Advanced Relational Databases

Microsoft Access

MS Access is a powerful personal database management system that doubles as a development platform. You can create comprehensive database applications including a user interface and complex querying and reporting functionality. The underlying Visual Basic for Application programming language provides experienced and novice developers alike a flexible environment to link applications to more complex DBMS' such as Microsoft SQL server (not covered in this tutorial).

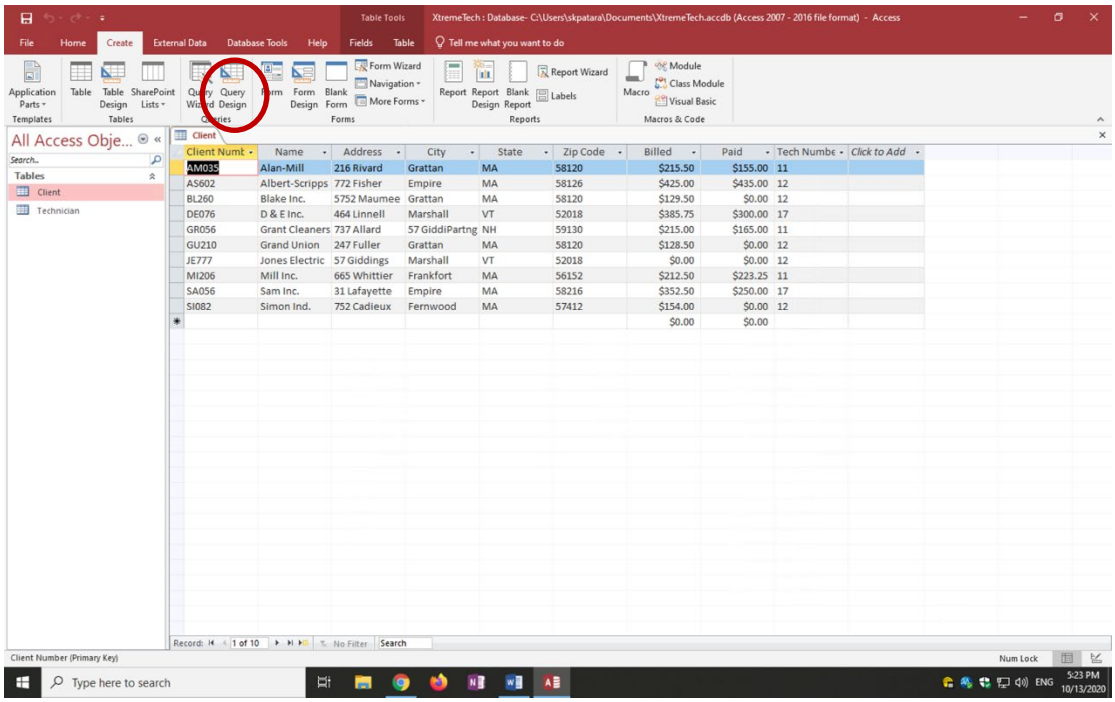
In this session you will create simple and aggregate queries using Microsoft's query builder. You will be introduced to Form and Report wizards which speeds up the creation of simple forms and reports.

You will need the database you created in the previous lab (available on Canvas).

