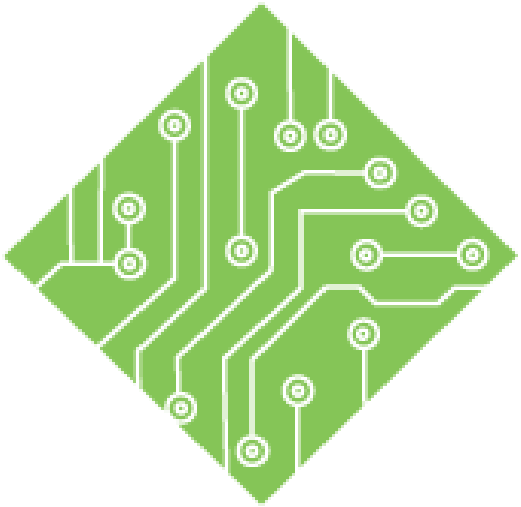
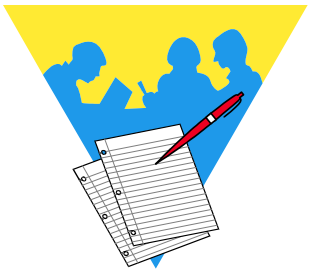


Lesson 3: Database Functions

Lesson Overview

You will cover the following concepts in this chapter:

- Database Functions
- Basic Syntax of D-Functions
- Creating a D-Function Formula
- Expanding D-Functions
- Adding Drop-Down Menu's



Lesson Notes

Database Functions

When there are large amounts of data within your datasets the D-Functions can make searching for specific information faster. While simple functions are used in many instances there will be times where calculations need to be made based on a set of criteria. The D-Functions allow you to query the dataset based on a defined set of specified criteria to control what is being calculated by modifying the formula's criteria. Using database functions is similar to advanced filtering; you must establish a criteria range before the function itself.

Note

When using the D-Function formulas you can change the criteria and the formula results will update immediately.

There are many D-Functions in *Excel* designed to help you extract subsets of data from within large datasets. The D-Functions include:

- ***DAVERAGE***: Calculates the average of values in a field of a list or database, that satisfy specified conditions
- ***DCOUNT***: Returns the number of cells containing numbers in a field of a list or database that satisfy specified conditions
- ***DCOUNTA***: Returns the number of non-blank cells in a field of a list or database, that satisfy specified conditions
- ***DGET***: Returns a single value from a field of a list or database, that satisfy specified conditions
- ***DMAX***: Returns the maximum value from a field of a list or database, that satisfy specified conditions
- ***DMIN***: Returns the minimum value from a field of a list or database, that satisfy specified conditions
- ***DPRODUCT***: Calculates the product of values in a field of a list or database, that satisfy specified conditions
- ***DSTDEV***: Calculates the standard deviation (based on a sample of a population) of values in a field of a list or database, that satisfy specified conditions
- ***DSTDEVP***: Calculates the standard deviation (based on an entire population) of values in a field of a list or database, that satisfy specified conditions
- ***DSUM***: Calculates the sum of values in a field of a list or database, that satisfy specified conditions.

Database Functions, continued

- *DVAR*: Calculates the variance (based on a sample of a population) of values in a field of a list or database, that satisfy specified conditions
- *DVARP*: Calculates the variance (based on an entire population) of values in a field of a list or database, that satisfy specified conditions

Basic Syntax of D-Functions

While each type of D-Function will return different values, they all share the same arguments. So the component which changes will be the function name itself as the argument structure remains consistent.

Here is the base syntax of the D-Function formula.

=Dfunction(Database,Field,Criteria)

The Arguments breakdown as follows:

- ***Database***: The range of cells containing the data being searched.
- ***Field***: This is the column being calculated on in the function.
  - If the column header has text then be sure that the text you enter here is wrapped within quotation marks.
  - Numeric values do not require quotations.
  - If using number to represent columns then enter a 1 for the first column, 2 for the second column, and so on.
- ***Criteria***: A cell range containing the conditions that must be met in order to be included in the final calculation.
  - Any range of cells can be the criteria argument, it must include at least one column label and a cell below the column label that defines the condition to be considered in the calculation.

