Title

FirstItem(GetView("Diligence Summaries";"Sorted and filtered view")):Legal Title

**Examples - Repeater/RecordSet:**Within these functions, you would include content controls such as "Title" or "Legal Title." These are record sets/pulls.

|  |  |
| --- | --- |
| ID | ID |
| Legal Title | Legal Title |
| Agreement Summary | Agreement Summary |

FirstItem(ReportItems)

FirstItem(GroupItems)

FirstItem(FindItems("Diligence Summaries";"Content Type";"Material Agreement"))

FirstItem(GetView("Diligence Summaries";"Sorted and filtered view"))

1. Remember that all queries must be inserted using a rich text content control. [↑](#footnote-ref-1)
2. Introduced in Version 3.7.16.06081 [↑](#footnote-ref-2)
3. Introduced in Version 3.7.16.06081 [↑](#footnote-ref-3)