Tutorial 4: Queries

Queries perform many functions, the most common of which is to extract data according to set criteria.

Some of the things you can do with queries:

- Bringing together the records from one or more tables.
- Sort data according to the values of one or more field.
- Select which fields to show
- Select which records to show by specifying selection criteria
- Perform calculations
- Performing actions on data e.g. Updating tables, deleting records

How to create a simple query

1 In the Database window, click the Queries tab, and then click New.
2 In the New Query dialog box, click Design View, and then click OK.
3 In the Show Table dialog box, click the tab that lists the objects whose data you want to work with.
4 Double-click the name of each object you want to add to the query, and then click Close.

5 If you have multiple tables or queries in the query, make sure they are connected to each other with a join line so that Microsoft Access knows how the information is related. If they aren't connected, create the join line yourself.

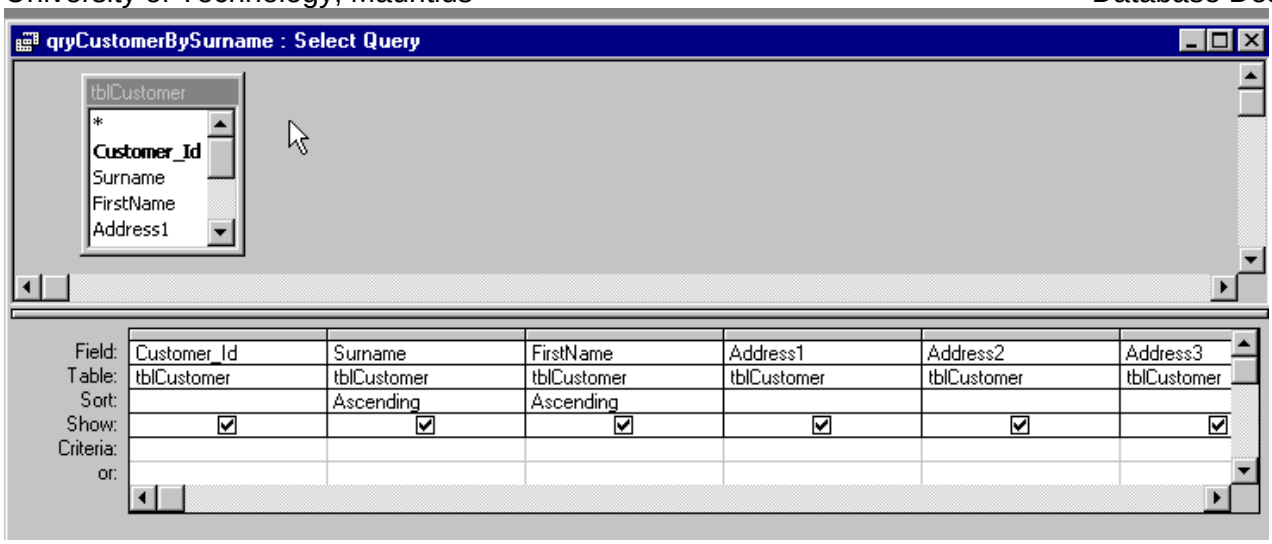
6 Add fields to the query by dragging the field names from the field list to the design grid.

7 Refine your query by entering criteria, adding a sort order, creating calculated fields, computing the sum, average, count, or another type of total on the data it retrieves, or otherwise modifying the query's design.

7 To save the query, click Save on the toolbar.

8 To see the results of the query, click View on the toolbar.

Example:



qryCustomerBySurname Query

This query shows all the fields from the Customer Table, sorted in ascending alphabetical order on the Customer's surname and forename.

