

# Databases: Queries and data analysis





## About These Activities

### Software Used

*Access 2013*

*Windows*

### Files Used

Inventory.accdb

New Dentists.accdb

### Finding the Exercise Files

All the files for these exercises have been provided for you on a network drive. Your area of the drive is called the **Home Drive H:** .

They can also be downloaded from the ITLC Portfolio (visit <http://portfolio.it.ox.ac.uk> and search for “databases queries” or similar search text to find the course pack for this course).

### Trusted Locations

In the IT teaching rooms, the Home Drive has been set up as an Access Trusted Location. To find out more about Access Trusted Locations, read the article in the ITLC Portfolio (visit <http://portfolio.it.ox.ac.uk> and search for “Access trusted locations” or similar search text).

### Web App or Desktop Database?

*Access 2013* can also be used to create an **app database**: where users work on the data via a web browser. This would require communication using *Office 365* or *SharePoint 2013* (not currently available at Oxford University), and is not the subject of this course. We will work on a **desktop database**, which is saved locally on your computer or a network drive.


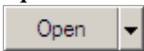
### Copyright



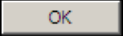
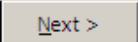
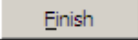








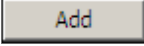
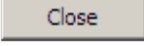
Pamela Stanworth makes this document and the accompanying slides available under a Creative Commons licence: Attribution-NonCommercial-ShareAlike CC BY-NC-SA. [pamela.stanworth@it.ox.ac.uk](mailto:pamela.stanworth@it.ox.ac.uk)



The Oxford University logo and crest and the IT Services logo are copyright of Oxford University and may only be used by Oxford University members in accordance with the University’s branding guidelines.


## Steps for the Learning Objectives

<b>Learning Objective One: Creating a query</b>	
<p>In the Inventory database, a small enterprise is cataloguing the equipment that it has purchased</p>	
<p><b>Task 1</b> Start <i>Access</i></p>	<p><b>Step 1</b> Start the computer if necessary</p>
	<p><b>Step 2</b> Click the <i>Windows</i> button on the taskbar at the bottom of the screen  In the <b>Start</b> menu, type <i>Access</i> in the search box</p> <p>Choose <i>Access</i> </p>
	<p><b>Step 3</b> If you are prompted for any user information, just click <b>OK</b></p>
	<p><b>Step 4</b> (On your office or home computer you might otherwise start the program using an <i>Access</i> icon on the Desktop)</p>
<p><b>Task 2</b> Open <i>Inventory.accdb</i>  It has been placed in your Home Drive H:\ (or in another place as directed by your teacher)</p>	<p><b>Step 1</b> Choose <b>File   Open</b></p>
	<p><b>Step 2</b> Browse to the Home Drive H:\  Alternatively, browse to a drive and folder as directed by your teacher  All the files you need for this course will be found here</p>
	<p><b>Step 3</b> Locate the file called <b>Inventory.accdb</b>  Open the file by selecting the filename then clicking</p> <p></p>
	<p><b>Step 4</b> If a security warning bar appears, stating that certain content in the database has been disabled, then read the Appendix</p>
<p><b>Task 3</b> Look at the list of tables and queries in the Navigation Pane</p>	

<p><b>Task 4</b></p> <p>Look at the relationships diagram to see the tables and joins in this database –</p> <p>click  on the <b>Database Tools</b> tab of the ribbon</p> <p>In this database, an administrator is looking after a collection of equipment, each item being assigned to a department</p> <p>When ready, close the Relationships diagram</p>	
<p><b>Task 5</b></p> <p>Use the wizard to create a query based on <b>tblAssets</b>, including these fields: <b>Asset Description, Make, Model and Date Acquired</b></p>	<p><b>Step 1</b></p> <p>Click  on the <b>Create</b> tab of the ribbon</p> <p>Select <b>Simple Query Wizard</b> and click </p> <p><b>Step 1</b></p> <p>In the list of available tables, choose <b>tblAssets</b></p> <p><b>Step 2</b></p> <p>Double-click on the fields called <b>AssetDescription, Make, Model and DateAcquired</b> to place these fields in the right-hand list</p> <p><b>Step 3</b></p> <p>Click  to move through the wizard</p>
<p><b>Task 6</b></p> <p>Save the query as <b>qryAssets</b></p> <p>Examine the results of <b>qryAssets</b></p>	<p><b>Step 1</b></p> <p>When asked by the wizard, give a suitable name for the new query: <b>qryAssets</b></p> <p>Make sure <b>Open the query to view information</b> is selected</p> <p><b>Step 2</b></p> <p>Click </p> <p><b>Step 3</b></p> <p>The query runs</p> <p>The results are presented in Datasheet View</p> <p>Notice that all records are included, but only those fields you requested</p>
<p><b>Task 7</b></p> <p>Switch between Design View and Datasheet View</p> <p>Close the query</p>	<p><b>Step 1</b></p> <p>Use  on the <b>Home</b> tab to switch to Design View</p> <p><b>Step 2</b></p> <p>In Design View, notice the table of assets with a list of all the available fields, in the top pane</p> <p>In the design grid (lower part), notice the fields that have been chosen and set out in columns</p>

