Database Access
Tables and Queries

By

Nada Khaleel

Database access – Objects

MS Access uses “objects" to help the user list and organize information and prepare specially designed reports. Access offers you Tables, Queries, Forms, Reports, Macros, and Modules when you create a database. Databases in Access are composed of many objects but the following are the major objects:

- Tables
- Queries
- Forms
- Reports

Together, these objects allow you to enter, store, analyze, and compile your data. Here is a summary of the major objects in an Access database;

1. Table

A table is an object that is used to define and store data. When you create a new table,

Access asks you to define fields which are also known as column headings

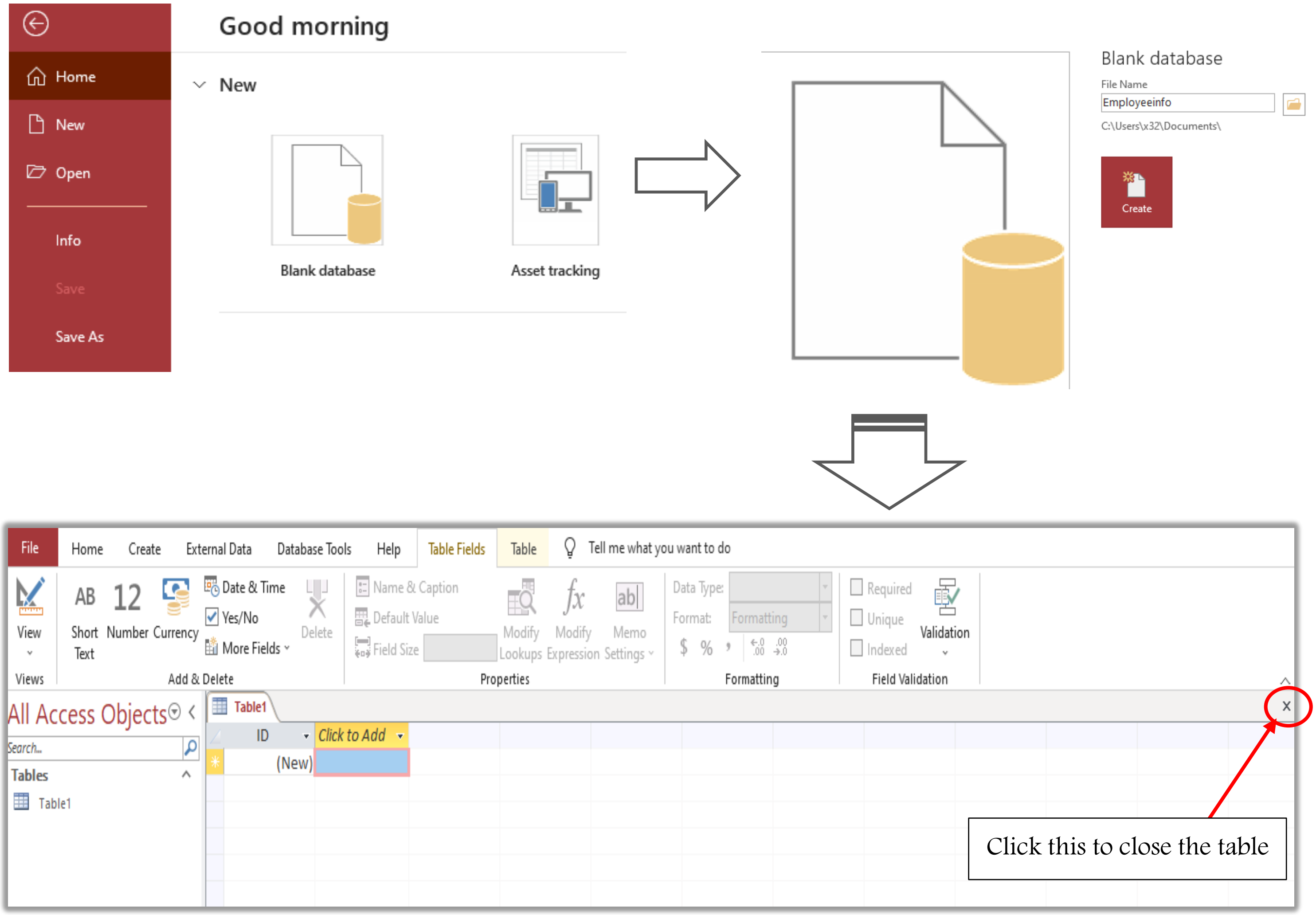
- Each field must have a unique name, and data type.
- Tables contain fields or columns that store different kinds of data, such as a name or an address, and records or rows that collect all the information about a particular instance of the subject, such as all the information about a customer or employee, etc.
- You can define a primary key, one or more fields that have a unique value for each record, and one or more indexes on each table to help retrieve your data more quickly.

***To create a Table, you need to have a database. So, to create a database follow these steps:**

1- Open the Microsoft office database

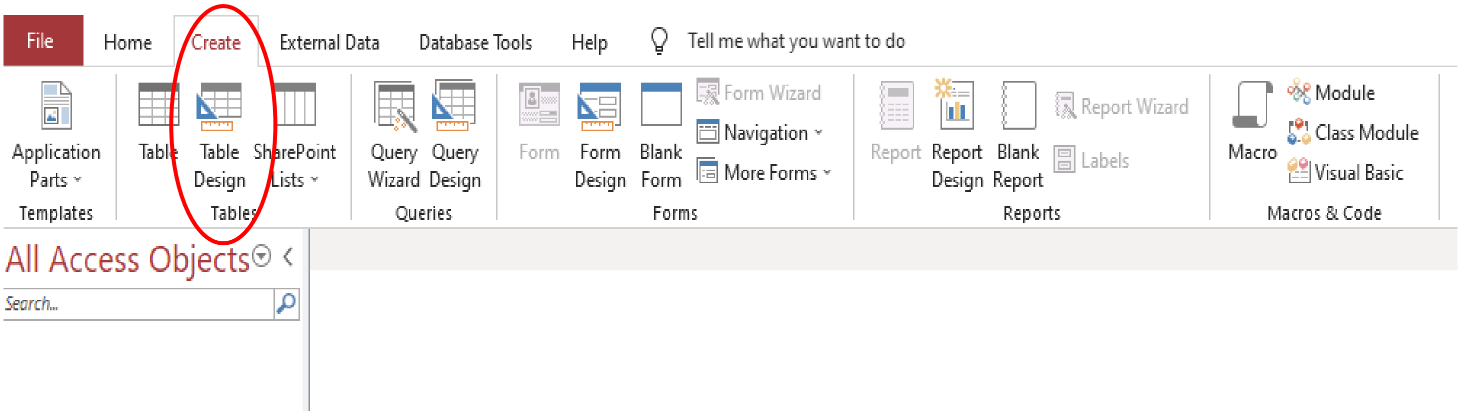
2- From the home tap click on the Blank database