Select New Query
Add Customer Table

Add all fields from the Customer Table :
Select all fields by

Clicking 1st field then click on the last field while holding SHIFT
Drag selected block into the query field grid

Sort Surname ascending
FirstName ascending

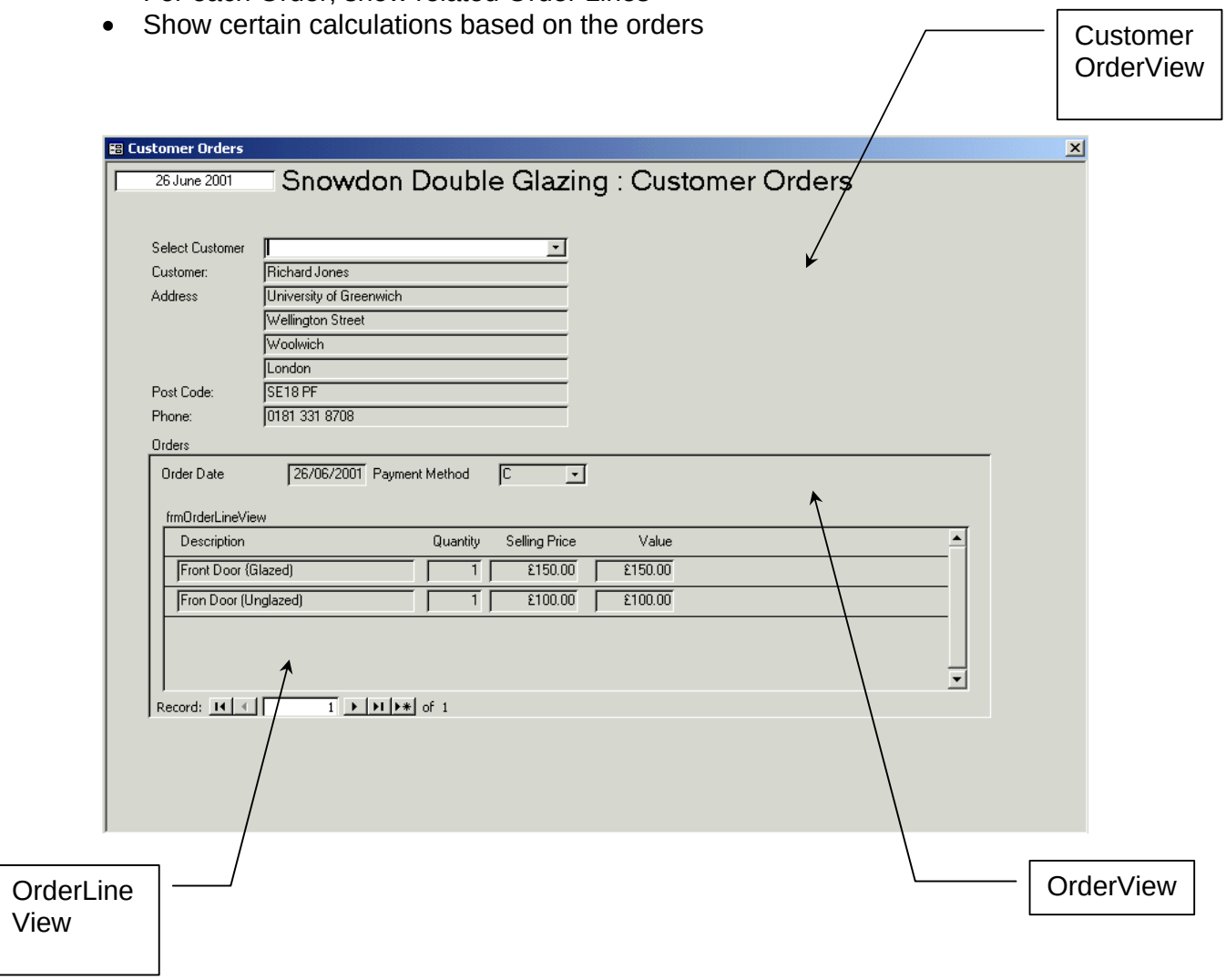
This will ensure that the Customers will be sorted primarily by Surname, and WITHIN THAT, by the FirstName. The sort order is governed by the order of the fields in the query with priority starting from the left.