Note: All of the database functions use the same argument format.

Some common and useful database functions are:

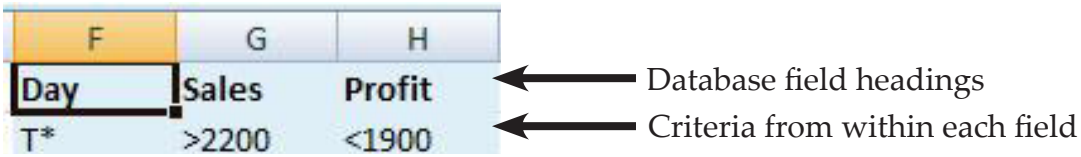
DAVERAGE	Used to average values in a field based on specified criteria
DSUM	Sums the values in a field that meet the entered criteria
DCOUNT	Counts the cells that contain numbers that meet the specified criteria
DMAX and DMIN	Return the largest and smallest values respectively from records that meet the specified conditions.
DPRODUCT	Multiplies values in a field according to specific conditions
DGET	Returns a single record value from a record that meets the specified conditions.

Creating a D-Function Formula

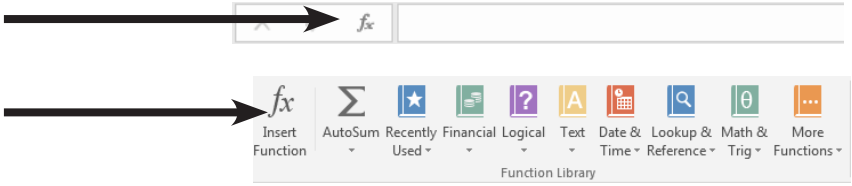
Standard database functions include those built-in functions that are in the **Database** category found in the function library of the *Insert Function* dialog.

Using the Insert Function Dialog

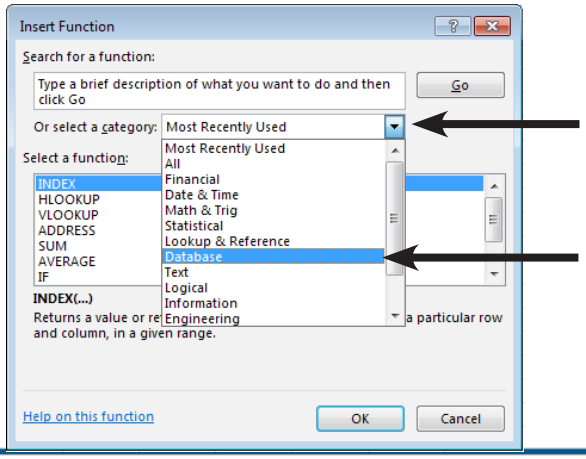
- Set up a criteria range outside of the database range of cells.



  - Use field headings from the database that you want the information to be filtered and calculated by.
  - The heading must match the headings in the database exactly. Consider copying and pasting these headings.
  - In the row under the field headings, type the criteria you want your calculation to be based on.
- Choose a cell to place the function in.
  - Click the **[Insert Function]** button on the **Formula Bar** or the **[Insert Function]** on the ***Formulas Tab***.



  - In the *Insert Function* dialog, choose **Database** from the *Or select a category:* drop-down list.
  - Select the database function you wish to use from the *Select a function:* field, then click the **[OK]** button.

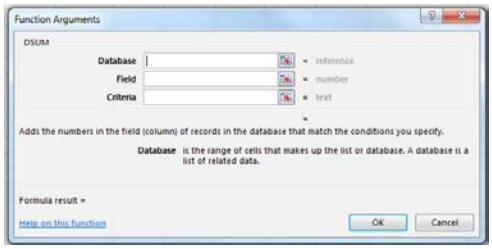


Note

You will not find the **Database** category in the **Function Library** on the ***Formulas Tab***. They are found in the **Database** category within the *Insert Function* window.