3- On the file name field type name for your database then click Create button

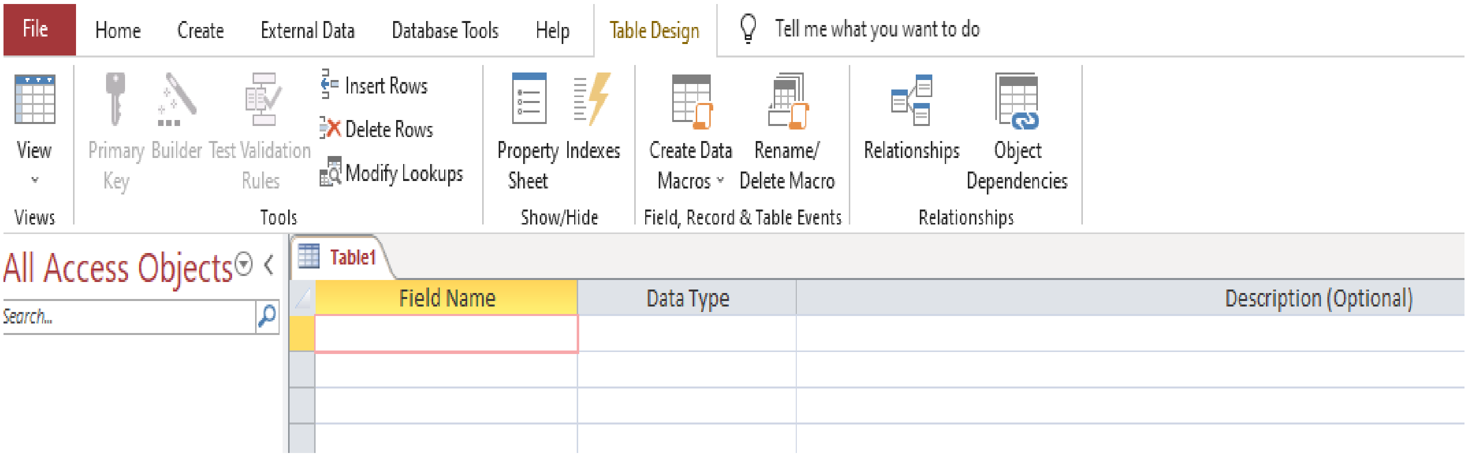


To create a table, follow these steps:

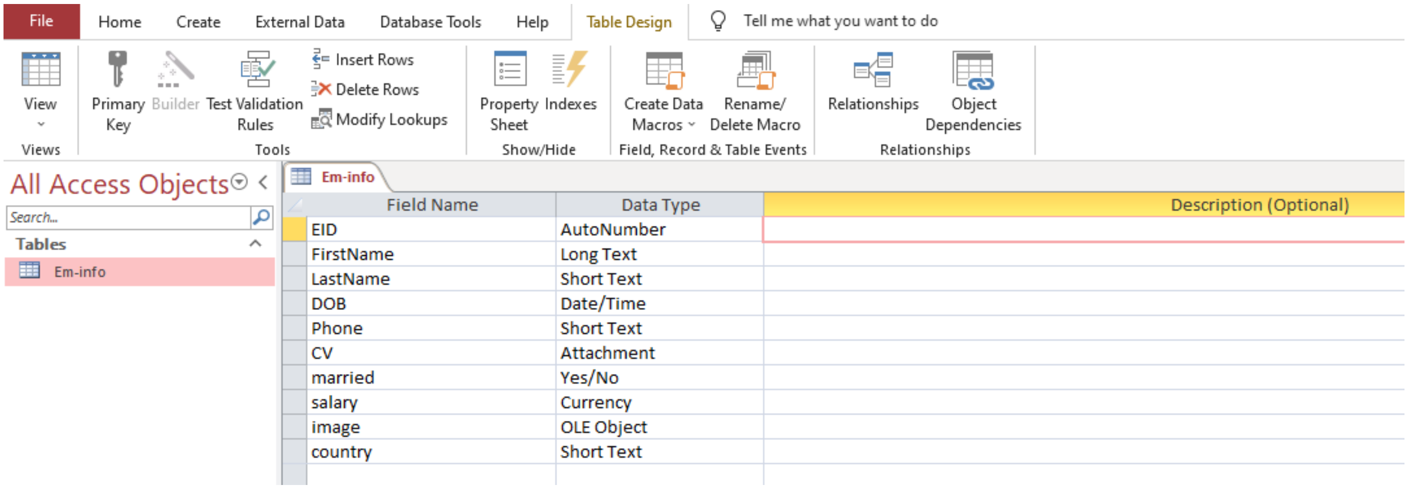
1. from the tool par choose to Create tap, then click on the Table Design



2. Then we should name fields of the table from the Field Name with the type of each field from the Data Type, with a description for the field and this is optional



