Database

I was given a dataset for Bike Sales from 2005 and I had to put it into a database.

The dataset contained the following headings: "Order Number","Title","First Name","Last Name","Address Line 1","Town","County","Post Code","Order Date","Make","Model","Price","Card Type","Card Number","Expiry Date"

I organized these into the following:

- **"Order Number","Title","First Name","Last Name","Address Line 1","Town","County","Post Code","Order Date","Card Type","Card Number","Expiry Date"**

- **"Make","Model","Price"**

- **"Order Number","Title","First Name","Last Name""Make","Model"**

And another containing all of the fields to be used as a reference to who purchased what. This is also used for the relationships as there are duplicate values

Here is how I have set up my tables:

Bikes:

Primary	Name	Type	Validation	Field Size
YES	Make	Text	N/A	50
YES	Model	Text	N/A	50
	Price	Currency	***Validation Rule:*** >99 And <1500 (Above £99 and below £1500) ***Validation Text:*** Please check, (Min £99 Max £1500)	N/A

Customer:

Primary	Name	Type	Validation	Field Size
YES	Order Number	Number	***Input Mask:*** 90000 (4 or 5 digit number)	5
	Title	Text	***Validation Rule:*** "Mr" Or "Mrs" Or "Miss" Or "Ms" Or "Dr" ***Validation Text:*** You must select either Mr, Mrs, Miss, Ms or Dr	4

	First Name	Text		20
	Last Name	Text		20
	Address Line 1	Text		50
	Town	Text		20
	County	Text		25
	Post Code	Text	***Input Mask:*** >LL09\ 0LL	12
	Order Date	Date/Time	***Validation Rule:*** <Date() ***Validation Text:*** Invalid Date!	N/A
	Card Type	Text	***Validation rule:*** "Vista" Or "Mister Card" Or "Socket" Or "Francard" Or "Armenian Express" ***Validation Text:*** We only support following cards: Vista; Mister Card; Socket; Francard and Armenian Express	50
	Card Number	Number	***Input Mask:*** 0000/0000/0000/9999/9999 ***Validation Text:*** Get the card number right!	Long Integer
	Expiry Date	Date/Time	***Validation Rule:*** >=#31/01/2006# (Newer than 31/01/2006) ***Validation Text:*** Your card has expired, please use a different card	N/A

The orders bikes table has the same properties as mentioned in the table above

Some of the field sizes have been reduced to save space, I have shortened the field size, if someone's name is more than 20 characters long, it should still be possible to identify them.

To link the tables together, we use primary keys. These are the primary keys:

- Customer Table: Order Number
- Orders Bikes: Order Number
- Bikes table: Make AND Model

The reason there is 2 primary keys on Bikes Table is that it is impossible to get a unique value on that table using only one key.

Here are my tables as shown in design view:

Bikes : Table (Local)

Field Name	Data Type
Make	Text
Model	Text
Price	Currency
s_GUID	AutoNumber

Customer : Table (Local)

Field Name	Data Type
Order Number	Number
Title	Text
First Name	Text
Last Name	Text
Address Line 1	Text
Town	Text
County	Text
Post Code	Text
Order Date	Date/Time
Card Type	Text
Card Number	Number
Expiry Date	Date/Time
s_GUID	AutoNumber

Orders_bikes : Table (Local)

Field Name	Data Type
Order Number	Text
Title	Text
First Name	Text
Last Name	Text
Make	Text
Model	Text
s_GUID	AutoNumber