Querying Tables

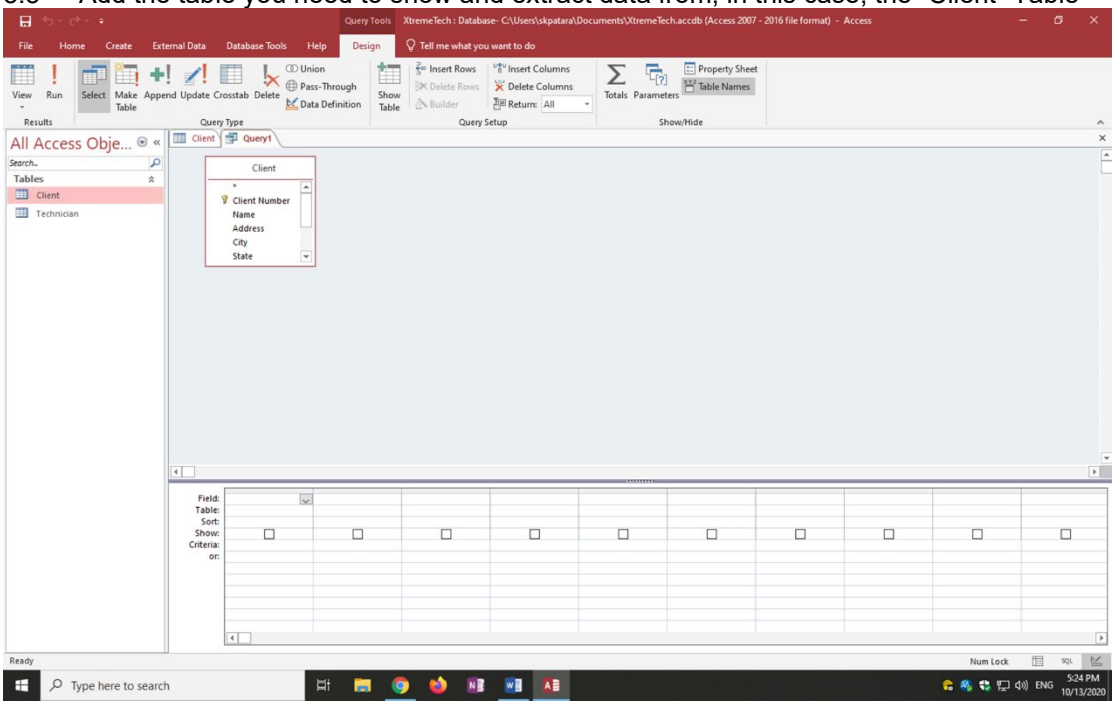
Queries are those objects in a database that allow us to view, change, and arrange data stored in tables. A lot of the time, we regard them as the source of records for forms and reports and to find the data that we want. There are various types of queries but in this tutorial, we will focus on the most common queries of all – the "Select queries". These are the queries that extract or show us data. They will show data that complies with specific criteria. A select query generates a logical table (named this way because it is not actually in the hard drive but in the memory of the computer, and every time we open it is recalculated).

1 Open the database you have created in the previous lab
2 Click to open the "Client" Table
3 Create a simple query using the Design View
  3.1 Click the Queries Design button under the "Create" menu



3.2 You will see a split window (see next screen capture): the top being the table area and the bottom part is the grid area

3.3 Add the table you need to show and extract data from, in this case, the “Client” Table



Every row has a function:

Field: To place the fields to use which will usually be the ones we want to see, it could be a field from the table or a calculated field.

Table: To specify the name of the table we want to extract the field from. This will be useful when we are defining queries based on more than one table.

Sort: To select the sorting function to arrange the resulting rows.

Show: If this checkbox is not marked, the column will not appear in the result. Leave this option unchecked when we want to use the field to define the query but do not want the field to appear in the result. E.g., we need the City field to extract all the clients from "Grattan", but we do not want the city to appear in the result as we already know that they are all from the same city.

Criteria: To specify the lookup criteria, a condition that records need to comply with to appear in the result of the query. It can be made up from one or from various conditions, joined by AND and OR operators.

or: To specify multiple conditions using this and the following rows.

3.4 In this case, select Client Number, Client Name, City, Billed amount and Paid amount. You can do this by drag and drop from the table above or select them using the pull-down menu.

3.5 Add a criteria, for example: add "Grattan" under "City"
You MUST add the field to the design grid first before entering the criterion. You can use:

- Use text data in a criterion such as "Grattan"
- Use a wildcard such as "G*", "*g", "*g*"
- Use a number in a criterion
- Use a comparison operator such as >300

3.6 Click "Run" (the red "!" on the tool bar)

3.7 Results of your queries will be shown

Create a parameter query

It is very useful to be able to find the data that you need by entering a criterion in the grid table. However, it is likely that your criterion will change depending on situations, such as instead of looking for clients from Grattan, you want to check the records of clients from Marshall. It will be rather tedious to create queries for every single city. Instead, we could create a parameter query so that we can enter different criterion depending on what information we need. In order to do so we replace the string of text with a variable in square bracket, e.g. [X].

When Access encounters a variable (i.e., something that is not a literal string, such as "X") during execution, it attempts to bind the variable to some value. If you enter something that it cannot recognize, as a last resort, Access will ask the user for the value of the parameter via the "Enter Parameter Value" dialog box.

3.8 If you enter "[X]" as a criterion under "City" and run it

3.9 A pop-up box will appear and ask you to enter the criterion

3.10 However, the box will not tell you what to enter, therefore it will be more user-friendly if you can enter something meaningful, such as "[Enter the city please]", then the phase will appear on the pop-up window. The phase within the [ ] is the parameter name.
NOTE: with parameter query, you need to enter the EXACT name of the city. Cannot use wild card

3.11 Give the query a name and Save it

Creating calculated fields in Queries

A calculated field is an output field that is made from other field values. A calculated field is not a field in a table; it is created in the query generator.