Creating a D-Function Formula, continued

- The *Function Arguments* dialog is displayed.



  - In the *Database* field;
    type your cell range, database name, or click into the text field and select the **Database** data set on the spreadsheet.
  - In the *Field* field;
    type the name of the column heading field that will be searched to extract your desired data, or click on the field name in the spreadsheet.
  - In the *Criteria* field;
    select the entire criteria range you made earlier.
  - Click the **[OK]** button to finish the formula.

Entering the Function Manually

Select the cell where the result is to be placed and entered the function directly into the cell or formula bar using the Autocomplete list. Adding a "D" to several standard functions will convert the function to a database function, which allows you to specify criteria to control and limit the results returned.

- Define the Criteria range of cells, as before.
- Select the result cell.
- Begin typing the function;
  - **=DSum(**
  - Define the Database range of cells or enter the Name if the range has been named.
  - **Comma**.
  - Enter the field heading, exactly as written in the database range. (If the Field name is text, it must be within quotation marks.)
  - **Comma**.
  - Enter the criteria range of cells that was defined at the beginning of this process.
  - Close the Parenthesis then tap the **Enter** key.

Note

When you are manually entering a formula, when the desired function is highlighted, use the **Tab** key to enter the function into the formula.

Note

While typing in a formula, watch the tooltip to help enter the formula. The segment of the tooltip that is bold represents the formula argument being entered.

Action 4.1 - Using a D-Sum function

Instructions:	Results/ Comments:
1. Open the **Class_List.xlsx** workbook.	The file is in the lesson folder.
2. Save the file as **My_Class_List**.	Save the file in the lessons folder.
3. Select cells **A1:N1**.	The selection covers all the column headers in the data set.
4. Copy the selected cells.	Right click and choose *Copy* or use the shortcut of **[Ctrl+C]**.
5. Select cell **P1** and paste the cells.	Right click and choose Paste or use the shortcut of **[Ctrl+V]**. You are beginning to establish the criteria range for the D-Function.
6. Select cell **P5** and type in; **< # of Students >**.	
7. Select cell **P6** and type in; **< # of Classes >.**	
8. Select cell **P7** and type in; **<Average # of Students >**.	
9. Select cell **P8** and type in; **< Total Earnings >**.	These cells represent the desired information to be extracted from the raw data set.
10. Select cell **R2**, type in; **< 2010 >.**	You will be looking for any 2010 entries in the Version column of data.
11. Select cell **S2**, type in; **< Level 2 >.**	You will be looking for any Level 2 entries in the Class column of data.
12. Select cell **Z2**, type in; **< >5 >.**	You will be looking for any number of students greater than five within the Number of Student column of data.
13. Select cell **Q5**.	This is where the DSUM formula will be entered

Instructions:	Results/ Comments:
14. Click the **[Insert function]** button on the Formula bar.	The *Insert Function* dialog opens.
15. In the *Insert Formula* dialog, select the following: **Category**: *Database* **Function**: *DSum* and click **[OK]**.	You have chosen the type of function to insert and the *Function Arguments* dialog opens.
16. In the *Function Arguments* dialog, input the following: **Database**: **A1:N267** **Field**: **Z1** **Set Criteria**: **P1:AC2** and click **[OK]** to apply the formula.	The **Database** field refers to the data set being queried. The **Field** field refers to which column of data will be summed. The **Criteria** field refers to the search parameters.
17. Cell **Q5** now lets you know how many students took a 2010 Level 2 class, but only if there more than five student in the class.	
18. Select cell **Z2** and delete the current contents.	This removes the specific parameter of more the five students in the class from the search criteria.
19. Notice that the value in cell **Q5** changes.	The value in the cell now show the full total of students who took 2010 Level 2 classes.
20. Save the file.	**[Ctrl+S]**.