Save the query

Creating a Customer Order View Form

We are now going to create a form which allows the user to

- Select a particular customer
- Show customer details
- Show related Orders
- For each Order, show related Order Lines
- Show certain calculations based on the orders



You should analyse your requirements before starting:

The CustomerOrderView Form shows data from the Customer table.

The OrderView subform shows data from the Orders table.
It will be linked to the main form through the Customer_Id.

The OrderLineView subform will be a subform of the OrderView subform.
It will show data both from the OrderLine table and the Stock table.
It will be linked to the OrderView subform through the Order_Id.

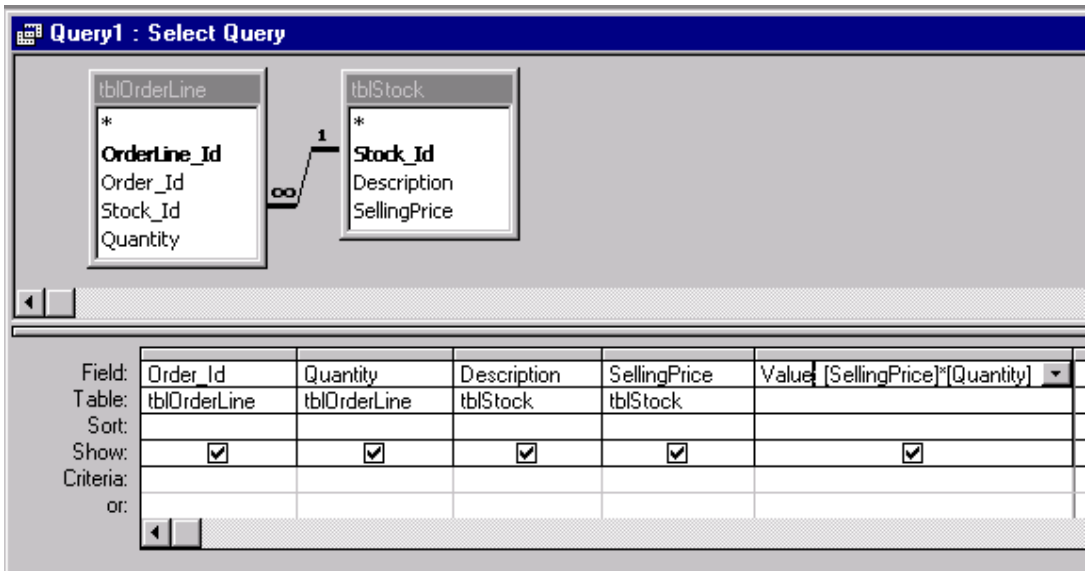
The form will show the calculated value of each order

The form will also show how many order lines there are for each order and the total value of the orders.

You need to construct these forms in reverse order.

The OrderLineView form

Create a query called *qryOrderLineStock* to show data from both the OrderLine table and the Stock table:



Calculated Fields

You can create fields in a query by using an expression to calculate values. The expression is entered in the Field row of a column. The Expression Builder should be used whenever you make reference to other objects and controls, since it is a great aid to getting the syntax correct.

You need to add an Expression or Calculated field in a new column of the query, which will calculate the value of the particular order.

In the Field row of an empty column, enter: *[SellingPrice]*[Quantity]*

The field will initially be called *Expr1* by Access.
Replace *Expr1* with the name *Value.*