For example, if we want to know how much each of the client owes the company, we need to calculate the outstanding amount, which will be Paid amount minus Billed amount.

3.12 You can continue and add this field into the last query or start a new one (which you will have to select clients name etc. again)

3.13 Type the expression "Outstanding amt: [Billed] – [Paid]" onto an empty field
The phrase "outstanding amt" gave the column its title, i.e. the heading. The field names "Billed and "Paid" must be enclosed square brackets

NOTE: as a RULE field names MUST ALWAYS be enclosed in square brackets!!!

3.14 If the calculated field output is not shown in Currency format. To convert the output to currency format, click the line above the calculated field expression, thus activating the column, then select View -. Properties. A "Field property" window will pop up. Click Currency, and then close the pop-up window.

3.15 Run the query. You can save the query in another name. Then back to Design View.

Performing a summary queries

In Access we can define a special query to calculate totals on the records of a table (or various tables related). In order to obtain those totals we use summary functions, so it is why we call them summary queries. It is important to remember that the resulting rows of a summary query have a distinct nature to the rest of the rows resulting from queries as each row corresponds to various rows in the source table. A 'non summary' query row corresponds to a row in the source table and contains data found in just one row of the source, whilst a row from a summary query corresponds to the summary of various rows from the source table, this difference does not permit us to edit the data.

This function is very useful when we want to know, for example, the total amount due from a single location or salesperson. Instead of listing all the billed amounts from a location and adding them manually, we can ask Access to add them up for us. It is particularly useful for orders with multiple items.

3.16 Open a new query and select the client table

3.17 Select the City and the Billed amount

3.18 Select the Design menu and click Totals to display the Total row (or click the Totals button on the toolbar)

3.19 Click the Total row under the City field, then click the drop-down arrow to display the summary functions. Click Group By.

3.20 Click the Total row under the Billed field and select Sum.

3.21 Run the query. Access will show you two columns, one with the city name and one with the TOTAL amount.
Apart from adding things up, this function also allows you to:

Count: a count of the number of instances in a field, i.e. the number of records

Average: the average of some field's values

Min: the minimum of some field's values

Var: the variance of some field's values

StDev: the standard deviation of some field's values

4 You can perform queries on multiple tables, but you MUST first specify their relationships (see previous exercises) before adding the tables into a query. Alternatively, you can perform a join operation in the Query window. You can see a line between matching fields.
   4.1 You can try using the Tech Number from the Tech table instead of from the Client table
5 You can run a saved Query whenever you need it.

Exercise

As a manager, you would like to know the **total** outstanding amount from clients under each technician. Design a query so that all **outstanding** amounts from all clients will be sorted according to technician's last name and added up together. Then you need to create a well-formatted REPORT for this query.

Integrating Excel Worksheet Data into Access Database

Some common reasons for using a database instead of a spreadsheet are:

- The worksheet contains a great deal of redundant data.
- Excel has a limit of 1,048,576 rows. No such limit in Access.
- The data to be maintained consists of multiple interrelated items.
- More powerful query and reporting capabilities.

Before converting the data, you need to create the database that will contain the data. To convert the data, you will use the Import Spreadsheet Wizard. In the process, you will indicate the first row contains the column headings. These column headings then will become the field names in the Access table. You should be aware that some of the steps might take a significant amount of time for Access to execute.

5.1 Select "External Data" Menu. Click the File that you would like to import. (you may need to select the correct file source first).
5.2 Follow the instructions from the Import Wizard.
5.3 Choose to store the data in a New Table
5.4 Choose your own Primary Key
5.5 It will ask if you want to index your field.

An index improves efficiency for sorting and finding records. On the other hand, indexes occupy space on your disk. They also require Access to do extra work. Access must maintain all the indexes that have been created up to date. So there are disadvantages as well as advantages when it comes to using indexes. As a guideline, you may want to create an index if:

- The field is a primary key of the table (but Access will do this for you automatically)
- The field is a foreign key in a relationship you have created (again Access will do this for you automatically)
- You frequently need to sort your data in this field
- You frequently need to locate records based on values in this field

5.6 Give the new table a name
5.7 After importing the data, then you will need to create a relationship of this new table with the existing table(s) before performing operations on them.

Exercise

Create the following table in Excel Table and import it into the Xtreme Technical Service database. Then create a relationship with the corresponding table. You need to decide which one is the primary key.

Part Number	Part Description	Tech Number
A0234	Bolt	11
B5675	Nail	12
C8845	Screw	17

You will encounter an error. Which parameters do you need to reset to resolve this problem?