	<p><b>Step 3</b></p>  <p>Use  on the <b>Home</b> tab to switch to Datasheet View</p> <p><b>Step 4</b></p> <p>When ready, close the query using </p> <p>Leave the file open, showing just the Navigation Pane</p>
<p><b>Task 8</b></p> <p>Start a new query in Design View</p> <p>Add two tables <b>tblAssets</b> and <b>tblDepartments</b></p> <p>These tables are already related by a join using <b>DeptID</b></p>	<p><b>Step 1</b></p>  <p>Click  to create a new query in Design View</p> <p><b>Step 2</b></p> <p>In the Show Table dialog choose <b>tblAssets</b> and <b>tblDepartments</b> by selecting the table name and clicking  or double-clicking a table name</p> <p>Repeat until you have both the required tables in the design grid behind</p> <p>(Do not worry if you collect some extras, they can be deleted later)</p> <p><b>Step 3</b></p> <p>Click  to use the design grid</p> <p>If an unwanted table has appeared, click it once to select it then press DELETE to delete</p> <p><b>Step 4</b></p> <p>Notice that the two tables are shown in the top pane, with a line joining the <b>DeptID</b> fields</p> <p>The top pane can be resized by dragging the divider between the two panes</p>
<p><b>Task 9</b></p> <p>Choose fields and set them up in the design grid:</p> <p>Asset ID</p> <p>Description</p> <p>Make</p> <p>Date for next maintenance</p> <p>Phone number for contacting the Dept</p>	<p><b>Step 1</b></p> <p>Set up a field in the first column of the design grid:</p> <p>Select the field name <b>AssetID</b> in the list for <b>tblAssets</b></p> <p>Drag the field name to the top of the first empty column in the grid</p> <p><b>Step 2</b></p> <p>The next fields you need are <b>AssetDescription</b>, <b>Make</b> and <b>NextMaintDate</b> from the <b>tblAssets</b></p> <p>Set these up in the design grid, perhaps by double-clicking a fieldname, or by clicking at the top of an empty column and picking a fieldname from the menu that drops down</p> <p><b>Step 3</b></p> <p>The final field, <b>ContactTelNum</b>, must be taken from the related table <b>tblDepartments</b></p>

<p><b>Task 10</b> Run the query Examine the results</p>	<p><b>Step 1</b></p> <p>Click  to switch to Datasheet View</p> <p>The query is run (at that moment <i>Access</i> interrogates the data in the tables)</p> <hr/> <p><b>Step 2</b></p> <p>Any records which appear in both tables with corresponding <b>DeptID</b> values are presented</p> <p>Only those fields you selected in the query design are shown</p> <hr/> <p><b>Step 3</b></p> <p>The data <i>can</i> be edited in a query</p> <p>For example, suppose that asset number 15, a Watkins bar code reader, needs its next service on 1 January 2018</p> <p>Locate that record, move into the <b>NextMaintDate</b> field, and type the required date to change the date in this list</p> <p>You have changed the value in the underlying table, <b>tblAssets</b> (check this if you're not sure)</p>
<p><b>Task 11</b> Close the query Save it, giving a suitable name</p>	<p><b>Step 1</b></p> <p>Close the query by clicking , saving any changes when asked</p> <p>When asked, give the query a name that conforms to the naming convention, such as <b>qryMaintenanceList</b></p>


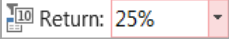
<b>Learning Objective Two: A query with criteria</b>	
<b>Task 1</b> Continue work in <code>Inventory.accdb</code>	
<b>Task 2</b> Create a query listing assets with their description, make, model, date acquired and purchase price	<b>Step 1</b> Create a new query in Design View, with <code>tblAssets</code> only
	<b>Step 2</b> Add these fields: <b>AssetDescription, Make, Model, DateAcquired, PurchasePrice</b>
	<b>Step 3</b> Run the query
<b>Task 3</b> Add criteria to show only laptops Save the query so far	<b>Step 1</b> Switch back to Design View for the same query In the <b>AssetDescription</b> column, on the Criteria row, type <code>laptop</code>
	<b>Step 2</b> Run the query
	<b>Step 3</b> Save the query, by clicking  on the Quick Access Toolbar (a suitable name might be <code>qryAssetsWithPrices</code> )
<b>Task 4</b> Edit the criteria so the query now shows only printers Then change the criteria to show all assets made by Watkins	<b>Step 1</b> In Design View of the same query, delete <code>laptop</code> from the Criteria row
	<b>Step 2</b> Enter <code>printer</code> as the Criteria for Description Run the query
	<b>Step 3</b> Delete any previous criteria (check <i>all</i> the columns) In the <b>Make</b> column, enter <code>watkins</code> Run the query
<b>Task 5</b> Edit the same query to show all assets which cost £200 or more	<b>Step 1</b> Delete any previous criteria (check <i>all</i> columns)
	<b>Step 2</b> In the <b>PurchasePrice</b> column, enter <code>&gt;=200</code> (note you do not need to type the £ symbol, as that is part of the format for this field)
	<b>Step 3</b> Run the query and look at the results