Action 4.2 - Using a D-Count function

Instructions:	Results/ Comments:
1. The **My_Class_List.xlsx** file should still be open.	If not, re-open the file from the lessons folder.
2. Select cell **Q6**.	In this cell you want to know how many times the 2010 Level 2 class has be run.
3. Enter the following formula **< =DCOUNT(A1:N267,T1,P1:AC2) >.**	The formula has counted every instance of a 2010 Level 2 class within the dataset.
4. Select cell **T2**, type in; **< >6-30-2016 >.**	By adding a new criteria to the search parameters , the search results are narrowed to only classes run after June 30th 2016.
5. Notice the changes in cells **Q5** and **Q6**.	The results in both cells are updated to reflect the additional search parameter added to the criteria range.
6. Select cell **T2**, type in; **< <6-30-2016 >.**	The formulas now return only 2010 Level 2 classes run before June 30th 2016.
7. Clear the contents in cell **T2**.	The search no longer is limited by any date constraints.
8. Save the file.	**[Ctrl+S]**.

Action 4.3 - Using a D-Average function

Instructions:	Results/ Comments:
1. The **My_Class_List.xlsx** file should still be open.	If not, re-open the file from the lessons folder.
2. Select cell **Q7**.	In this cell you want to know the average number of students who attended 2010 level 2 classes.
3. Enter the following formula **< =DAVERAGE(A1:N267,Z1,P1:AC2) >.**	The formula returns the average number of students that attended 2010 Level 2 classes. Although, it is showing the results with decimals.
4. Double click into cell Q7.	By double clicking a cell, you editing the cell contents and in this case able to modify the formula.
5. Edit the formula to this : **< =ROUND(DAVERAGE(A1:N267,Z1,P1:AC2),0) >.**	By wrapping the D-Average formula inside a ROUND function and specifying the number of decimals as 0, forces the results of the original formula to be rounded to a whole value. You could have also decreased the number of decimals by click the [Decrease Decimals] button in the **Number Group** on the ***Home Tab***.
6. Add another DSUM formula in cell **Q8** to calculate the earnings from the 2010 Level 2 classes.	The formula should be **=DSUM(A1:N267,AC1,P1:AC2).**
7. Right click cell **Q8** and click the **[$]** button in the Mini Toolbar.	The cell now has the Accounting formatting applied.
8. Save the file.	**[Ctrl+S]**.

Expanding D-Functions

So far, you have successfully been using the D-functions to find information in the dataset by searching for a single item within each column of the dataset. When you need to find data based on more than one data point within a column, it becomes necessary to add more rows within the criteria range in the formula. Each additional row allows you to expand the search parameters used in the formula.

When adding search criteria along a single row in the formula, you have been looking for all the criteria to be met in order to get a result. As you add new search criteria in the same column of the criteria range, the formula searches for all matching data points within the column.

Below are some examples of how the criteria can be arranged to search for specific information.