I linked these tables together to minimise the risk of deleting data that is linked to other tables:
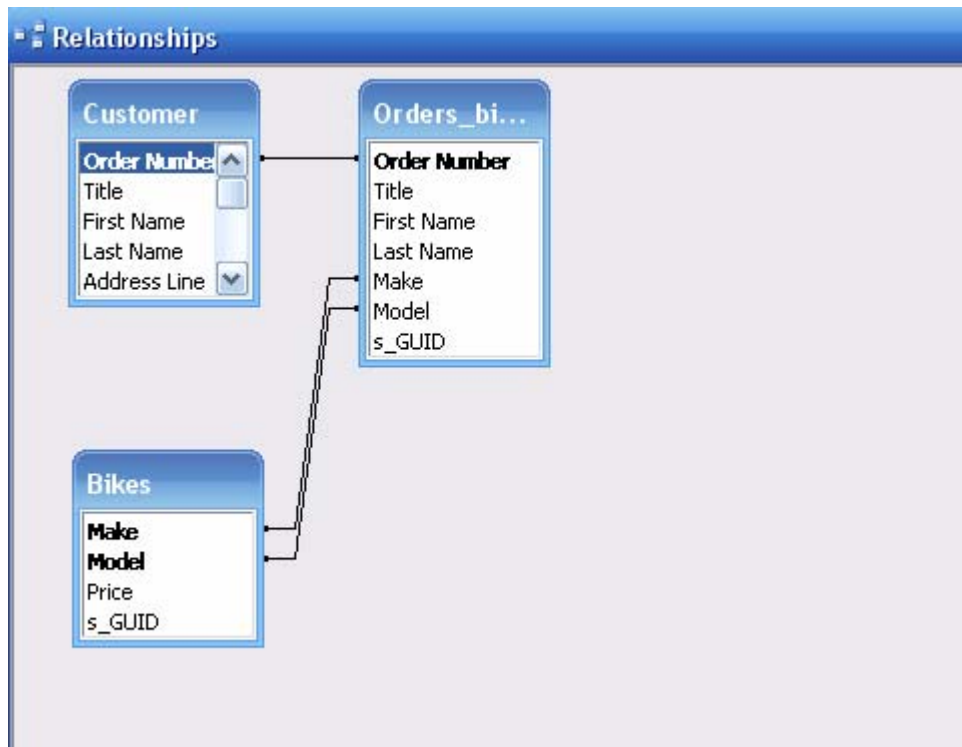
Then I had to import the data. This data was provided in a CSV (Comma Separated variables) text file. A CVS file looks something like this:

"Order Number","Title","First Name","Last Name","Address Line 1","Town","County","Post Code","Order Date","Make","Model","Price","Card Type","Card Number","Expiry Date"
5407,"Mr","Oliver","Orwell","20 Radford Street","Trowmoor","Yorkshire","TR4 5MG",1/1/2005 00:00:00,"Giant","Terrago Disc",£389.98,"VISTA","328903146907",31/5/2007 00:00:00
5408,"Ms","Nicola","Fountain","66 Burnside Ave","Lugeby","Derbyshire","LU5 9JW",1/1/2005 00:00:00,"Giant","XTC 2",£799.00,"Socket","914632896117",31/12/2010 00:00:00
5409,"Mr","Morgan","Eggleton","29 Mychett Street","Colby","Hampshire","CO9 5XQ",1/1/2005 00:00:00,"Tifosi","CK5A 04",£878.98,"Socket","372143398639",31/8/2006 00:00:00
5410,"Ms","Nicole","Upjohn","22 Moor Street","Nailsham","Suffolk","NA1 3IM",1/1/2005 00:00:00,"Giant","Terrago FS2 MTB",£299.99,"VISTA","566466531889",31/3/2008 00:00:00

That was the first 4 entries, as you can see, it is not the easiest to read, but thanks to Access being able to understand the CSV system, it is possible to make it more readable

Order Number	Title	First Name	Last Name	Address Line 1	Town	County	Post Code	Order Date	Make
5407	Mr	Oliver	Orwell	20 Radford Stre	Trowmoor	Yorkshire	TR4 5MG	01/01/2005	Giant
5408	Ms	Nicola	Fountain	66 Burnside Av	Lugeby	Derbyshire	LU5 9JW	01/01/2005	Giant
5409	Mr	Morgan	Eggleton	29 Mychett Stre	Colby	Hampshire	CO9 5XQ	01/01/2005	Tifosi
5410	Ms	Nicole	Upjohn	22 Moor Street	Nailsham	Suffolk	NA1 3IM	01/01/2005	Giant

That is part of the first four again, this time in a neat and tidy table.