<p><b>Task 6</b> Use this query to show all laptops which cost over £350</p>	<p><b>Step 1</b> Delete any previous criteria In the <b>Description</b> column enter the criteria <b>laptop</b></p>
	<p><b>Step 2</b> In the <b>PurchasePrice</b> column, in the same row, enter <b>&gt;350</b></p>
	<p><b>Step 3</b> Run the query If you do not get 3 records, check both criteria and confirm that each one is in the correct column and in the correct row</p>
<p><b>Task 7</b> Revise the query to show all assets made by Watkins, Wilkinson or Wood Associates (hint: use the * wildcard symbol)</p>	<p><b>Step 1</b> Delete any previous criteria In the <b>Make</b> column, enter the criteria <b>w*</b> <i>Access edits this to Like "w*"</i></p>
	<p><b>Step 2</b> Run the query</p>
<p><b>Task 8</b> Revise the query to show all machines whose model number is 180 to 190 (use <b>Between</b>)</p>	<p><b>Step 1</b> Delete any previous criteria In the <b>Model</b> column, enter the criteria <b>between 180 and 190</b></p>
	<p><b>Step 2</b> Run the query</p>
<p><b>Task 9</b> Revise the query to show all machines which are not laptops</p>	<p><b>Step 1</b> Delete any previous criteria In the <b>AssetDescription</b> column, enter the criteria <b>not laptop</b></p>
	<p><b>Step 2</b> Run the query</p>
<p><b>Task 10</b> Find all records for machines whose acquisition was not recorded (no value has been entered in the <b>DateAcquired</b> field)</p>	<p><b>Step 1</b> Delete any previous criteria Add the field <b>DateAcquired</b> to an empty column on the query design grid (if necessary)</p>
	<p><b>Step 2</b> Under <b>DateAcquired</b>, enter <b>Is Null</b></p>
	<p><b>Step 3</b> Run the query This suggests that 22 of the machines have not yet been sold, as no <b>DateAcquired</b> has been entered</p>

<p><b>Task 11</b> Revise this query again, to show a list of laptops and PCs and printers</p>	<p><b>Step 1</b> Delete any previous criteria In the <b>Description</b> column, enter <b>laptop</b></p> <p><b>Step 2</b> In the next row of the <b>Description</b> column, enter <b>PC</b>, and on the row below enter <b>printer</b></p> <p><b>Step 3</b> Run the query</p>
<p><b>Task 12</b> Sort the results in order of the date the asset was bought (acquired), oldest items first  Then re-sort in alphabetical order of Department name, with a secondary sort of Purchase Price</p>	<p><b>Step 1</b> In the <b>DateAcquired</b> column, in the <b>Sort</b> row, choose <b>Ascending</b></p> <p><b>Step 2</b> Run the query and notice the order the records appear</p> <p><b>Step 3</b> Remove the previous sort setting: select <b>Not Sorted</b> in the drop-down list  Move the <b>DeptName</b> column so it appears to the left of the <b>PurchasePrice</b> column</p> <p><b>Step 4</b> Select <b>Ascending</b> in the <b>Sort</b> row in the <b>DeptName</b> column and also in the <b>PurchasePrice</b> column</p> <p><b>Step 5</b> Run the query and check the sorting is as expected (alphabetical order by Department name with a secondary sort based on the purchase price)</p>
<p><b>Task 13</b> Close the query, saving if necessary</p>	

<b>Learning Objective Three: Create some interesting kinds of query</b>	
<b>Task 1</b> Continue work in <code>Inventory.accdb</code>	
<b>Task 2</b> Create a new query based on <code>tblAssets</code>  Set up criteria so that the user can ask, when they run the query, for only one kind of asset	<b>Step 1</b> Create a new query in design view Base it on <code>tblAssets</code>
	<b>Step 2</b> Choose the fields <code>AssetDescription</code> , <code>Make</code> , and <code>DateAcquired</code> and <code>DateSold</code>
	<b>Step 3</b> In the <b>Criteria</b> row, enter <code>[Which kind of machine?]</code> in the <code>AssetDescription</code> field
<b>Task 3</b> Test the query and save it as <code>qrySingleMachineType</code>	<b>Step 1</b> Run the query  In the <b>Enter Parameter Value</b> dialog, type <code>laptop</code>  Only laptop records are displayed
	<b>Step 2</b> Run the query again, requesting only printers
	<b>Step 3</b> Save the query with the name <code>qrySingleMachineType</code> (do not close it yet)
<b>Task 4</b> Change the criteria so that the user can ask for machines by make, just giving the first character	<b>Step 1</b> Remove the <code>AssetDescription</code> criteria from the query design grid
	<b>Step 2</b> In the <code>Make</code> field, enter the parameter criteria: <code>Like[Which make do you want?]&amp;"**"</code>
<b>Task 5</b> Test the query and save it as <code>qryMachinesByMake</code>	<b>Step 1</b> Run the query
	<b>Step 2</b> When asked, type a single letter such as <code>s</code> , so as to display only records whose makes begin with that letter
	<b>Step 3</b> Think: how would you use this query to display only machines made by Watkins?
	<b>Step 4</b> Choose <code>File Save As</code> , and give the query the name <code>qryMachinesByMake</code>

