

# Designing and Creating Databases

Terry Marris November 2009

## ***11 Sorting and Indexing***

We see how to sort and index a database, in particular, we see how to

1. apply an index to secondary key fields
2. sort records according to given criteria

### **11.1 Primary Key Fields**

I once had two students with identical names (Bhavin Patel), gender (male) and date of birth (16 May 1985) in the same class. How could I be sure to give the right student the right assignment marks on the mark record sheet? Well, each student had their own student id number assigned by the College on enrolment. So one student was called Bhavin Patel 13579, and the other Bhavin Patel 24680.

A primary key field uniquely identifies a record; no two records have the same primary key.

So far we have used the key field provided by Access and usually named ID as our primary key field. But we can define our own primary key field, as we shall see.

### **11.2 Secondary Key Fields**

A secondary key field provides an index to speed up the searching process. StudentSurname could be a candidate for a secondary key field. Its values need not be unique.

For example, how many people visiting a doctor's surgery can remember their National Health Service (NHS) Number? Each person registered with a GP has his or her own NHS Number and this is used to uniquely identify their record. But a person visiting their surgery (usually) knows their own name and so patient records are indexed on surname. Having found all the records with the surname Smith the receptionist would ask questions such as: *date of birth?* and *address?* to establish which particular Smith is standing before them.

### 11.3 Scenario

ShockAndAwe sells and maintains intruder alarm systems. You are employed to design and maintain a database of customer's payments for servicing intruder alarm systems. You start by sketching out a few example records.

CustomerAccounts

Account Number	Customer Name	PostCode	Address	Current Balance	Transaction		
					Date	Amount	Type
PLA005	Plant	LE1 3WA	19 Painter St	0.00	31 Oct 09	150.00	payment
COP003	Copper	LE1 3WL	13 St John St	-375.00	30 Sep 09	375.00	invoice

When a customer is sent an invoice to pay for servicing completed, their current balance is debited by the transaction amount, the transaction amount is updated and the transaction type is set to invoice. A minus sign in the Current Balance means the customer owes the company money.

When a customer sends a payment, their current balance is credited with the amount paid, the transaction amount is updated and the transaction type is set to payment.

AccountNumber is the primary key field. The account number comprises the first three letters of the customer's name followed by three digits to guarantee uniqueness.

The table is indexed on CustomerName and on PostCode to speed up searches of the database, especially useful when the database becomes huge.

To meet management requests you intend to provide queries sorted on CustomerName, PostCode and money owed by the customer - the queries can then become the basis for reports. You could sort the database table itself - but the sort is not saved. You could sort when creating the reports themselves - but with reports your primary focus is on layout and presenting information. So we choose to create the queries ready sorted as the basis for the reports.

1. start a new, blank database. Choose Design view and name the table tblCustomerAccounts
2. change FieldName ID to AccountNumber, and set Data Type to Text
3. set Field Size = 6, Required = Yes, Allow Zero Length = No, Indexed = Yes (No Duplicates)
4. for CustomerName set Field Size = 25, Required = Yes, Allow Zero Length = No, Indexed = Yes (Duplicates OK)
5. complete the rest of the table, setting the General attributes (Field Size, Required, Allow Zero Length and Indexed as appropriate). But ...

6. the Transaction Type has just two text values: payment and invoice. You could create a lookup list to minimise typing. Choose Lookup Wizard from the Data Type list. Choose *I will type in the values that I need*. Next.
7. supply the list of values you want to use, one per row.

8. click Finish.