3. Enter the following info into a table:

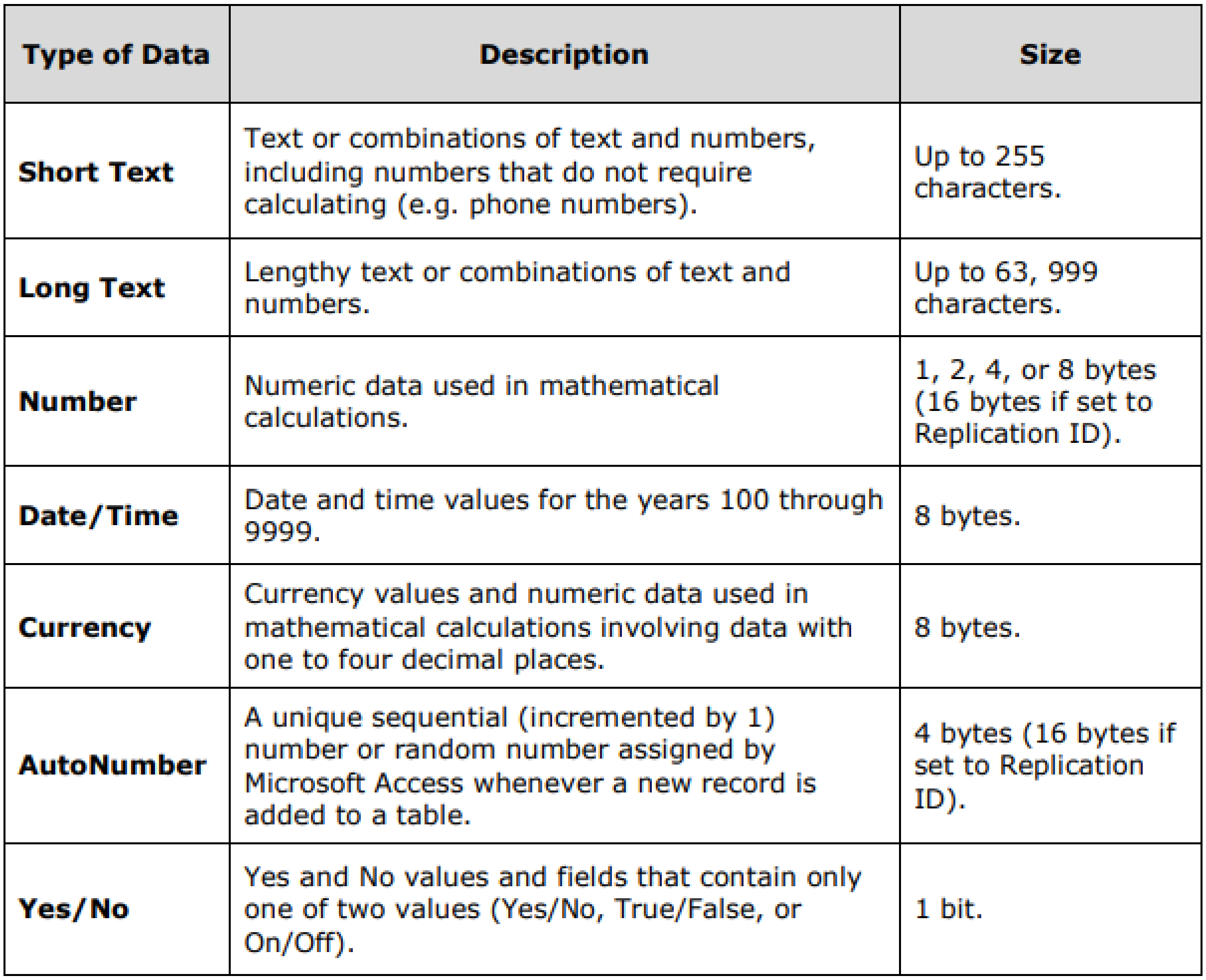


Field Name	Data Type	Description (Optional)
EID	AutoNumber	
FirstName	Long Text	
LastName	Short Text	
DOB	Date/Time	
Phone	Short Text	
CV	Attachment	
married	Yes/No	
salary	Currency	
image	OLE Object	
country	Short Text	

4. Every field in a table has properties and these properties define the field's characteristics and behavior. The most important property of a field is its data type. A field's data type determines what kind of data it can store. MS Access supports different types of data, each with a specific purpose.

- The data type determines the kind of values that users can store in any given field.
- Each field can store data consisting of only a single data type.

Here are some of the most common data types you will find used in a typical Microsoft Access database.



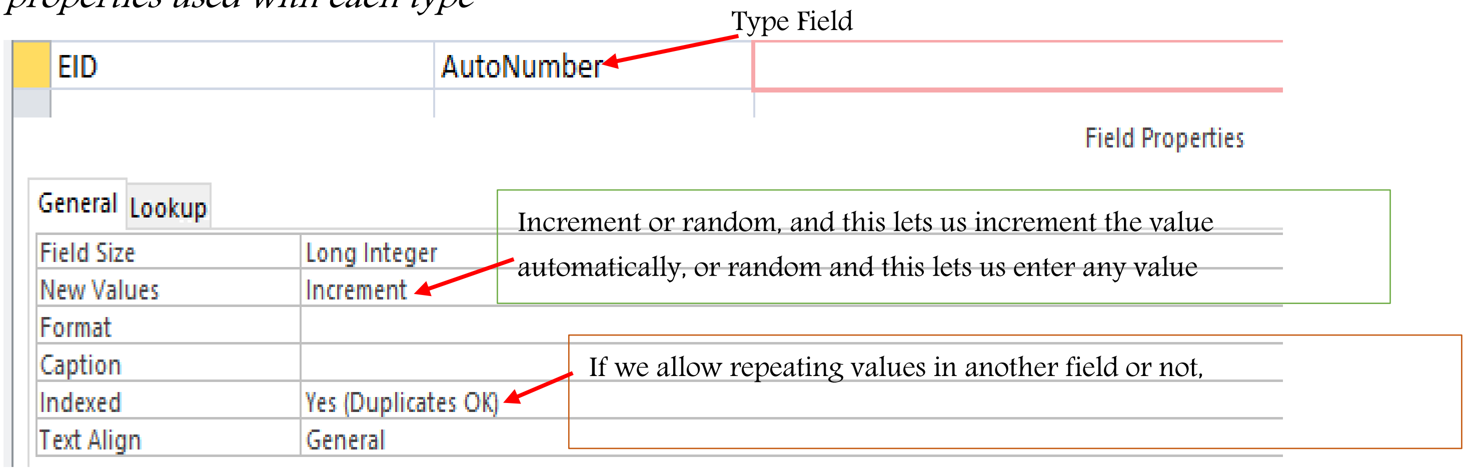
Type of Data	Description	Size
Short Text	Text or combinations of text and numbers, including numbers that do not require calculating (e.g. phone numbers).	Up to 255 characters.
Long Text	Lengthy text or combinations of text and numbers.	Up to 63, 999 characters.
Number	Numeric data used in mathematical calculations.	1, 2, 4, or 8 bytes (16 bytes if set to Replication ID).
Date/Time	Date and time values for the years 100 through 9999.	8 bytes.
Currency	Currency values and numeric data used in mathematical calculations involving data with one to four decimal places.	8 bytes.
AutoNumber	A unique sequential (incremented by 1) number or random number assigned by Microsoft Access whenever a new record is added to a table.	4 bytes (16 bytes if set to Replication ID).
Yes/No	Yes and No values and fields that contain only one of two values (Yes/No, True/False, or On/Off).	1 bit.