<p><b>Task 6</b> Create a new query based on <b>tblAssets</b></p> <p>Include <b>Description</b>, <b>Serial Number</b> and <b>Purchase Price</b></p> <p>Set the query to display only the 5 most expensive items</p>	<p><b>Step 1</b> Click  on the <b>Create</b> tab</p> <p><b>Step 2</b> Base the new query on <b>tblAssets</b></p> <p><b>Step 3</b> Choose the fields <b>Description</b>, <b>SerialNumber</b> and <b>PurchasePrice</b></p> <p><b>Step 4</b> Set a descending sort by <b>PurchasePrice</b></p> <p><b>Step 5</b> Click the <b>Return Top Values</b> control  and choose 5, so as to display the 5 items with the top purchase prices</p> <p><b>Step 6</b> Run the query and observe the result</p>
<p><b>Task 7</b> Now display only the 10 cheapest items</p>	<p><b>Step 1</b> In Design View, sort the records in ascending order of purchase price</p> <p><b>Step 2</b> Use the <b>Return Top Values</b> control and enter 10</p> <p><b>Step 3</b> Run the query and observe that only the 10 cheapest items are listed</p>
<p><b>Task 8</b> Change the query to display the 10% of items bought most recently</p>	<p><b>Step 1</b> Remove the sort and add a <b>DateAcquired</b> field to the grid</p> <p><b>Step 2</b> Sort by this field in descending order</p> <p><b>Step 3</b> Set the Top Values control to 10% (type 10% in the control, then press ENTER)</p> <p><b>Step 4</b> Run the query Close the query, saving</p>

**Learning Objective Four: Create some housekeeping queries**


**Task 1**

Continue work in `Inventory.accdb`

**Task 2**

Create a new query to find duplicate asset entries

**Step 1**

Click  on the **Create** tab

Choose **Find Duplicates Query Wizard**

Choose to base the query on `tblAssets`

**Step 2**

Fields that might contain duplicate information include **AssetDescription** and **SerialNum**

**Step 3**

Additional fields to show should include **Make**, **Model** and **DateAcquired**

**Step 4**

Name the query `qryFindDupAssets`

**Step 5**

The query runs when you finish the wizard

**Step 6**

Note any duplicate records

(Now you might go ahead and delete unwanted records)

Close the query

**Task 3**

Create a new query to find any assets which have not yet been assigned to any Department (`DeptID`)

**Step 1**

Click  then choose **Find Unmatched Query Wizard**

Choose `tblAssets`

**Step 2**

`tblDepartments` contains related records

The field `DepartmentID` (or `DeptID`) is in both tables

**Step 3**

In query results, show the fields **AssetDescription**, **Make**, **Model** and **SerialNum**


Name the new query `qryFindUnmatchedAssets`

**Step 4**


The query runs when you finish the wizard

Here you might edit the data to reconcile the values between the two tables





Close the query


<p><b>Task 4</b>                  Create a new crosstab query:                  Base it on  <b>qryPriceByDescrAndDept</b>                  Choose <b>Description</b> and  <b>DeptName</b> as the row and                  column headings                  Calculate the Sum of  <b>PurchasePrice</b></p>	<p><b>Step 1</b>                  Click  and choose <b>Crosstab Query Wizard</b>                  Base the new query on a previous query  <b>qryPriceByDescrAndDept</b></p> <hr/> <p><b>Step 2</b>                  Choose <b>AssetDescription</b> for the row headings</p> <hr/> <p><b>Step 3</b>                  Choose <b>DeptName</b> for the column headings</p> <hr/> <p><b>Step 4</b>  <b>PurchasePrice</b> is the field for calculations                  Choose the <b>Sum</b> function</p> <hr/> <p><b>Step 5</b>                  Name the new query  <b>qryCrossTabAssetsAndDepts</b></p>
<p><b>Task 5</b>                  Run the query                  Look at the money spent by each department on purchasing each type of asset                  This crosstab helps you compare the spending of the departments in the various                  areas of expenditure</p>	
<p><b>Task 6</b>                  Close the query, saving if you wish</p>	