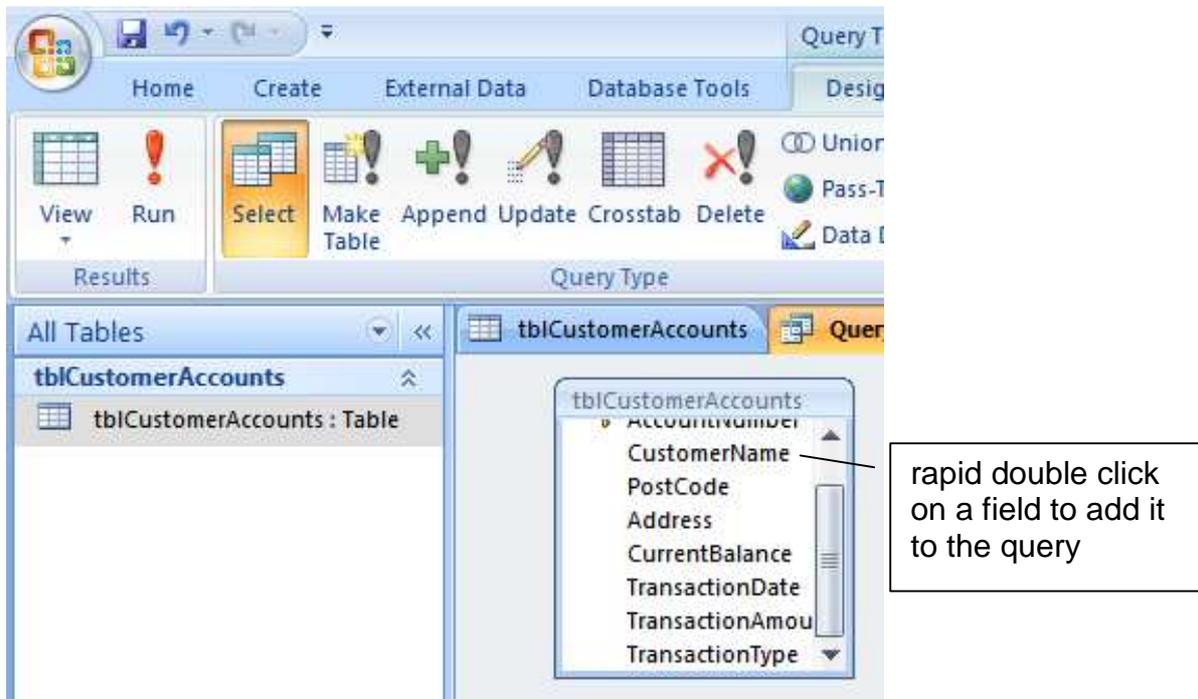
Now, when you come to populate the table (i.e. add data to it) you do not need to type out either payment or invoice in full - you can just type in the first letter i.e. either p or i, and the complete word is written for you. (Or you can just click on the drop down list arrow when you come to the TransactionType column.)

AccountNumber	CustomerName	PostCode	Address	CurrentBalance	TransactionDate	TransactionAmount	TransactionType
BUT001	Button	LE1 3WK	24 Bart St	-£250.00	26-Sep-09	£225.00	invoice
CAS001	Case	LE1 3WI	17 Jude St	£0.00	28-Sep-09	£150.00	payment
COP003	Copper	LE1 3WC	13 Job St	-£375.00	30-Sep-09	£375.00	invoice
GRE003	Greensmith	LE1 3WM	16 Manik St	£0.00	29-Sep-09	£325.00	payment
KIN001	King	LE1 3WV	23 Plant St	-£125.00	25-Sep-09	£225.00	invoice
KIN002	King	LE1 3WO	29 Orlick Way	£0.00	01-Nov-09	£200.00	payment
KIN003	King	LE1 3WM	25 Manik St	£0.00	01-Nov-09	£175.00	payment
MAD003	Maddock	LE1 3WI	17 Jude St	£0.00	30-Sep-09	£300.00	payment
PAT009	Patel	LE1 3WO	27 Orlick Way	-£300.00	25-Sep-09	£350.00	invoice
PLA005	Plant	LE1 3WV	19 Plant St	£0.00	31-Oct-09	£150.00	payment
SAM001	Sampson	LE1 3WK	22 Bart St	£0.00	30-Sep-09	£275.00	payment
TUR002	Turner	LE1 3WC	22 Job St	-£125.00	25-Sep-09	£250.00	invoice
*				£0.00			