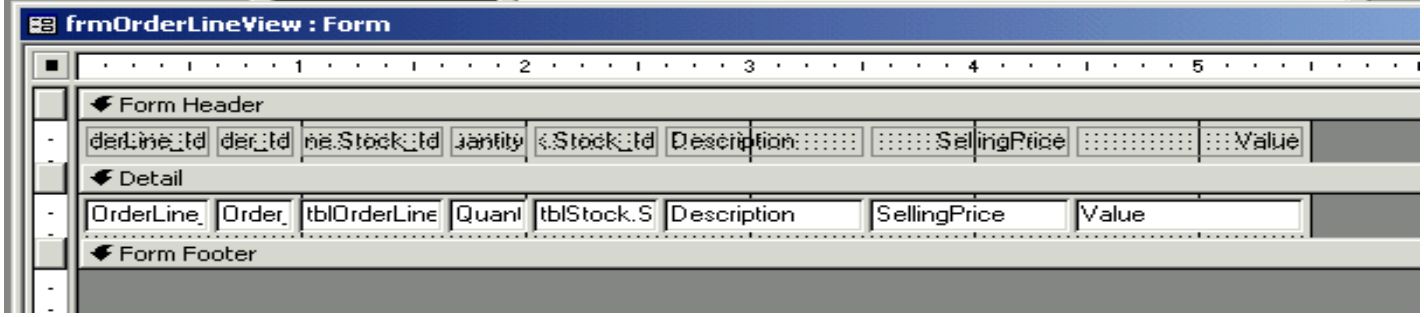
It is not necessary to store the value of an order in a table in the database, because it can be *derived* or *induced* from other data whenever it is required.

Create an OrderLineView form, based on the *qryOrderLine_Stock* query, using the AutoForm:Tabular form wizard.

Save the form under the name frmOrderLineView

Change the following properties:

Scroll bar	Vertical
Allow additions	No
Record selectors	No
Navigation buttons	No



Disable all the text box controls by setting the Enabled property of each to No, and the Locked properties to Yes. You can also change the Background colour of these text boxes to Transparent, to indicate to the user that they cannot be edited.

The form in design view will look something like this:

Alter the form so that when it is viewed it looks like this:

The OrderView Form

Create a query called qryOrderByDateDesc which sorts the Order table by the OrderDate in descending order. This therefore shows the most recent orders first.

Create a column Order form based on the qryOrderByDateDesc query using the Autoform Columnar form wizard.

Save the form with the name frmOrderView.

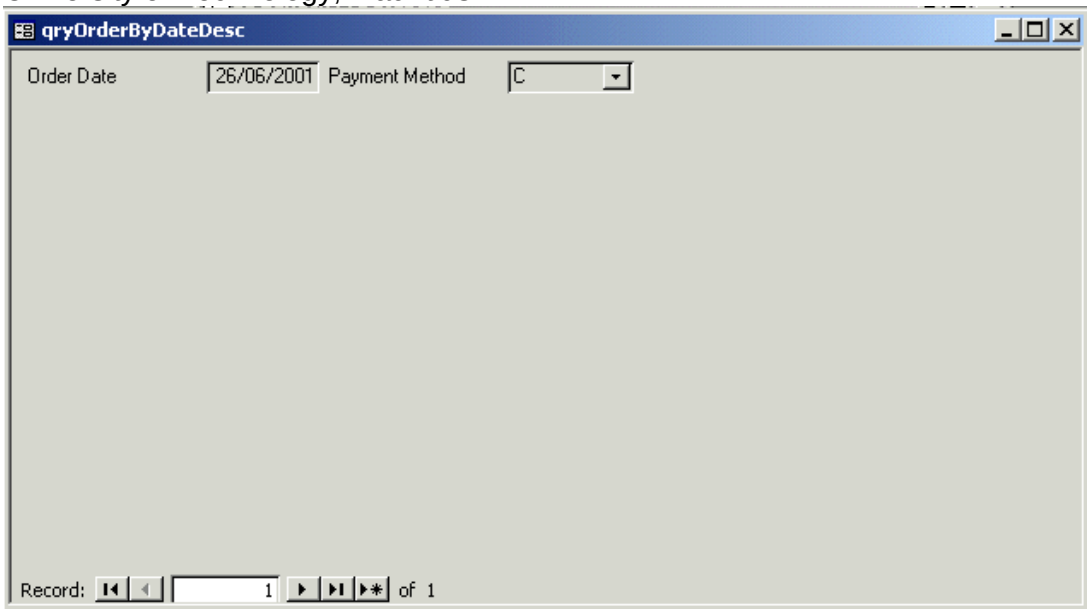
Change the following properties:

Scroll bar	Neither
Allow additions	No
Record selectors	No
Navigation buttons	Yes
Dividing Lines	No

Disable all the text box controls and change the background to Transparent.

