Join Types in a Query

In order for accurate and relevant data to be extracted to a dynaset (the query results), there must be valid joins between multiple tables (see our Access 2013 - Introduction to Queries quick reference guide).

The join type is also important. The default type is an inner join, where records are only included in the dynaset if there is matching data in the join fields of both tables. You can, however, create an outer join, where all the records from the one table appear, even if there is no matching data in the other table.

To specify the join type, you need to open the Join Properties dialog box.

There are three types of join available:

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Only include rows where the joined fields from both tables are equal.</td>
<td>This is the default inner join.</td>
</tr>
<tr>
<td>2</td>
<td>Include ALL rows from &lt;&lt;first table&gt;&gt; (the primary table) and only those records from &lt;&lt;second table&gt;&gt; (the foreign table) where the joined fields are equal.</td>
<td>This is a left outer join. Assume that &lt;&lt;first table&gt;&gt; is a table of customers and &lt;&lt;second table&gt;&gt; a table of orders. You have added to the query grid the Customer Name field from the customer table and the Order ID field from the Orders table. The resulting dynaset would show ALL the customers’ names but in cases where a customer has not got any related records in the Orders table, it will leave a blank. This could be useful to identify customers in your database who haven’t placed any orders for your products.</td>
</tr>
<tr>
<td>3</td>
<td>Include ALL rows from &lt;&lt;second table&gt;&gt; (the foreign table) and only those records from &lt;&lt;first table&gt;&gt; (the primary table) where the joined fields are equal.</td>
<td>This is a right outer join. Using the same tables and fields as in type 2 above, the resulting dynaset would show ALL the Order IDs but in cases where an order ID has not got a related customer name, it will leave a blank. This could be useful to identify orders which do not have a recognisable customer due to input error or because the customer records have been inadvertently deleted or lost.</td>
</tr>
</tbody>
</table>

In the outer join extract below, it is possible to see that B&B Sporting Goods, Champs and The Complete Athlete have no order numbers. It is safe to assume, therefore, that they have not purchased any products. This would have not been possible to see with an inner join because the three shops names would not have appeared in the query.

1. Create a new query or open an existing query in Design view.
2. Create joins between tables as required (see our Access 2013 - Introduction to Queries quick reference guide).
3. Double-click the middle segment of the join line.

4. Select the required join type.

5. Click OK. Direction of arrow indicates that ALL records will be shown from the Customers table.

6. Run the Query.

To return in the dynaset just the records from one table that do not have a match in the other:

1. In Query Design View, add Is Null criteria to a field from the table where there is no match (the table where the arrow head points).
Using the Crosstab Query Wizard

Crosstab queries are used to group and summarise information into a datasheet format. Crosstab queries make it easier to totals for common data.

A crosstab query can display, for example, the number of products sold (row headings) by sales representative (column headings). The number of products sold by each sales representative would appear at the intersection of the corresponding row and column. Crosstab queries can also involve other functions such as sum, average, max and min.

Values in a crosstab query cannot be changed.

1. Select the Create tab.
2. In the Queries group, click Query Wizard.
3. In the New Query dialog box, select the Crosstab Query Wizard option.
4. Click OK.
5. From the list, select the table/query that contains the records you want to retrieve.
6. Click the Next button.
7. From the Available Fields: list, add the field you want to use as row headings to the Selected Fields: list.
8. Click the Next button.
9. Select the field you want for column headings.
10. Click the Next button.
11. In the Fields: list, click on the field you want to calculate.
12. In the Functions: list, click the required function.
13. Click the Next button.
14. Enter a name for the query and click Finish.

Using the Find Duplicates Query Wizard

You can use a Find Duplicates Query Wizard to find duplicate records in a table. For instance, there may be identical records stored in a table, when only one record is necessary. The Find Duplicates Query Wizard will locate and display records in which the specified field has the same values.

1. Select the Create tab.
2. In the Other group, click Query Wizard.
3. In the New Query dialog box, click the Find Duplicates Query Wizard item.
4. Click the OK button.
5. In the first dialog box, select the table/query whose records you want to display.
6. Click the Next button.
7. Select the table which contains the related records.
8. Click the Next button.
9. If necessary, specify the joining field.
   a) Select the common field from each list.
   b) Click the  button to join them.
10. Click the Next button.
11. Choose which fields from the Available fields: list to display in the query.
12. Click the Next button.
13. Enter a name for the query.
14. Click the Finish button.

Using the Unmatched Query Wizard

The Find Unmatched Query Wizard allows you to locate and display records in one table for which there is no match in a related table. For example, locating from a customers table, those who have not placed any orders. The unmatched Query Wizard helps you create an Outer Join query (see page 1 of this document).