Version	Class
2016	level 2

Looking only for version 2016 Level 2 classes

Version	Class
2016	
2013	

Looking only for versions 2013 and 2016, any classes

Version	Class
2016	
	level 2

Looking for all version 2016 and all Level 2 classes

Version	Class
2016	
2013	level 2

Looking for all version 2016 and only version 2013 Level 2 classes

Action 4.5 - Expanding the Criteria Range

Instructions:	Results/ Comments:
1. The **My_Class_List.xlsx** file should still be open.	If not, then re-open the file.
2. Change the criteria ranges from **P1:AC2** to **P1:AC3** in cells **Q5:Q8**.	You are adding another row each the criteria range to allow for multiple parameters to be searched for within the formulas.
3. Select cell R3, type in; **< 2013 >**.	You are now searching for all records of 2010 Level 2 and any 2013 version classes.
4. Notice that all the values in cells **Q5:Q8** have been updated.	The values now reflect the expanded search criteria .
5. Select cell **R3** and press the **[Delete]** key.	The cell contents are removed.
6. Select cell **T2**, type in; **< >6-30-2016 >.**	You are again search for classes run after June 30th 2016.
7. Select cell **T3**, type in; **< <10-1-2016 >.**	You are now also limiting the search to classes run before October 1st 2016. In this manner you are able to search for classes run within a specific time period.
8. Notice the values in cells **Q5:Q8** are updated.	
9. Save the file.	**[Ctrl+S]**.

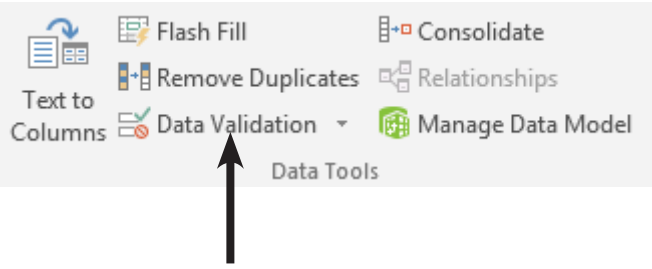
Adding Drop-Down Menu's

To continue the simplification of data retrieval from large datasets using the D-FunctionS, you will now use the Data Validation tool to add drop-down menus within the criteria range.

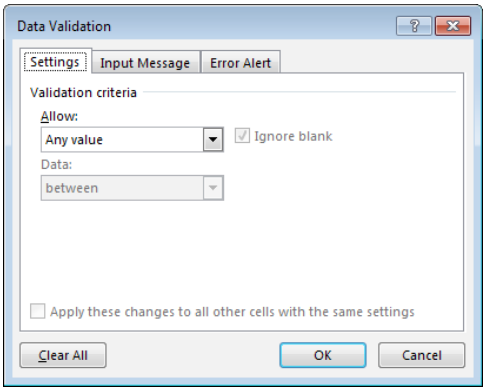
The first set in this process will be to extract unique values from columns within the dataset using the Advanced Filter tool used earlier. Once you have extracted the unique values from each column you can move on to the next step in the process.

Data Validation Lists

To create the drop-down menu use a Data Validation List. Select the cell where to drop-down list needs to be placed. To access the Data Validation tool; go to the ***Data Tab*** and click the **[Data Validation]** button in the **Data Tools Group**.
The *Data Validation* dialog opens



- In the **Allow** field, choose *List* from the available options.



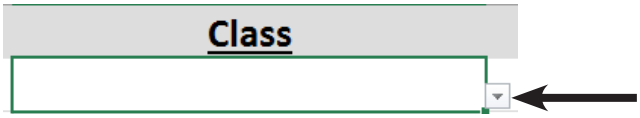
Adding Drop-Down Menu's, continued

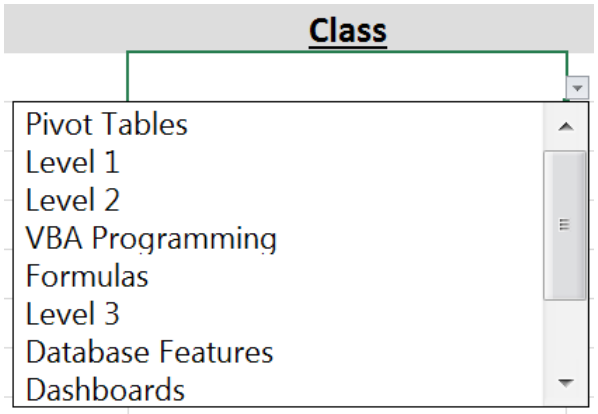
- Click into the **Source** field,
  - You can type your list in manually here. If you are typing the list in yourself, use a comma to separate each list entry.