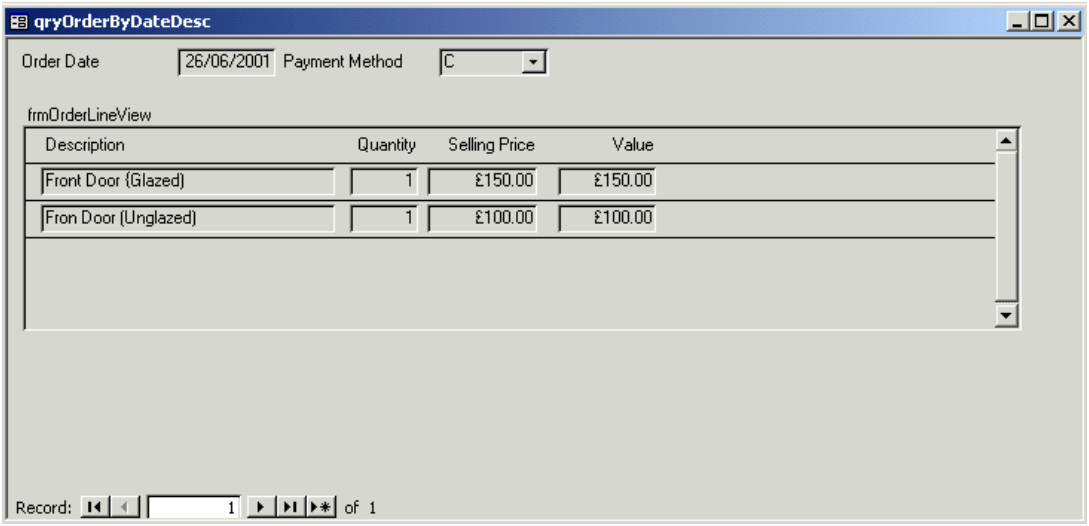
Hide the key fields (Customer_Id, Order_Id)

The form when viewed should look like this:



Place the OrderLineView form as subform onto the OrderView form. Check that the Master and Child links are OK - the forms should be linking on the Order_Id fields.

When viewed, the OrderView form should look like this:



CustomerView Form

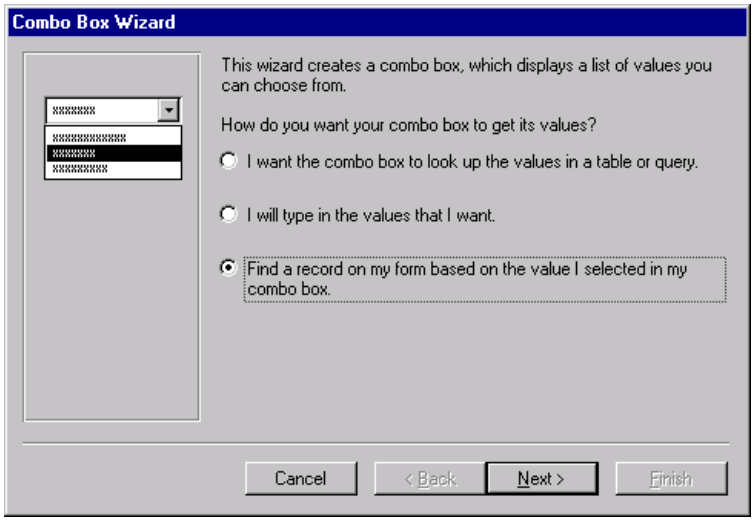
Often it is possible to take an existing form and re-use it for another purpose.
In the database window, copy the frmCustomer form (Ctrl-C) and paste it (Ctrl-V) to a new form called frmCustomerView.