- If you use previous versions of Access, you will notice a difference for two of those data types.
- In Access 2013 has two data types — short text and long text. In previous versions of Access, these data types were called text and memo.
- The text field is referred to as short text and your memo field is now called long text.

Here are some of the other more specialized data types, you can choose from in Access:

Data Types	Description	Size
Attachment	Files, such as digital photos. Multiple files can be attached per record. This data type is not available in earlier versions of Access.	Up to about 2 GB.
OLE objects	OLE objects can store pictures, audio, video, or other BLOBs (Binary Large Objects)	Up to about 2 GB.
Hyperlink	Text or combinations of text and numbers stored as text and used as a hyperlink address.	Up to 8,192 (each part of a Hyperlink data type can contain up to 2048 characters).
Lookup Wizard	The Lookup Wizard entry in the Data Type column in the Design view is not actually a data type. When you choose this entry, a wizard starts to help you define either a simple or complex lookup field. A simple lookup field uses the contents of another table or a value list to validate the contents of a single value per row. A complex lookup field allows you to store multiple values of the same data type in each row.	Dependent on the data type of the lookup field.
Calculated	You can create an expression that uses data from one or more fields. You can designate different result data types from the expression.	You can create an expression that uses data from one or more fields. You can designate different result data types from the expression.

***Each type has different properties in the Field Properties, let's see the more properties used with each type*



Type Field

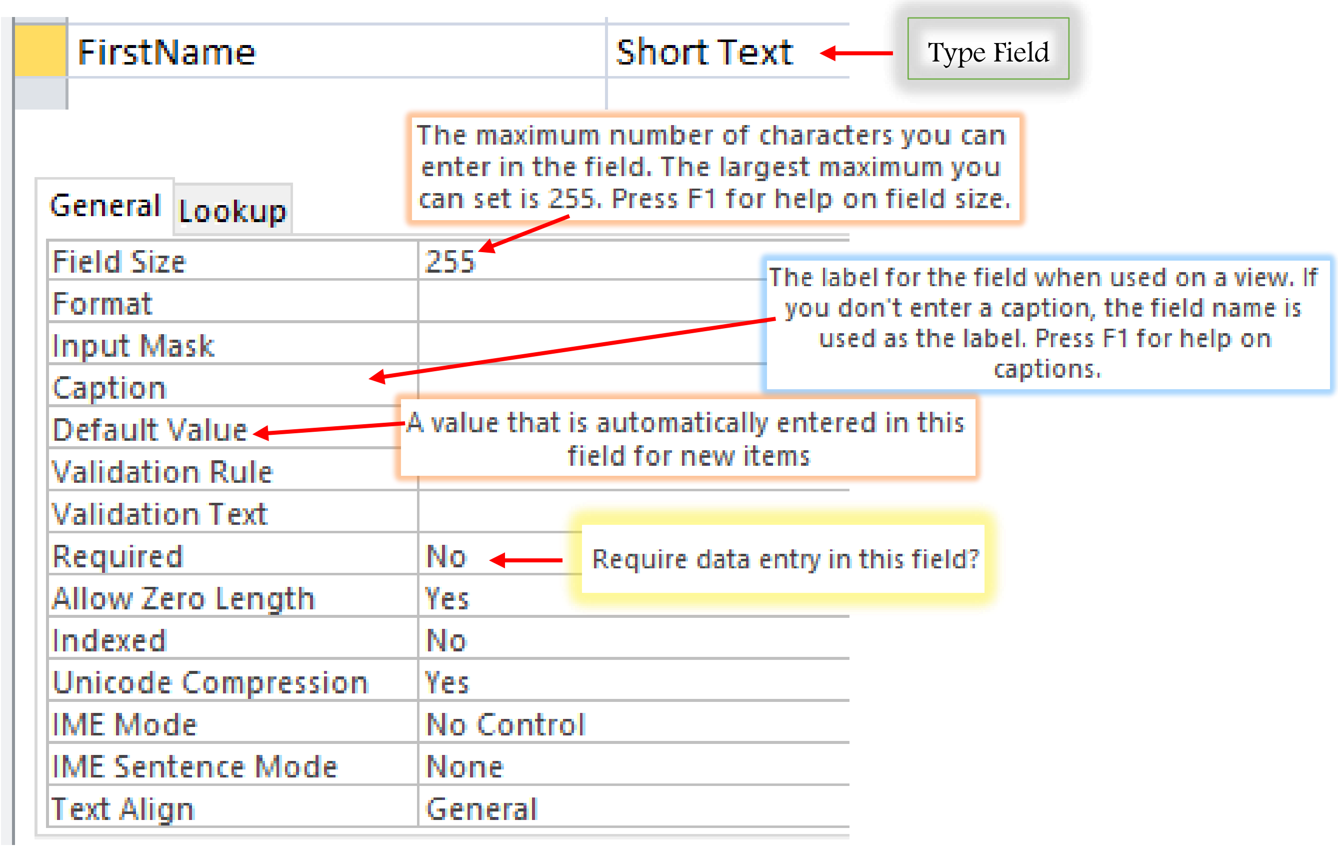
EID	AutoNumber

Field Properties

Field Size	Long Integer
New Values	Increment
Format	
Caption	
Indexed	Yes (Duplicates OK)
Text Align	General

Increment or random, and this lets us increment the value automatically, or random and this lets us enter any value