These were then transferred to the tables I set up using queries
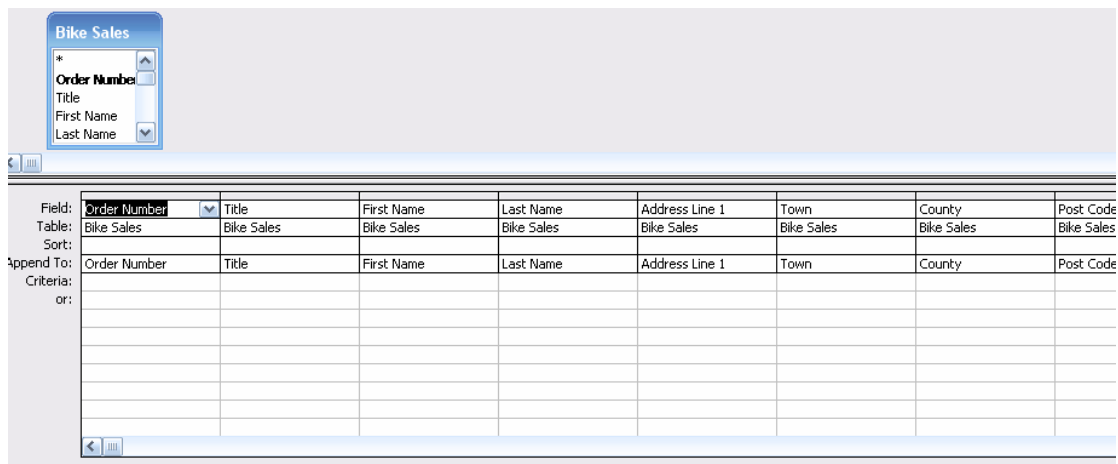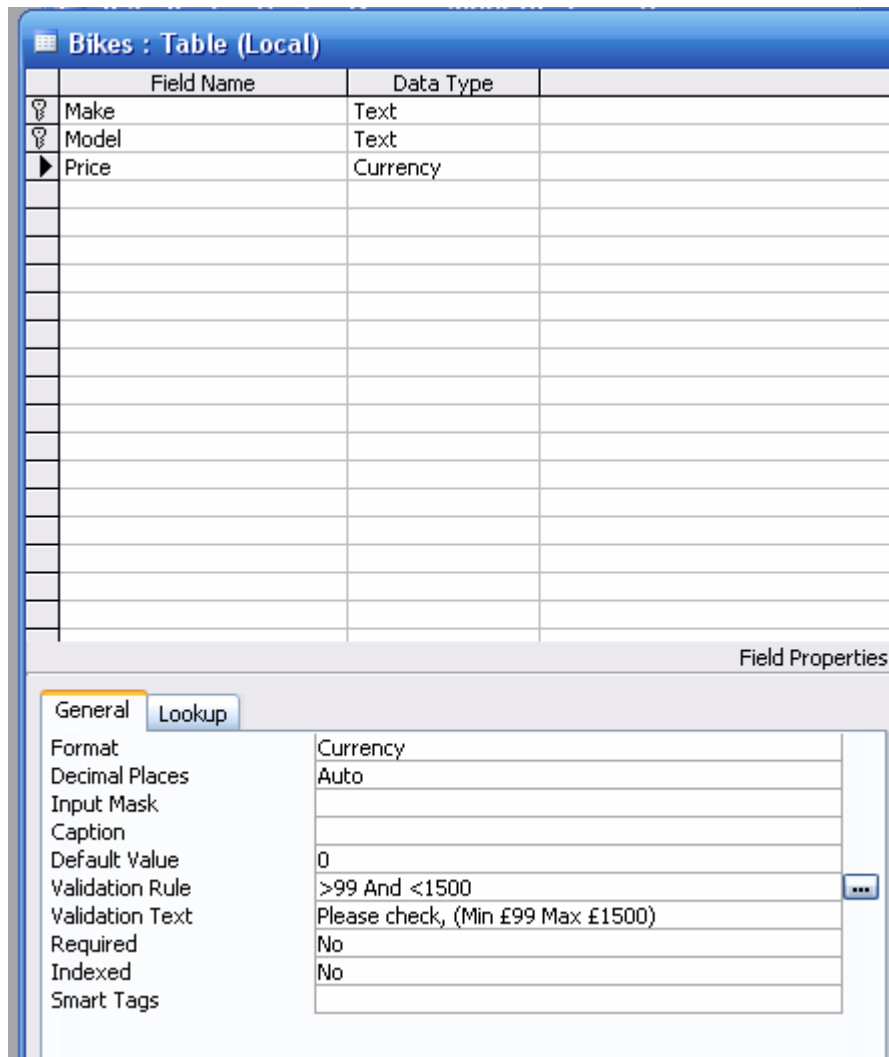