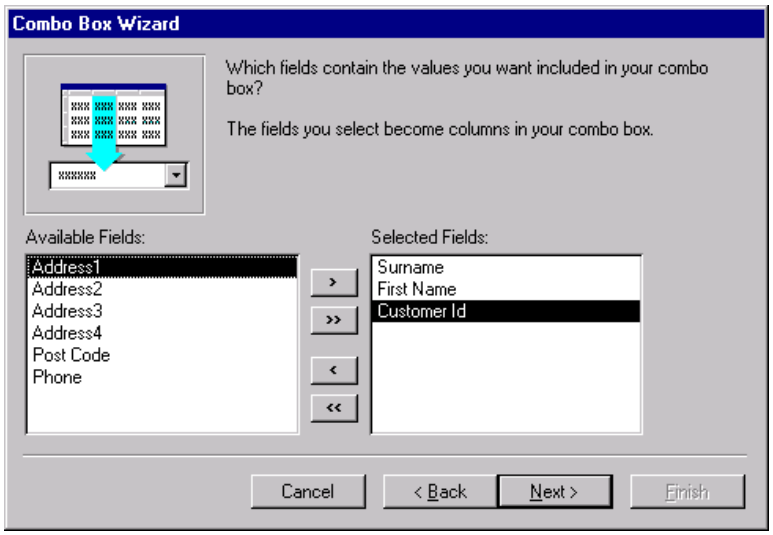
Finding a record to display by using a Combo Box

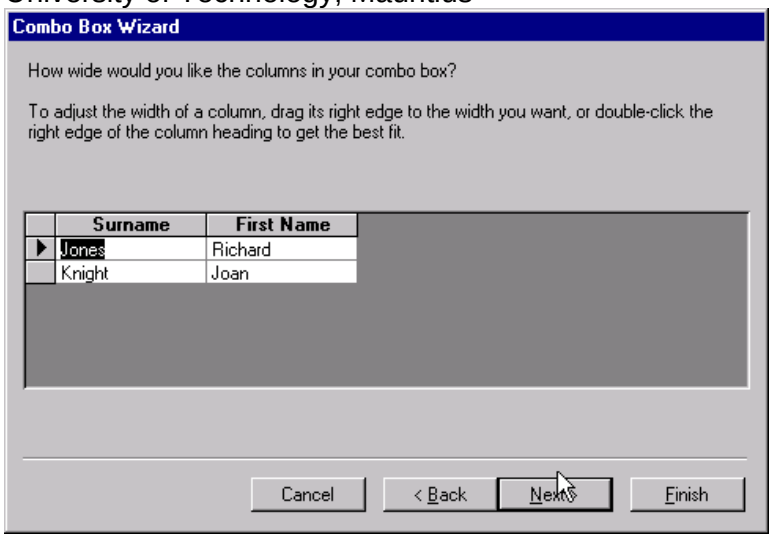
This is the third use of the combo box.

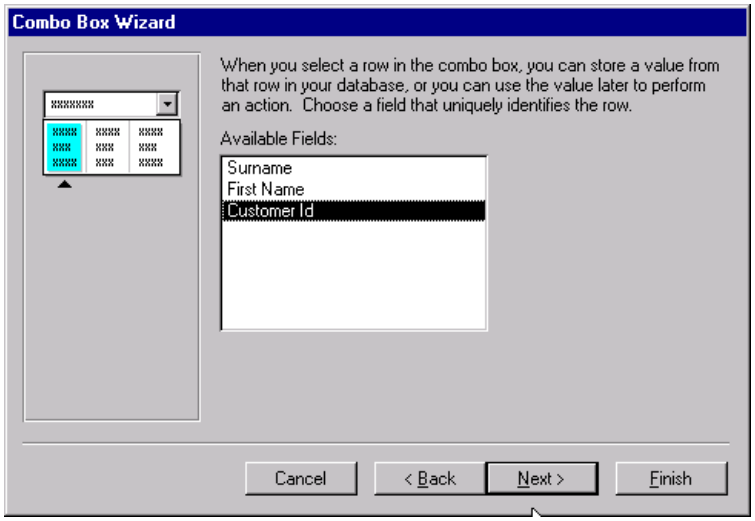
Delete the Customer_Id control on the CustomerView form..