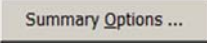

<b>Learning Objective Five: Calculations in a query</b>	
<p><b>Task 1</b> Create a new query based on <b>tblAssets</b></p> <p>Include asset descriptions and purchase price</p>	<p><b>Step 1</b> Create a new query in Design view Base the query on <b>tblAssets</b></p>
	<p><b>Step 2</b> Choose the fields <b>AssetDescription</b> and <b>PurchasePrice</b></p>
<p><b>Task 2</b> Add a calculated column to find the depreciated value of each asset (we are told this is 1/3rd of the purchase price)</p> <p>Edit the column heading to be <b>Depreciated</b></p>	<p><b>Step 1</b> In an empty column on the query design grid, enter an expression in the Field row: <b>Depreciated: [PurchasePrice]/3</b></p>
	<p><b>Step 2</b> Run the query and check that the calculated values are plausible</p>
<p><b>Task 3</b> Add more columns for dates when an asset was acquired and sold</p> <p>Calculate how many days each asset was kept; give a suitable column heading</p>	<p><b>Step 1</b> Add new columns for the fields <b>DateAcquired</b> and <b>DateSold</b></p>
	<p><b>Step 2</b> In an empty column, enter an expression in the Field row: <b>DaysKept: [DateSold]-[DateAcquired]</b></p>
	<p><b>Step 3</b> Run the query and check the calculated values are plausible</p>
<p><b>Task 4</b> Add a column to show the asset make and model concatenated into one text</p>	<p><b>Step 1</b> In an empty column in the query design grid, enter an expression in the Field row: <b>AssetName: [Make]&amp;" "&amp;[Model]</b></p> <p>Note: make sure there is a space between the double quote marks</p>
	<p><b>Step 2</b> Run the query and check the calculated values are sensible</p>
<p><b>Task 5</b> Close the query, saving it as <b>qryAssetDetails</b></p>	
<p><b>Task 6</b> Create a new query based on <b>tblAssets</b></p> <p>Include <b>AssetID</b> and <b>AssetDescription</b></p>	<p><b>Step 1</b> Create a new query Base it on <b>tblAssets</b></p>
	<p><b>Step 2</b> Include the fields <b>AssetID</b> and <b>AssetDescription</b></p>


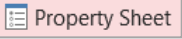
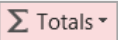
<p><b>Task 7</b></p> <p>Group records by Asset Description</p> <p>Count how many of each type of asset has been purchased</p> <p>Save and close the query</p>	<p><b>Step 1</b></p> <p>Click  to show the Total row</p>
	<p><b>Step 2</b></p> <p>Under <b>AssetDescription</b>, choose <b>Group By</b></p>
	<p><b>Step 3</b></p> <p>Under <b>AssetID</b>, choose <b>Count</b></p>
	<p><b>Step 4</b></p> <p>Run the query and inspect the result</p>
<p><b>Task 8</b></p> <p>Save the query as <b>qryCountEachAssetType</b>, and close it</p>	


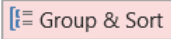
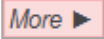


<b>Learning Objective Six: Calculations on a form</b>	
<p><b>Task 1</b> Continue work in <b>Inventory.accdb</b> Open <b>frmAllAssets</b> in Design View</p>	
<p><b>Task 2</b> These activities are described in Design View, but can also be carried out in Layout View if you prefer</p>	
<p><b>Task 3</b> Add an unbound control Enter an expression to calculate the VAT payable on the purchase price (this is 20% of the price paid)</p>	<p><b>Step 1</b> Click  then click in a space in the form, to add an unbound control</p>
	<p><b>Step 2</b> In the unbound control, enter an expression: <code>=0.2*[PurchasePrice]</code></p>
	<p><b>Step 3</b> Click in the associated label and replace <b>Text :</b> with <b>VAT payable</b></p>
<p><b>Task 4</b> Add a calculated control which shows the head of department's name and phone number in this way: Jim Hacker (72345) – by using the Expression Builder</p>	<p><b>Step 1</b> Click  then click in a space in the form to add an unbound control</p>
	<p><b>Step 2</b> Click  <b>Property Sheet</b> to show the Property Sheet, and select the new unbound control so that its properties are listed</p>
	<p><b>Step 3</b> Click in the <b>Control Source</b> row, then click  to open the Expression Builder</p>
	<p><b>Step 4</b> Choose <b>frmAllAssets</b> in the left panel, then double-click <b>DeptHead</b> in the middle, to start building an expression</p>
	<p><b>Step 5</b> Build up this expression, using a combination of typing characters and choosing Operators and field names from the list: <code>= [DeptHead] &amp; " (" &amp; [ContactTelNum] &amp; ") "</code> Take care that the space and opening bracket are in quote marks, and the closing bracket is in quote marks</p>

	<p><b>Step 6</b> When ready, click  to insert the expression into the <b>Control Source</b> property of the text box Click on the grey desktop to finalise the calculated control</p>
<p><b>Task 5</b> You may now want to delete the separate controls for <b>DeptHead</b> and <b>ContactTelNum</b> from the report</p>	
<p><b>Task 6</b> View the form and inspect the two calculated controls on several records Close the form, saving changes</p>	

<b>Learning Objective Seven: Calculations on a report</b>	
<b>Task 1</b> Continue work in Inventory.accdb	
<b>Task 2</b> Using the Report wizard, create a new columnar report on the assets  For each asset, include the description, depreciable life, purchase price, make and model	<b>Step 1</b> Click  <b>Report Wizard</b> on the <b>Create</b> tab of the ribbon to start the wizard
	<b>Step 2</b> In the wizard, choose the table called <b>tblAssets</b>
	<b>Step 3</b> Choose these fields:  <b>description, depreciable life, purchase price, make and model</b>
	<b>Step 4</b> Group by Description (so that all laptops are put together, all printers, and so on)  Set a sort order if you wish
<b>Task 3</b> Use the Summary Options button to set up calculations: you want to know the average purchase price for each group, and the longest expected life	<b>Step 1</b> On the Sort Order page of the wizard, click   In the summary options page, tick the box for <b>Average</b> under <b>Purchase Price</b>  Tick the box for <b>Maximum</b> under <b>Depreciable Life</b>
	<b>Task 4</b> Complete the wizard  Name the report <b>rptAssetPrices</b>  Examine the finished report in Print Preview  Close the report
	<b>Step 1</b> Complete the wizard with options of your choice
	<b>Step 2</b> In the last part of the wizard, give a name for the report <b>rptAssetPrices</b>
	<b>Step 3</b> When you finish the wizard, the report opens in Print Preview
	<b>Step 4</b> Confirm that the expected fields are included  Navigate between the pages and look at the data  Notice the calculations that summarise the groups
	<b>Step 5</b> Click  to close the report