If we allow repeating values in another field or not,



FirstName	Short Text

Type Field

General | Lookup

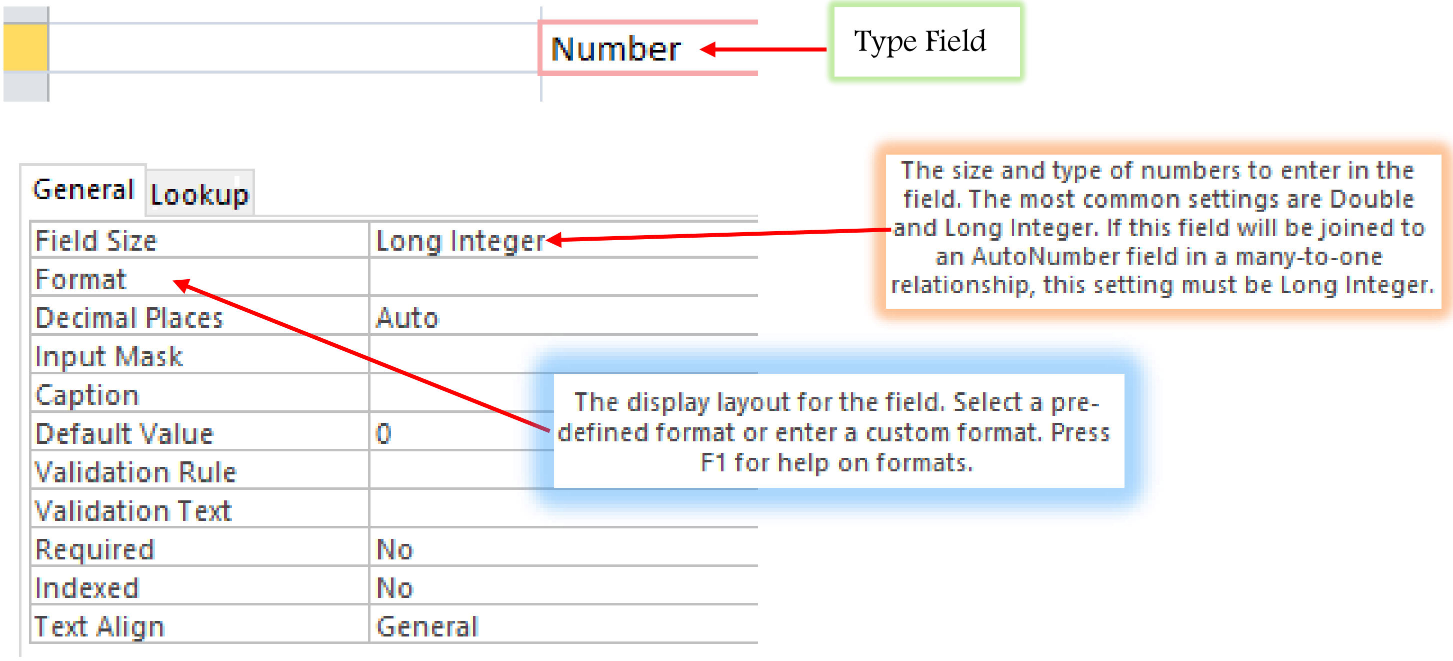
Field Size	255
Format	
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	No
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Text Align	General

The maximum number of characters you can enter in the field. The largest maximum you can set is 255. Press F1 for help on field size.

The label for the field when used on a view. If you don't enter a caption, the field name is used as the label. Press F1 for help on captions.

A value that is automatically entered in this field for new items

Require data entry in this field?



	Number

Type Field

General | Lookup

Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	0
Validation Rule	
Validation Text	
Required	No
Indexed	No
Text Align	General

The size and type of numbers to enter in the field. The most common settings are Double and Long Integer. If this field will be joined to an AutoNumber field in a many-to-one relationship, this setting must be Long Integer.

The display layout for the field. Select a pre-defined format or enter a custom format. Press F1 for help on formats.

Using the Input Mask Wizard

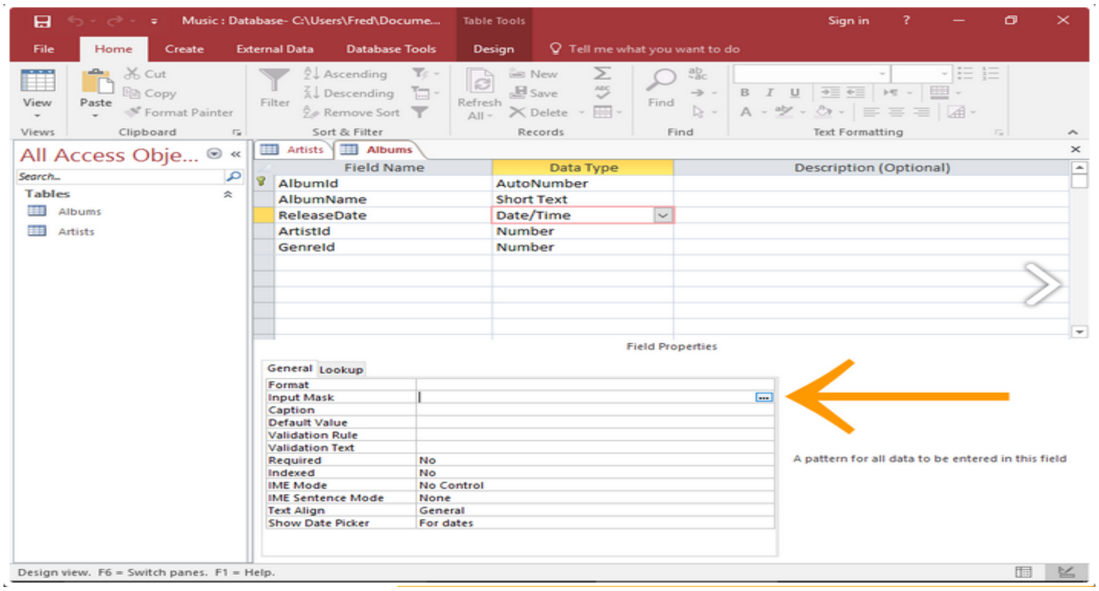
We're now going to apply an input mask to the **Birth Date** field of our Albums table. Our input mask will ensure that users enter each table's birth date in the correct format.

First step

1- Click on the **Birth Date** field then look to the bottom frame. The bottom frame displays the properties of this field.

2- Click somewhere in the **Input Mask** row in the bottom frame row. A small button with three dots will appear. This button launches the Input Mask Wizard.

3- Click that little **[...]** button with the three dots.



Artists | Albums

Field Name	Data Type	Description (Optional)
AlbumId	AutoNumber	
AlbumName	Short Text	
ReleaseDate	Date/Time	
ArtistId	Number	
GenreId	Number	

Field Properties

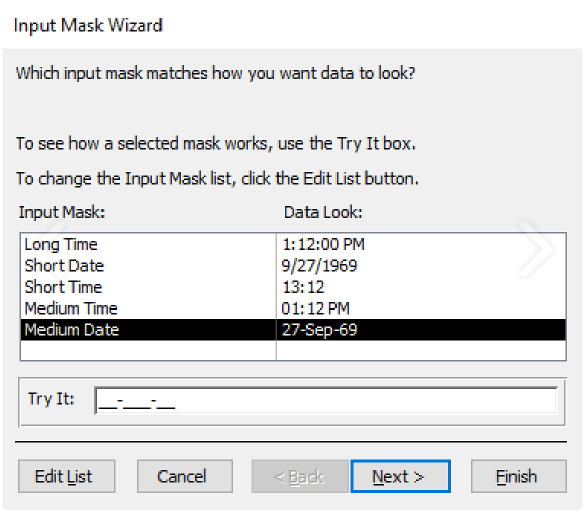
General | Lookup

Format	
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	No
IME Mode	No Control
IME Sentence Mode	None
Text Align	General
Show Date Picker	For dates

A pattern for all data to be entered in this field

Second step

The Input Wizard Mask presents you with some options for how you'd like the data to be entered. You can select any option and test it in the **Try It** field at the bottom of the wizard. Select Medium Date and click Next.