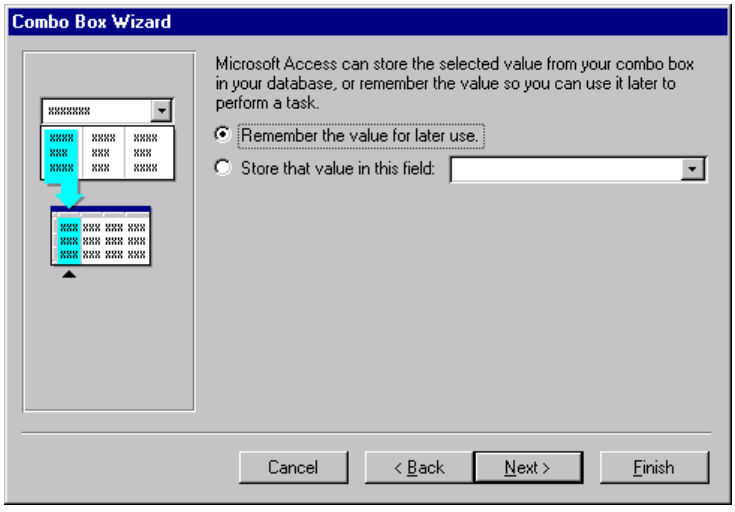
Add a combo box to select a record on the form based on the value selected in the combo box.

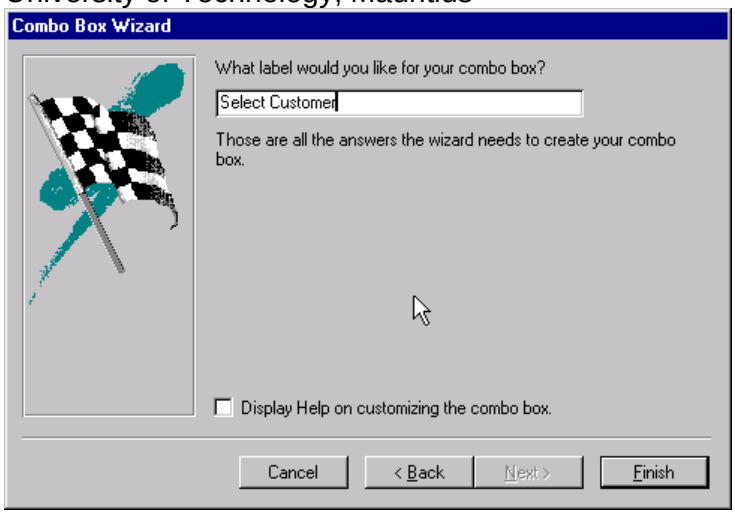





Combo Box Wizard
How wide would you like the columns in your combo box?
To adjust the width of a column, drag its right edge to the width you want, or double-click the right edge of the column heading to get the best fit.
Surname
First Name
Jones
Richard
Knight
Joan
Cancel
< Back
Next >
Finish


Combo Box Wizard
When you select a row in the combo box, you can store a value from that row in your database, or you can use the value later to perform an action. Choose a field that uniquely identifies the row.
Available Fields:
Surname
First Name
Customer Id
Cancel
< Back
Next >
Finish


Combo Box Wizard
Microsoft Access can store the selected value from your combo box in your database, or remember the value so you can use it later to perform a task.
Remember the value for later use.
Store that value in this field:
Cancel
< Back
Next >
Finish



See if you can change the properties of the combo box so that the customers appear in surname descending order.

Take the scroll bars and navigation buttons off the form, and change its properties so that records cannot be added [AllowAdditions – No].

Disable all the controls except for the combo box.

The form should look like this:

Concatenating the FirstName and Surname fields

Instead of having the customer's surname and FirstName appearing in separate text boxes, you can concatenate [join] them together to appear in one text box.

Create a new text box the form and enter the following in the *Control Source* property:
=[FirstName] & " " & [Surname]

Delete the original *Surname* and *FirstName* text boxes.

Now you can place the OrderView form onto the CustomerView form and check that the links are OK. The whole Customer Orders form is now complete.

CHECKPOINT: SDGCheck4