<p><b>Task 5</b> In the same report, use an unbound text box in the Detail to calculate the (depreciated) end date of each asset - use the <b>DateAcquired</b> and the <b>DepreciableLife</b></p> <p>NB Multiply the number of years of depreciable life by 365 to find the number of days</p> <p>NB2 We will neglect leap years for simplicity</p>	<p><b>Step 1</b> Open the new report in Design View</p> <p><b>Step 2</b> Click  then click in a space in the <b>Detail</b> section of the report, to create an unbound control</p> <p><b>Step 3</b> Click in the unbound control and type an expression for the end date of an asset: <code>=[DateAcquired]+([DepreciableLife]*365)</code></p>
<p><b>Task 6</b> Assign the <b>Long Date</b> format</p> <p>Give suitable text for the label</p>	<p><b>Step 1</b> With the calculated control selected, click  if necessary to display the Properties</p> <p><b>Step 2</b> On the <b>Format</b> tab of the Property Sheet, select the format <b>Long Date</b></p> <p><b>Step 3</b> Delete the text in the associated label Type <b>End date</b></p>
<p><b>Task 7</b> Add a summarising control in the Report Footer, to calculate the total of the purchase prices</p>	<p><b>Step 1</b> Select the Purchase Price control (in the Detail)</p> <p><b>Step 2</b> Click </p> <p><b>Step 3</b> Choose <b>Sum</b> from the list of functions</p> <p><b>Step 4</b> A control appears in the Report Footer, with the expression: <code>=sum([PurchasePrice])</code></p>
<p><b>Task 8</b> Format the control to show currency; give suitable label text</p>	<p><b>Step 1</b> With the calculated control selected, edit its properties: select the <b>Currency</b> format</p> <p><b>Step 2</b> Replace the label text with <b>Total Spend</b></p>
<p><b>Task 9</b> Preview the report, and examine the calculations after each record and at the end of the report</p> <p>Name the report <b>rptAssetDepreciation</b>, and close it</p>	


<p><b>Task 10</b> Note the following tasks are described using the <b>Group, Sort and Totals</b> pane, but you can achieve similar results using  in a grouped report</p>	
<p><b>Task 11</b> Open the report <b>rptAssetsByDepartment</b> in Design View</p>	<p><b>Step 1</b> Preview <b>rptAssetsByDepartment</b> Inspect the way the records are laid out: assets for each department are grouped together</p>
	<p><b>Step 2</b> Switch to Design View Notice the different sections of the report: the Detail, the Report Header and Footer, the Page Header and Footer and the grouped DeptID Header and Footer</p>
<p><b>Task 12</b> Use the <b>Group, Sort and Total</b> pane to add a calculated control in the DeptID Footer, to show the average purchase price spent by each department</p>	<p><b>Step 1</b> Click </p>
	<p><b>Step 2</b> In the <b>Group, Sort and Total</b> pane, click the <b>GroupID</b> bar Click  for more options</p>
	<p><b>Step 3</b> Click the arrow beside <b>with no totals</b> to show some further options</p>
	<p><b>Step 4</b> Set <b>Total</b> on <b>PurchasePrice</b> Set <b>Type</b> to be <b>Average</b> Check <b>Show Grand Total</b> and <b>Show in Group Footer</b></p>
	<p><b>Step 5</b> A control appears in the DeptID Footer, with the expression: <b>=average([PurchasePrice])</b> Another control with the same expression appears in the Report Footer</p>
<p><b>Task 13</b> Format and label both of the new controls</p>	<p><b>Step 1</b> With the calculated control in the group footer selected, apply Currency format</p>
	<p><b>Step 2</b> Replace the label with the text <b>Average Purchase for the Department</b> You may need to resize or move the label</p>
	<p><b>Step 3</b> Repeat this for the grand total in the Report Footer</p>

<p><b>Task 14</b> Add another summarising control, which shows how many assets each department holds</p>	<p><b>Step 1</b> Use the Totals option on the <b>DeptID</b> Group bar (in the Group, Sort and Total pane) to set: <b>Total by Asset Description</b> <b>Type is Count Records</b> <b>Show Grand Total</b> <b>Show in Group Footer</b></p>
<p><b>Task 15</b> Run the report and look at the values calculated for each department Close the report, saving changes</p>	
<p><b>Task 16</b> Close the <b>Inventory</b> database file, leaving <i>Access</i> open</p>	



**Learning Objective Eight: Conditional formatting**

In this database, a dental surgery is keeping track of patients and their appointments