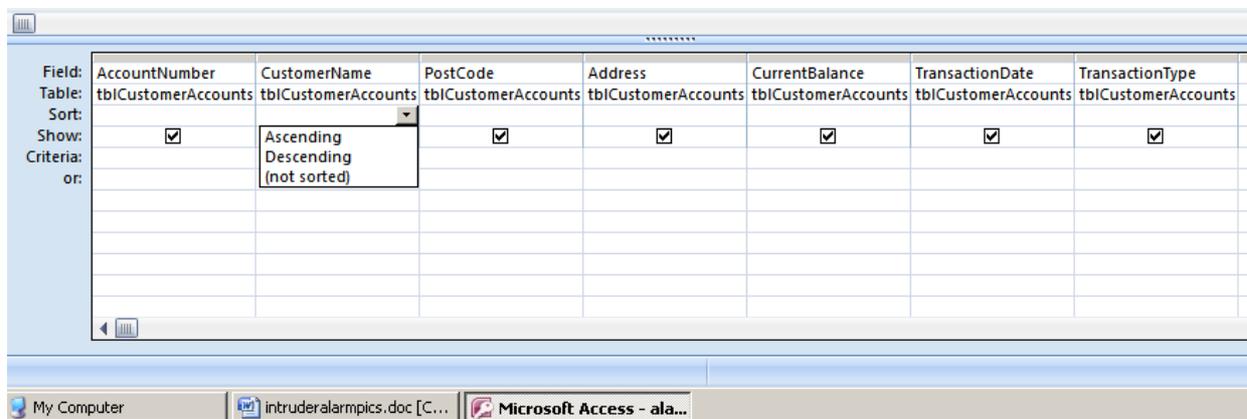
## 11.4 Sorting

We sort the database in alphabetical order by CustomerName, and store the result in a query.

1. choose Create, Query Design
2. select tblCustomerAccounts. Choose Add then Close
3. add each field in turn (you could rapid double click on each field)



4. in the Sort row, in the CustomerName column, click the cell then click the Sort down arrow and choose Ascending



## 5. run the query (! Run command top left of Ribbon)

AccountNumber	CustomerName	PostCode	Address	CurrentBalance	TransactionDate	TransactionType
BUT001	Button	LE1 3WK	24 Bart St	-£250.00	26-Sep-09	invoice
CAS001	Case	LE1 3WI	17 Jude St	£0.00	28-Sep-09	payment
COP003	Copper	LE1 3WC	13 Job St	-£375.00	30-Sep-09	invoice
GRE003	Greensmith	LE1 3WM	16 Manik St	£0.00	29-Sep-09	payment
KIN003	King	LE1 3WM	25 Manik St	£0.00	01-Nov-09	payment
KIN002	King	LE1 3WO	29 Orlick Way	£0.00	01-Nov-09	payment
KIN001	King	LE1 3WV	23 Plant St	-£125.00	25-Sep-09	invoice
MAD003	Maddock	LE1 3WI	17 Jude St	£0.00	30-Sep-09	payment
PAT009	Patel	LE1 3WO	27 Orlick Way	-£300.00	25-Sep-09	invoice
PLA005	Plant	LE1 3WV	19 Plant St	£0.00	31-Oct-09	payment
SAM001	Sampson	LE1 3WK	22 Bart St	£0.00	30-Sep-09	payment
TUR002	Turner	LE1 3WC	22 Job St	-£125.00	25-Sep-09	invoice
				£0.00		

## 6. if the result is ok, right click on the Query1 tab, choose Save. Save the query as qrySortedByName, OK

### Exercise 11.1

1. For the AlarmServicing database shown above, provide two queries:
  - a. sorted in alphabetical order by post code
  - b. sorted by debt, largest first

**We have seen** how to sort and index a database.

**Next, we** look at forms.

### Bibliography

GROVER C, MACDONALD M & VANDER VEER E *Office 2007 The Missing Manual* O'Reilly 2007