1. Select the Create tab.
2. In the Other group, click Query Wizard.
3. In the New Query dialog box, click the Find Unmatched Query Wizard item.
4. Click the OK button.
5. In the first dialog box, select the table/query whose records you want to display.
6. Click the Next button.
7. Select the table which contains the related records.
8. Click the Next button.
9. If necessary, specify the joining field.
   a) Select the common field from each list.
   b) Click the  button to join them.
10. Click the Next button.
11. Choose which fields from the Available fields: list to display in the query.
12. Click the Next button.
13. Enter a name for the query.
14. Click the Finish button.
Creating a Calculated Field

Access allows you to create ‘expressions’ that calculate new field values; for example, you can create an expression that multiplies the value in the Quantity field by the value in the Price field to calculate a sale value.

In expressions, field names are enclosed in square brackets ([ ]); numbers are not. For example, to calculate 20% of sales and display the results in a column named Commission, you would enter Commission:0.2*[Sales] in the design grid. (The colon separates the column name from the expression.)

Fields used in an expression must be available in the tables that underlie the query although they do not have to be displayed in the query results.

1. Open the required query in Design view.
2. Click in a blank Field row.
3. Type a name for the calculated field followed by a colon, eg. ItemPrice:
4. Type the expression required for the calculation. Field names must be in square brackets, eg. [Quantity] * [UnitPrice]
5. Press the [Enter] key.
6. View or Run the query.

TIP: You can add criteria to a calculated query to remove nonessential records.

TIP: To help you see your calculated field expression more easily while you are creating it, you can open the Zoom window by holding down the [Shift] key and pressing F2.

Formatting a Calculated Field

Once you have created a calculated field, you can change its properties as you would any other field on the design grid. The Format property determines how data appears in the query results (Datasheet view). For example, you can change the properties of a calculated field so that the field values display as currency.

1. Open the required query in Design view.
2. Click in the field to be formatted.
3. In the Show/Hide group, click Property Sheet.
4. In the Property Sheet, click the Format down arrow.

Creating a Function Query

Access allows you to create a query that groups records by a selected field and then applies a function that calculates a value within the grouped fields.

<table>
<thead>
<tr>
<th>LName</th>
<th>FName</th>
<th>SumOfNum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adams</td>
<td>George E.</td>
<td>9</td>
</tr>
<tr>
<td>Adams</td>
<td>Stephanie L.</td>
<td>3</td>
</tr>
<tr>
<td>Brown</td>
<td>Nathan T.</td>
<td>5</td>
</tr>
<tr>
<td>Lacey</td>
<td>Angela H.</td>
<td>5</td>
</tr>
<tr>
<td>Norris</td>
<td>Henry D.</td>
<td>4</td>
</tr>
<tr>
<td>Stevenson</td>
<td>Thomas A.</td>
<td>4</td>
</tr>
</tbody>
</table>

A grouped and totalled report

Access allows you to perform more than one calculation on a field. For example, you can group records by weekly sales and then find both the minimum and maximum values.

There are several types of functions from which you can choose. The most commonly used functions are listed in the following table:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>Totals the values in the calculated field</td>
</tr>
<tr>
<td>Average</td>
<td>Averages the values in the calculated field</td>
</tr>
<tr>
<td>Count</td>
<td>Counts the number of records in the calculated field</td>
</tr>
<tr>
<td>Max</td>
<td>Finds the highest value in the calculated field</td>
</tr>
<tr>
<td>Min</td>
<td>Finds the lowest value in the calculated field</td>
</tr>
</tbody>
</table>

1. Open the required query in Datasheet view.
2. Select, if necessary, the Design tab under Query Tools.
3. In the Show/Hide group, click the Totals button. An extra Total: row appears near the top of the query grid.
4. Add to the first column at the left of the query grid, the first field that you want to group on (eg. LastName).
5. Leave the Total: row set to Group By.
6. If required, add to the second column from the left of the query grid the second field that you want to group on, eg. FirstName.

7. Leave the Total: row set to Group By.

8. If necessary, continue adding more fields to group on in sequence at the left of the query grid. In practice, you will only be grouping on one or two fields.

9. Add in the next column to the right of the Group By fields, the field you want to create totals from. In most cases this will be a number or currency data type unless you just want to count when it can be a text data type.

10. Click the down arrow in the Total: row.

11. Select a function to aggregate by.

12. Run the query.

---

**Set Top Values in a Query**

You can limit the results of a query so that only the highest or lowest values in a field appear in the query results. You can also limit the number of records to a specific number or percentage of all records being queried.

The field for which you are setting the top or bottom values must be sorted. If the field is sorted in descending order (Z to A, 0 to 9), the top values will be found. If the field is sorted in ascending order (A to Z, 0 to 9), the bottom values will be found.

1. Open the required query in Design view.
2. Select the Sort row of the required field.
3. Select the order you require (Ascending or Descending).
4. In the Query Setup group, click the Return list arrow.

---

**Creating A Parameter Query**

A parameter query is designed to prompt the user for input each time it is run; Access then runs the query based on the criterion (parameter) entered.

You can add multiple parameters to a query. When you run the query, a prompt will appear for each parameter in it.

1. Open the required query in Design view.
2. Click in the Criteria row of the desired field.
3. Type the prompt surrounded by square brackets (eg. Enter the region you want to see data for):
4. Press [Enter]