This then, was used to get the data into my tables:
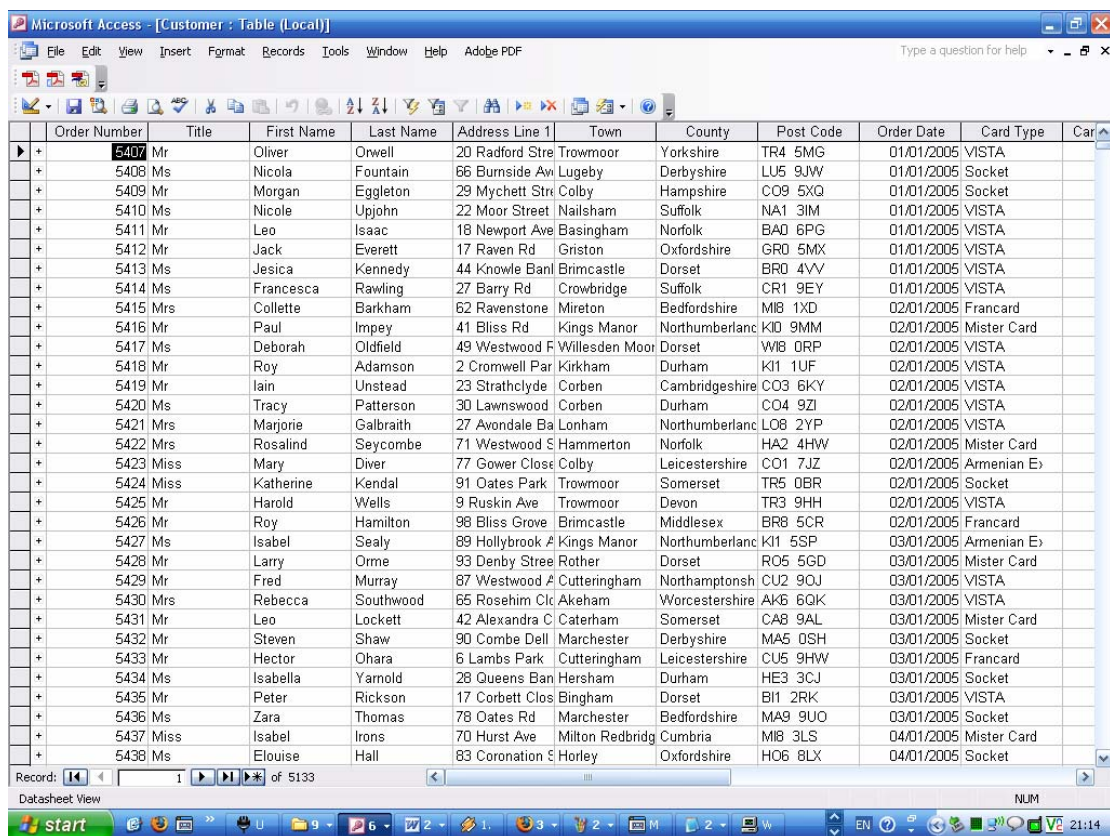
The next thing I did was to set up validation checks, this will help prevent human error when adding data later on:

The first thing I added was a range check on the price:

The ">99 and <1500" means that the end user can not add a value that is either below £99 or above £1500