-OR-

  - Refer to a cell range that contains the list entries by highlighting the desired cells.
- You could add Input and Error Messages by clicking the appropriate tab in the dialog.
- Click the **[OK]** button.

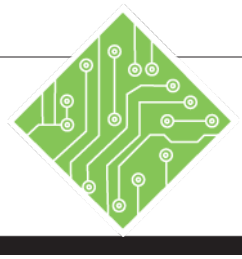
The active cell now has a drop-down arrow when selected. Click the arrow allows users to choose any item from the list you created.





Action 4.6 - Extracting Lists from the Dataset

Instructions:	Results/ Comments:
1. The **My ClassList.xlsx** file should still be open.	If not, then re-open the file.
2. Select cells **C1** and **D1**.	These are two of the columns that you will extract unique values from.
3. Copy the cells and paste them into cells **P10** and **Q10**.	This is where the filtered data will be placed.
4. Select cell **I1**	This is another of the columns that you will extract unique values from.
5. Copy the cells and paste them into cell **R10**.	This is where the filtered data will be placed.
6. Select cell **L1**	This is another of the columns that you will extract unique values from.
7. Copy the cells and paste them into cell **S10**.	This is where the filtered data will be placed.
8. On the ***Data Tab,*** in the **Sort & Filter Group,** click the **[Advanced]** button.	The *Advanced Filter* dialog opens.
9. Click the ***Copy to another location*** radio button.	The returned values will now be copied to another location in the spreadsheet.
10. Click into the **List Range:** field and type in; **< A1:N267 >**.	This is the dataset that contains the source data.
11. Click into the **Criteria range:** field and type in: **< P10 >.**	This is the field within the data set to be searched.
12. Click into the **Copy to:** field and type in; **< P10:P11 >**.	This is where the data will be placed, if you don't include the column header the filter will return all columns from the dataset.
13. Check the ***Unique records only*** checkbox.	As implied by the name, only unique values will be returned by the filter.



Instructions:	Results/ Comments:
14. Click the **[OK]** button.	The dialog closes and the warning dialog appears. This dialog asks if you want to extend the copy location range to include all the found records.
15. Click the **[Yes]** button.	The data is placed.
16. Repeat steps 8 through 15 for each of the remaining columns of required data.	Make the necessary adjustments to the filtering for each column of data.

Action 4.7 - Using Data Validation to Create Drop-downs

Instructions:	Results/ Comments:
1. The **My_Class_List.xlsx** file should still be open.	If not,then re-open it.
2. Select cell **R2**.	This is the first cell where you will place the drop-down menu for users to choose items.
3. Click the ***Data Tab***, then click the **[Data Validation]** Button in the **Data Tools Group**.	The *Data Validation* dialog opens.
4. On the ***Setting Tab*** in the *Data Validation* dialog, choose *List* from the **Allow:** field drop-down.	This is the type of input setting the ***Data Validation*** tool inserts.
5. Click into the **Source**: field.	This is where you can define the list manually or enter a cell range containing the values to be used as the list.
6. Highlight cells **P11:P14**.	These cells contains the list entries.
7. Click the **[OK]** button.	The *Data Validation* dialog closes and a drop-down arrow in displayed in cell **R2**.
8. Use the Autofill handle to pull down to cell R3.	Cell **R3** now is also setup.
9. Repeat steps 2 through 8 to add drop-down list for cells **S2:S3**, **X2:X3**, and **AA2:AA3** respectively.	Each section of the criteria range used in the D-FunctionS formulas now has drop-down to make user input flawless. Make sure you refer to the correct source rang of cell in each ***Data Validation***.
10. Clear any values currently within the Criteria range.	The D_FUNCTION formulas should all read 0.
11. Try using the new drop-downs to modify the data being returned by the D-Function formulas.	Depending on the choices made using the drop-downs, the formula values change.
12. Save and close the file.	**[Ctrl+S]** and **[Ctrl+W]**.