Which input mask matches how you want data to look?

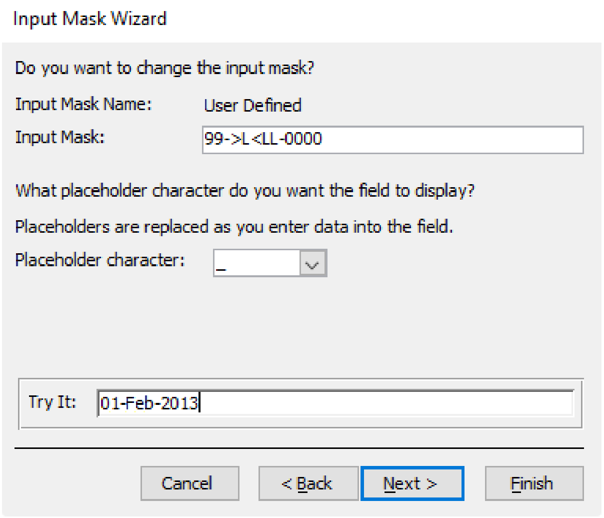
To see how a selected mask works, use the Try It box.

To change the Input Mask list, click the Edit List button.

Input Mask:	Data Look:
Long Time	1:12:00 PM
Short Date	9/27/1969
Short Time	13:12
Medium Time	01:12 PM
Medium Date	27-Sep-69

Try It: __-___-__

Third step



Input Mask Wizard

Do you want to change the input mask?

Input Mask Name: User Defined

Input Mask: 99->L<LL-0000

What placeholder character do you want the field to display?

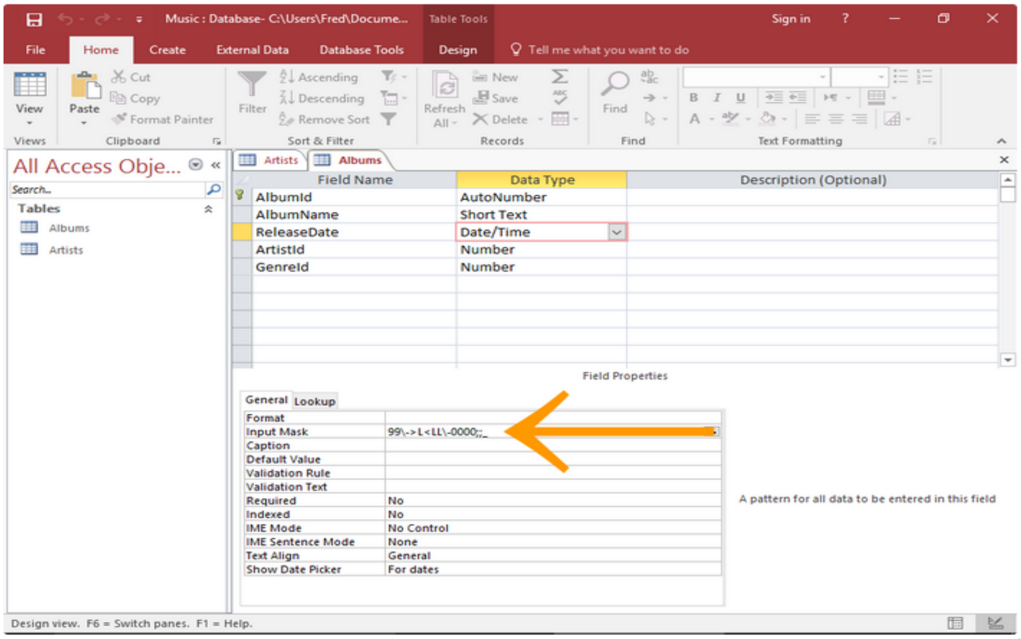
Placeholders are replaced as you enter data into the field.

Placeholder character: _

Try It: 01-Feb-2013

This screen allows you to make adjustments to the input mask. You can test any adjustments in the **Try It** field is at the bottom of the wizard.
We'll make one minor adjustment. Add two zeros, so that the input mask becomes: **99->L<LL-0000**
This will ensure the user enters the date as **DD-MMM-YYYY.**
For example, 20-Mar-2016. Once you're happy with the input mask, click Next.

At the end step, we should click finish such as this Screen to begin entering data in the birth date field.



Field Name	Data Type	Description (Optional)
AlbumId	AutoNumber	
AlbumName	Short Text	
ReleaseDate	Date/Time	
ArtistId	Number	
GenreId	Number	

Field Properties

General Lookup

Format	
Input Mask	99\->L<LL\-0000;;_
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	No
IME Mode	No Control
IME Sentence Mode	None
Text Align	General
Show Date Picker	For dates

A pattern for all data to be entered in this field

Design view. F6 = Switch panes. F1 = Help.