For the "title" section (eg where you put Mr or Mrs etc.) life has been made simpler for the end user as this has been set up to be a drop down box, and will allow them to just click the one that's applicable. A validation rule has also been set up in case the user decides to type the title in manually.
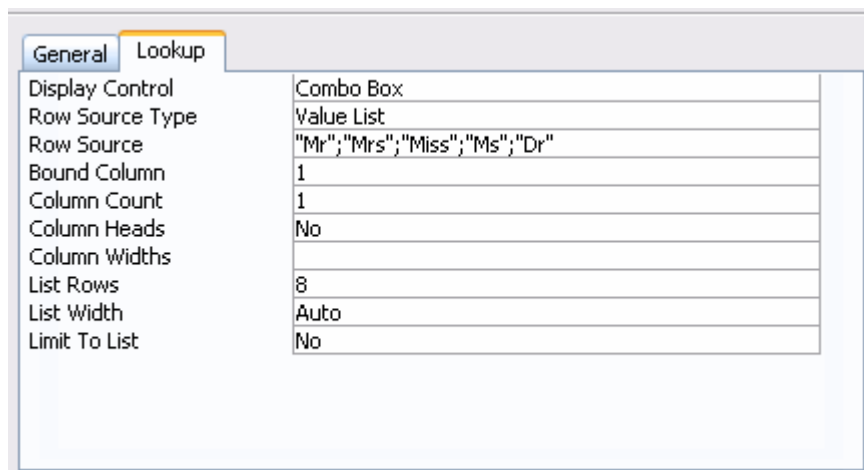
The same thing was done with the "card type" option

On some entries, it was possible to set up an input mask, that allow real time checking on the data that is being entered, the typing will stop if the wrong data is being attempted to be entered into it.

Here is the code explained:

>	Everything is typed in uppercase
L	A letter is REQUIRED
0	A number is REQUIRED
9	A number is OPTIONAL

Here are some more codes that I have not used:

|	
?	An OPTIONAL letter
#	A number or a space
A	A letter or number is REQUIRED
a	A letter or number is OPTIONAL
C	Any character or space is OPTIONAL
<	Everything is typed in lowercase

Some data is optional, but some can NOT be done without (e.g. card numbers) to stop the entry being added without the vital information, this is very easy to achieve, all that you have to do is enable the "required" flag

|	
Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	0000/0000/0000/9999/9999
Caption	
Default Value	0
Validation Rule	
Validation Text	Get the card number right!
Required	Yes
Indexed	No
Smart Tags	

General   Lookup

I did various tests on the database to make sure it worked, here is what happens
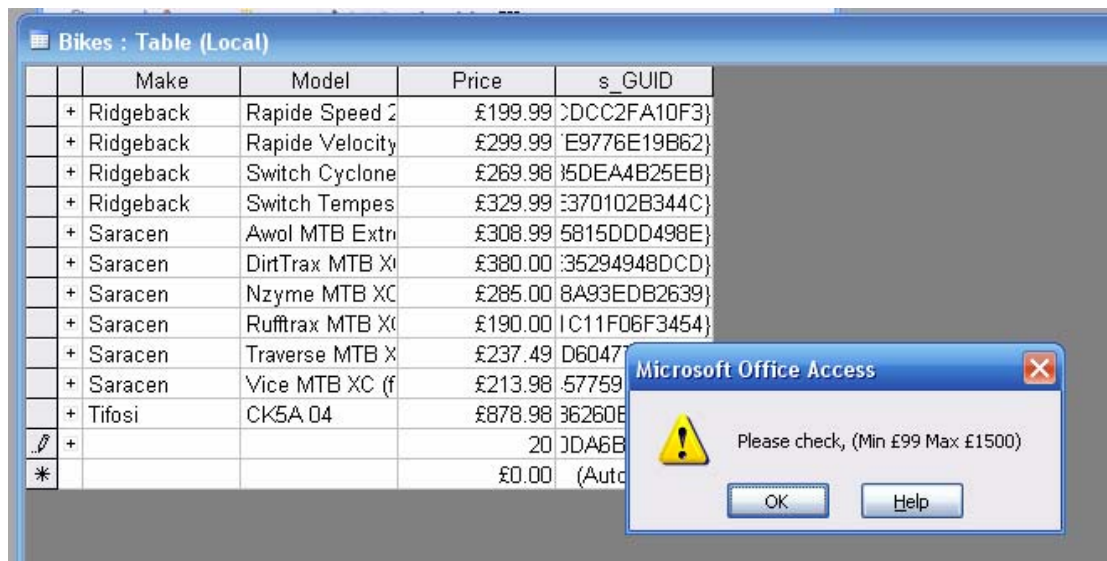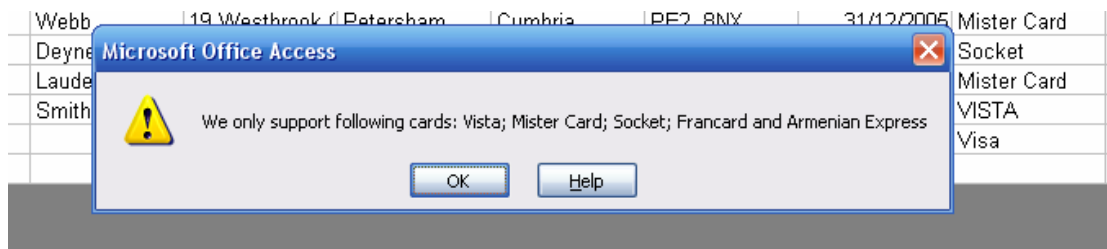