**Task 1**  
 Open **New Dentists.accdb**  
 Look at the Relationships diagram to discover the tables and their links

**Step 1**  
 Display the **Open** dialog eg. using  on the **File** menu

**Step 2**  
 Find and open **New Dentists.accdb**

**Step 3**  
 Click  on the **Database Tools** tab, to display the Relationships diagram

**Step 4**  
 Examine the tables and their fields, and the links between them  
 In this database, a dental surgery is keeping track of patients and their appointments  
 Close the diagram



**Task 2**  
 Open **frmPatientsPlain** in Design View

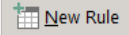
**Step 1**  
 Open **frmPatientsPlain** in Design View, by right-clicking the form name in the Navigation Pane then selecting **Design View**

**Step 2**  
 Examine the form, noticing the controls that are in place




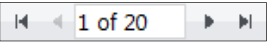
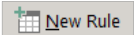


**Task 3**  
 Set conditional formatting so that if a patient was born before 1<sup>st</sup> January 1960 their Date Of Birth control is emphasised

**Step 1**  
 Select the **Date Of Birth** control (the text box control where the data values appear, not the label beside it)

**Step 2**  
 On the **Format** tab of the ribbon, click   
 In the **Conditional Formatting** dialog, confirm that the **DOB** field control is selected

**Step 3**  
 Click 





**Step 4**  
 The new rule will **Check values in the current record**  
 Set it to **Format only cells where the field value is less than 1/1/1960**









	<p><b>Step 5</b> In the Preview, choose a bright fill colour and a contrasting font colour</p> <p><b>Step 6</b> Click  to create the rule Click  to close the dialog</p>
<p><b>Task 4</b> Examine several records, to see whether the conditional formatting is working as expected</p>	<p><b>Step 1</b> Switch to Form View by clicking  on the <b>Home</b> tab The records are displayed one at a time</p> <p><b>Step 2</b> Use the navigation buttons at the bottom of the form to page between all the records </p> <p><b>Step 3</b> When you reach a patient who was born before 1<sup>st</sup> January 1960, confirm that the DOB field control has changed colour</p>
<p><b>Task 5</b> Add another rule to the DOB field control, to draw attention if a date is accidentally entered that is after today's date  Hint: Look among the pre-defined functions for the computer's date</p>	<p><b>Step 1</b> Switch to Design View With the Date Of Birth control selected, display the Conditional Formatting dialog again</p> <p><b>Step 2</b> Click  to add a new rule The new rule is to <b>Check values in the current record</b></p> <p><b>Step 3</b> Set up to <b>Format only cells where</b> the field value is more than today's date: Choose <b>Field Value Is More Than</b></p> <p><b>Step 4</b> Click  In the Expression Builder: In the first column choose <b>Functions</b> then <b>Built-In Functions</b> In the middle column choose <b>Date/Time</b> In the last column choose <b>Date</b> Click </p> <p><b>Step 5</b> Check that the expression box now contains <b>Date ( )</b> In the Preview, choose another eye-catching fill colour and a contrasting font colour</p>



	<p><b>Step 6</b></p> <p>Click <input type="button" value="OK"/> to create the rule</p> <p>This control now has 2 conditions set: if the Date Of Birth is pre-1960 it appears with a bright colour to remind the staff , and if it is after today it appears with another colour to point out that this is likely to be a mis-type</p> <p>Click <input type="button" value="OK"/> to close the dialog</p>
<p><b>Task 6</b></p> <p>Run the form</p> <p>Confirm that the conditional formatting is appearing as expected</p> <p>Save changes and close the form</p>	

<b>Learning Objective Nine: Action queries</b>	
<p><b>Task 1</b> Continue working in the database <b>New Dentists.accdb</b> All the Odd-job Men are now to be known as Senior Administrative Supervisors</p>	
<p><b>Task 2</b> Create a query based on <b>tblStaffContactInfo</b> Include all fields Name it <b>qryStaffContactInfo</b></p>	<p><b>Step 1</b> Click  in the <b>Create</b> tab</p>
	<p><b>Step 2</b> In the Add dialog, choose <b>tblStaffContactInfo</b></p>
	<p><b>Step 3</b> Set up columns for the fields <b>ID</b>, <b>StaffMember</b>, <b>email</b> and <b>jobtitle</b></p>
	<p><b>Step 4</b> Save the query as <b>qryStaffContactInfo</b> Run the query and look briefly at the data</p>
<p><b>Task 3</b> Create an update query called <b>qryOddJobUpgrade</b>, to make this change Run the update query</p>	<p><b>Step 1</b> Work on <b>qryStaffContactInfo</b> to create an update query</p>
	<p><b>Step 2</b> In the <b>jobtitle</b> field, enter <b>Odd- job man</b> in the <b>Criteria</b> row Run the query to confirm that only the 2 Odd-job men are included</p>
	<p><b>Step 3</b> In Design View, click  The <b>Update To</b> row appears in the design grid</p>
	<p><b>Step 4</b> In the <b>Update To</b> row, in the <b>Jobtitle</b> field, enter <b>Senior Administrative Supervisor</b></p>
	<p><b>Step 5</b> Click  to run the update query Accept the warning about updating 2 rows</p>
	<p><b>Step 6</b> Close the query, saving it as <b>qryOddJobUpgrade</b></p>
	<p><b>Step 7</b> Open <b>tblStaffContactInfo</b> and check that the odd-job men now have new job titles</p>