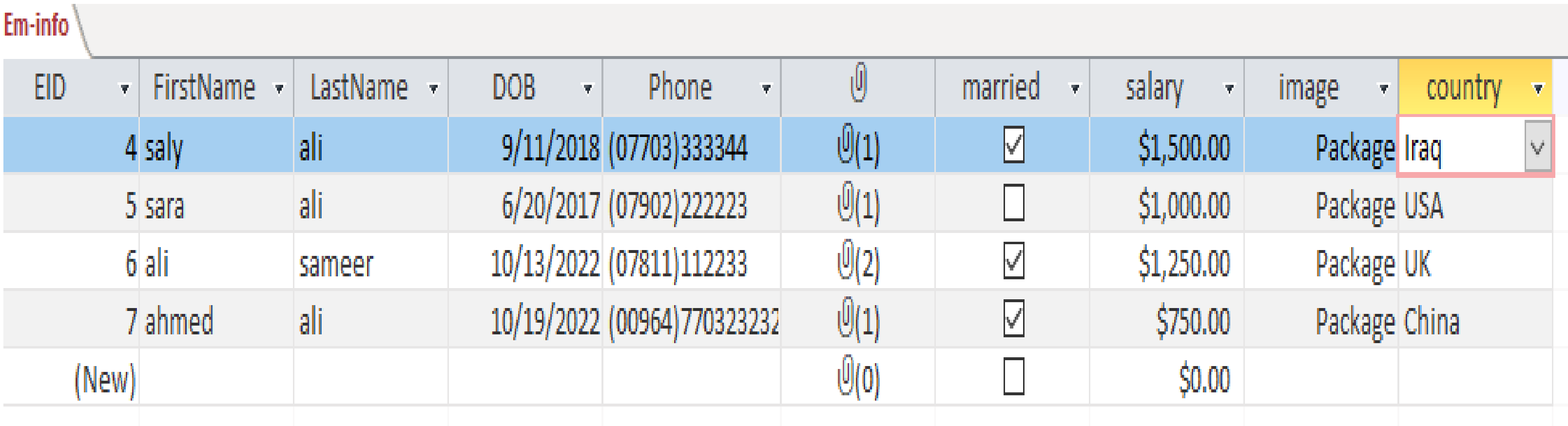
We just created an input mask that looks like this: **99->L<LL-0000**

This specifies how the date should be entered. Here's a breakdown of each part:

- The 99 part defines two optional numbers (i.e. for the day). So, the user has a choice of entering that part of the date. But if they do, it must be a number.

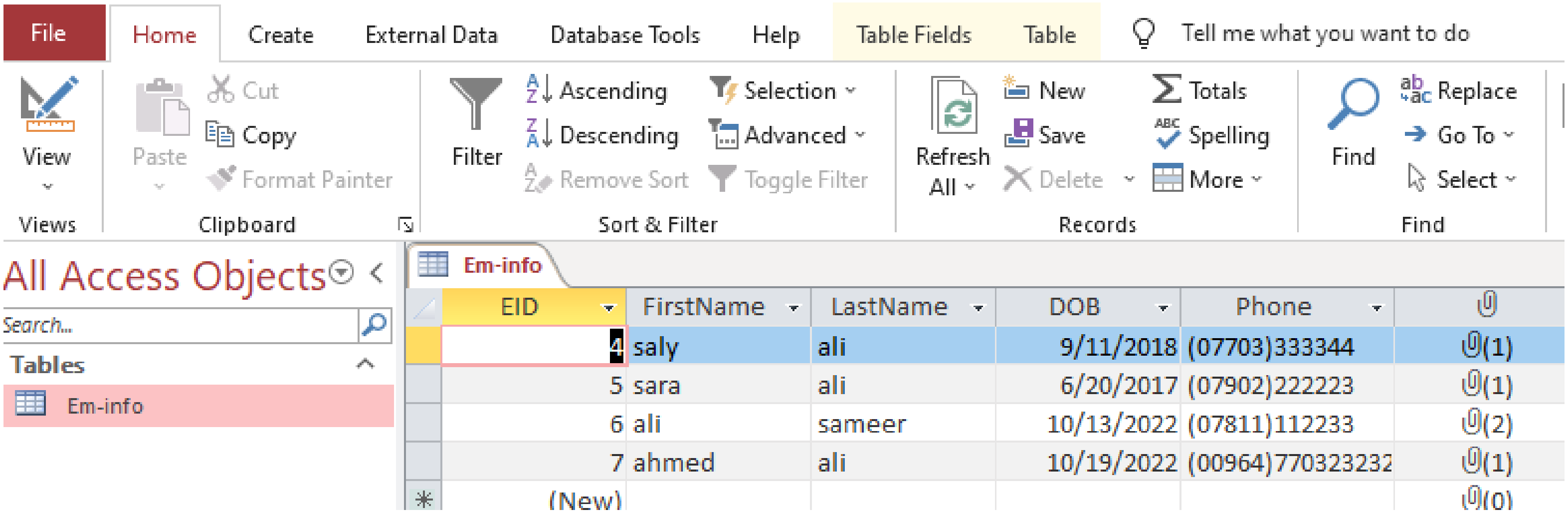
- The >L specifies that the user must enter an uppercase letter for that part of the date.
- The <LL specifies two lowercase letters for that part of the date.
- The 0000 specifies four numbers for that part of the date.

lookup wizard :



EID	FirstName	LastName	DOB	Phone	📎	married	salary	image	country
4	saly	ali	9/11/2018	(07703)333344	📎(1)	☑	$1,500.00	Package	Iraq
5	sara	ali	6/20/2017	(07902)222223	📎(1)	□	$1,000.00	Package	USA
6	ali	sameer	10/13/2022	(07811)112233	📎(2)	☑	$1,250.00	Package	UK
7	ahmed	ali	10/19/2022	(00964)770323232	📎(1)	☑	$750.00	Package	China
(New)					📎(0)	□	$0.00		

**Note that the country field type is "lookup wizard, but shown as "short text"*



EID	FirstName	LastName	DOB	Phone	📎
4	saly	ali	9/11/2018	(07703)333344	📎(1)
5	sara	ali	6/20/2017	(07902)222223	📎(1)
6	ali	sameer	10/13/2022	(07811)112233	📎(2)
7	ahmed	ali	10/19/2022	(00964)770323232	📎(1)
(New)					📎(0)

From the "Home" tape, we can find several options that let us do some operations, as shown in the table below.