This message appeared when I put in a value lower than £99, I also got the same message when I put something in minus number.
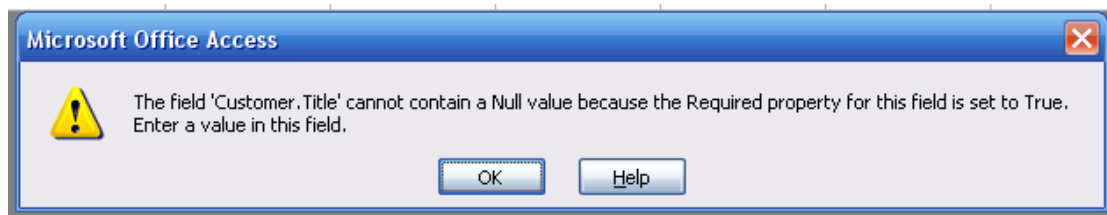
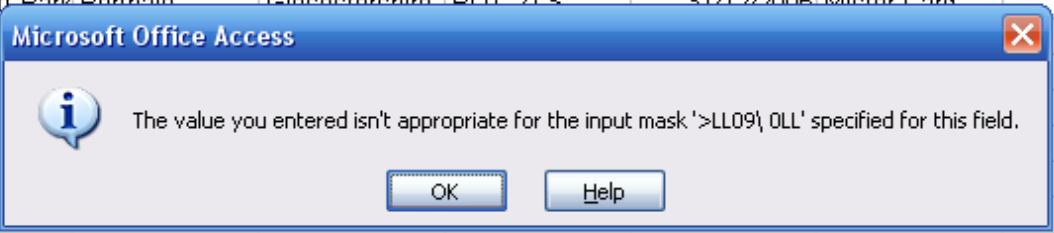If there is nothing typed into the Title value, then this message will appear
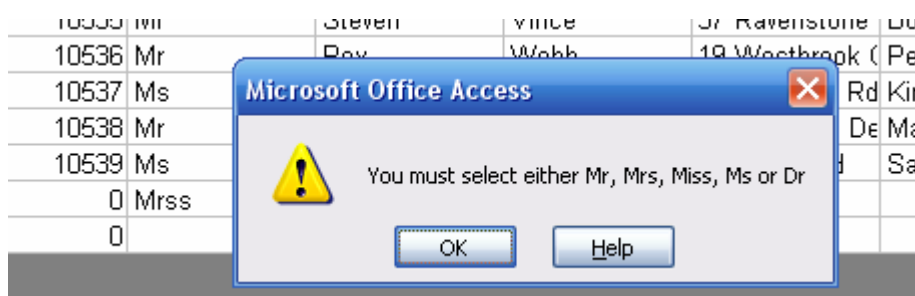
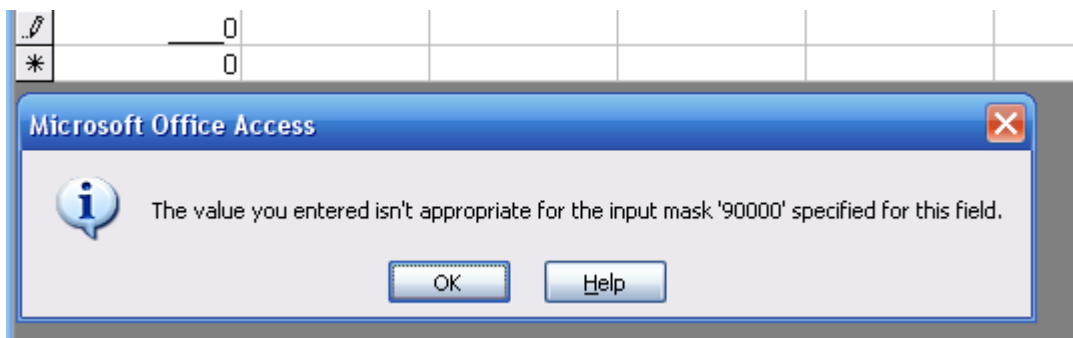If the wrong value into the Title section, you will be greeted with this message
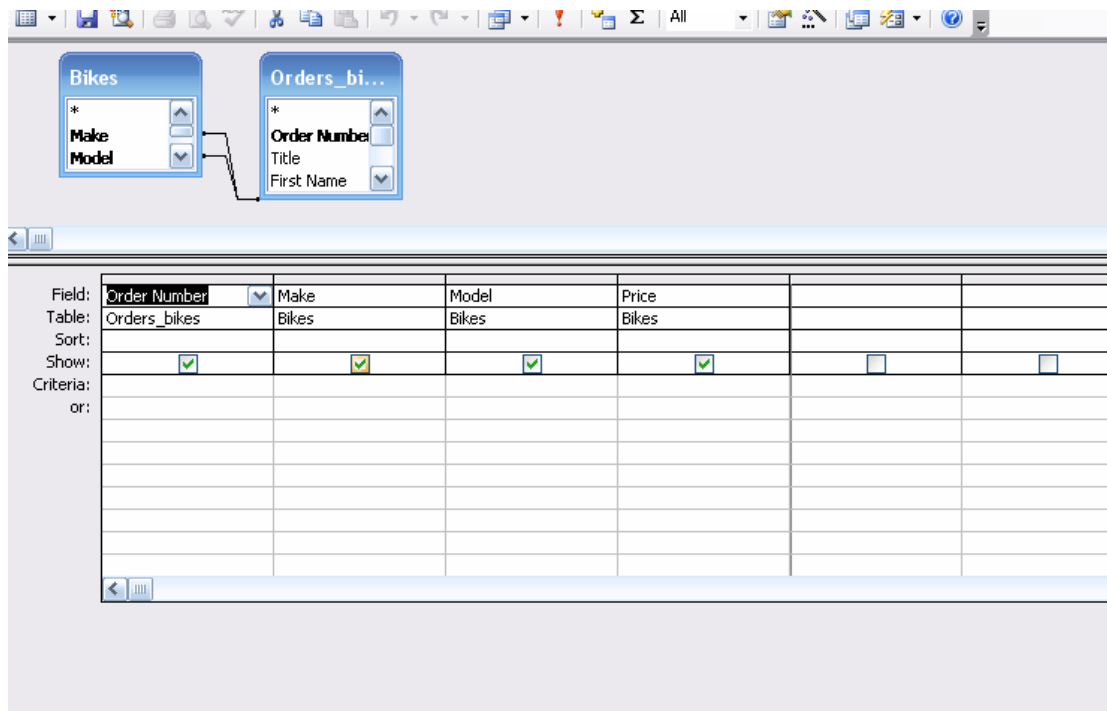

Same if your card is not supported

Microsoft Office Access

The value you entered isn't appropriate for the input mask '>LL09\ 0LL' specified for this field.

OK   Help

ET__ ___   Vista

If you are unable to type something into the box, leave it unfinished and a box will appear telling us what the syntax should be!

___0
0

Microsoft Office Access

The value you entered isn't appropriate for the input mask '90000' specified for this field.

OK   Help

This is the query to find a particular order:

To make it easier, I made a form, lets say we wanted to find order number 10213:

This gives a nice finishing touch to the database.