<p><b>Task 4</b> As a precaution, make a backup copy of <b>tblAppointment</b>, naming it <b>tblApptCopy</b></p>	<p><b>Step 1</b> Select <b>tblAppointment</b> and copy it (e.g. use  )</p> <p><b>Step 2</b>  a copy into the list of tables Give the new table the name <b>tblApptCopy</b> This copy of the data may be needed in case of a mistake with an action query</p>
<p><b>Task 5</b> In <b>tblAppointment</b>, use a Delete query to remove all records for appointments which were not kept</p>	<p><b>Step 1</b> Create a new query based on the table <b>tblAppointment</b> (use <b>Create query</b> in Design View)</p> <p><b>Step 2</b> Include the fields <b>AppointmentID</b> and <b>AppointmentKept</b></p> <p><b>Step 3</b> Set criteria 0 under <b>AppointmentKept</b> (recall that value zero 0 represents “no” in a Yes/No field) Run the query</p> <p><b>Step 4</b> Use  and convert this to a Delete query</p> <p><b>Step 5</b> Click  to run the Delete query Accept the warning about deleting 21 records</p>
<p><b>Task 6</b> Save this Delete query as <b>qryFailedAppointments</b></p>	<p><b>Step 1</b> Close the query, saving it when asked as <b>qryFailedAppointments</b></p> <p><b>Step 2</b> Open <b>tblAppointments</b> again, and notice that the records have been reduced to 59 Close the table</p>
<p><b>Task 7</b> For those patients who go to school, make a list of the patient’s name, their school and a contact phone number at the school</p>	<p><b>Step 1</b> Create a new query using <b>tblPatient</b> and <b>tblSchool</b> Include <b>FirstName</b> and <b>LastName</b> from <b>tblPatient</b> Include <b>SchoolName</b> and <b>ContactPhone</b> from <b>tblSchool</b></p> <p><b>Step 2</b> Run the query and look at the 7 records</p>

<p>Use a query to make a new table with this information called <b>tblSchoolContacts</b></p> <p>Save this query as <b>qrySchoolContacts</b></p>	<p><b>Step 3</b></p>  <p>Use  to convert the query to a make-table query</p> <p>Specify the new table name <b>tblSchoolContacts</b></p>										
	<p><b>Step 4</b></p>  <p>Click  to run the make-table query</p> <p>Accept the warning about pasting 7 records</p>										
	<p><b>Step 5</b></p> <p>The new table is seen in the Navigation Pane</p>										
	<p><b>Step 6</b></p> <p>Close and save the query as <b>qrySchoolContacts</b></p>										
<p><b>Task 8</b></p> <p>The details of some additional people are given in <b>tblMorePeople</b>; append them to <b>tblPatient</b></p> <p>Save this query as <b>qryAppendMorePeople</b></p>	<p><b>Step 1</b></p> <p>Create a new query based on <b>tblMorePeople</b></p> <p>Include all fields except <b>ID</b></p>										
	<p><b>Step 2</b></p>  <p>Use  and convert this to an Append query</p> <p>Select the table <b>tblPatient</b> in the current database</p>										
	<p><b>Step 3</b></p> <p>In the <b>Append To</b> row, match up the field names:</p> <table border="0" data-bbox="769 1173 1260 1321"> <tr> <td><b>Surname</b></td> <td><b>LastName</b></td> </tr> <tr> <td><b>Forename</b></td> <td><b>FirstName</b></td> </tr> <tr> <td><b>Address4</b></td> <td><b>PostCode</b></td> </tr> <tr> <td><b>TelNo</b></td> <td><b>HomePhoneNum</b></td> </tr> <tr> <td><b>DateOfBirth</b></td> <td><b>DOB</b></td> </tr> </table>	<b>Surname</b>	<b>LastName</b>	<b>Forename</b>	<b>FirstName</b>	<b>Address4</b>	<b>PostCode</b>	<b>TelNo</b>	<b>HomePhoneNum</b>	<b>DateOfBirth</b>	<b>DOB</b>
	<b>Surname</b>	<b>LastName</b>									
	<b>Forename</b>	<b>FirstName</b>									
	<b>Address4</b>	<b>PostCode</b>									
<b>TelNo</b>	<b>HomePhoneNum</b>										
<b>DateOfBirth</b>	<b>DOB</b>										
<p><b>Step 4</b></p>  <p>Click  to run the append query</p> <p>Accept the warning about appending 43 records</p>											
<p><b>Step 5</b></p> <p>Save the query as <b>qryAppendMorePeople</b>, and close it</p>											
<p><b>Step 6</b></p> <p>Open <b>tblPatient</b> and confirm that the records have been appended</p>											
<p><b>Task 9</b></p> <p>Close all queries and tables</p> <p>Close the database, leaving the computer at the <i>Windows</i> desktop</p>											