Tools	Description
Filter, Selection	take a database value and return a new database value that exposes only data that satisfy a predicate.
Toggle filter	The Toggle Filter button indicates the state of the Filter and Filter On properties. The button remains disabled until there is a filter to apply.
Ascending and Descending	Using to order data depending on the column choice
New, Delete, Save, Refresh	Used to add a new row or delete a row after choosing it, then we should save the change of data and can refresh it
Totals	To compute the total for fields of type Number
More	Several options for the layout of the record
Find, Replace	Use to search for some data, and the “Replace” to replace data

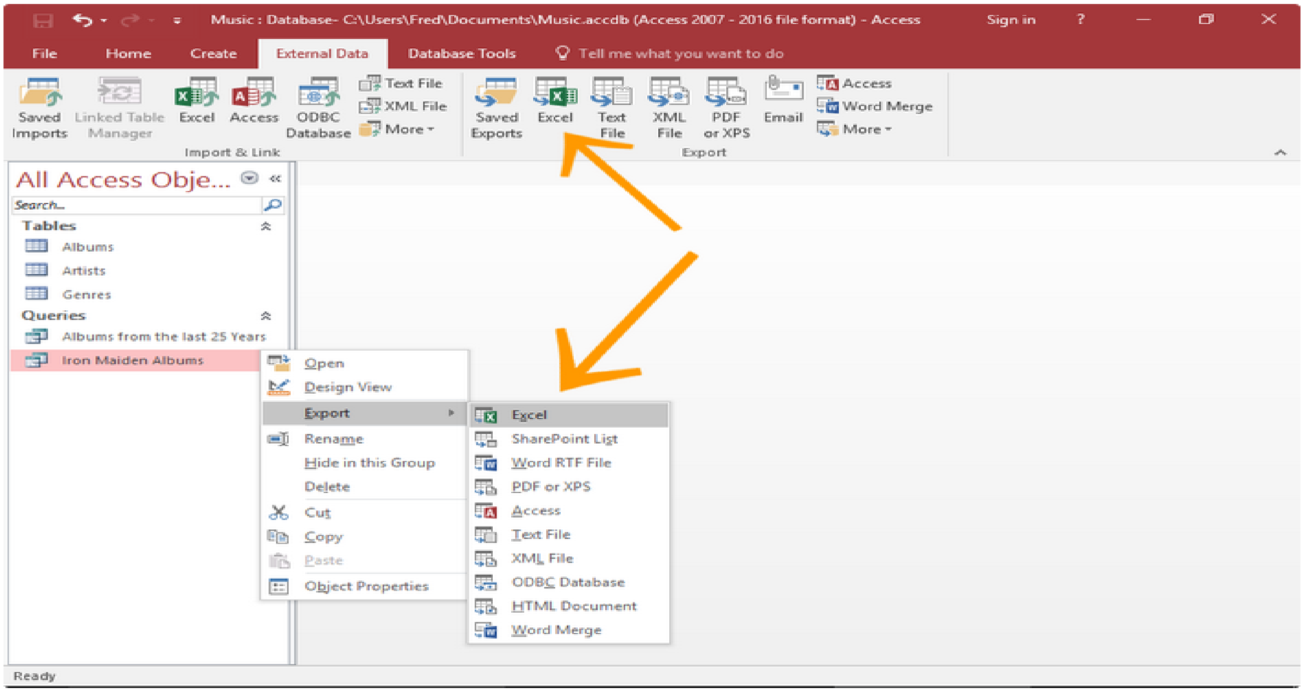
Export data from external files

Access 2016 includes options for exporting data to other sources. You can export to a variety of formats, including:

Excel file, Text file (CSV, TSV, etc.), XML file, PDF file, XPS file, Email, Word document, SharePoint list, HTML document, Another Access database ODBC data source (such as SQL Server), You can export from a table, a query, a form, or a report.

Export a table to Excel

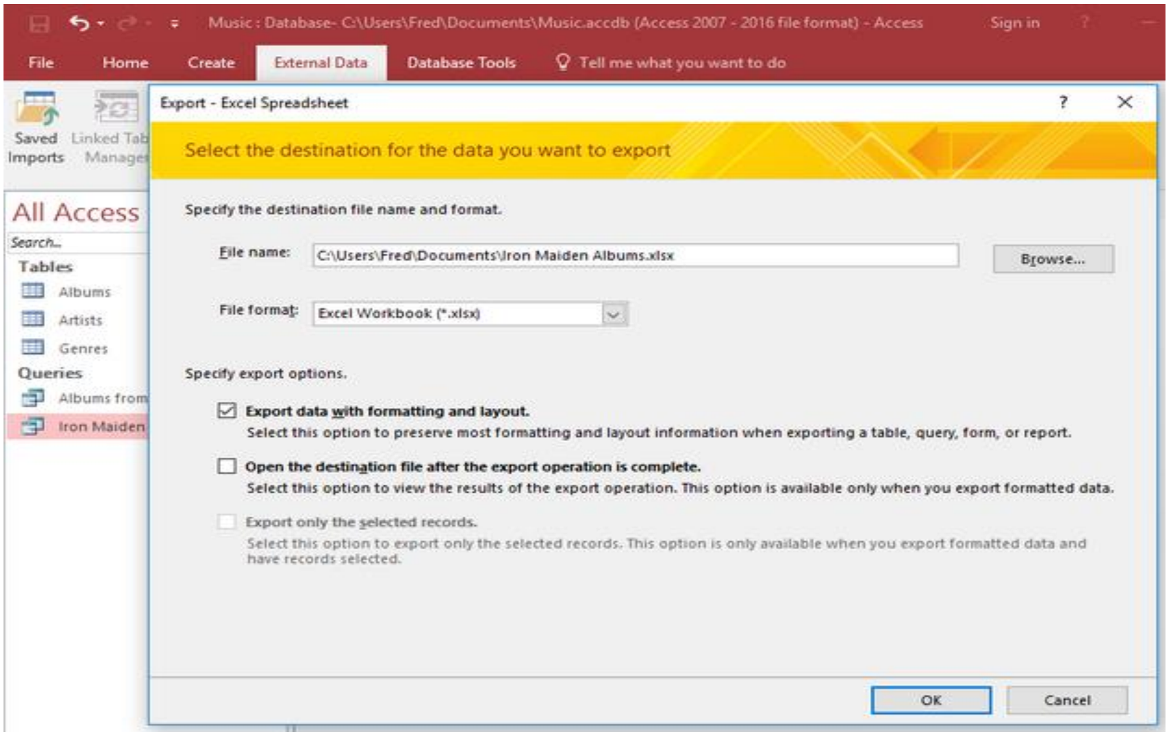
Let's export the results of the previous table that we saved previously. We'll export the table results to an Excel spreadsheet in this case.



1- Right-click on the table in the left menu (that we want to export) and select Export > Excel from the contextual menu.

Export Options

Review the options. In our case, we'll keep the file name as Access has suggested. However, we'll put a tick next to Export data with formatting and layout. Click OK to export the data.



Select the destination for the data you want to export

Specify the destination file name and format.

File name: C:\Users\Fred\Documents\Iron Maiden Albums.xlsx

File format: Excel Workbook (*.xlsx)

Specify export options.

☑ Export data with formatting and layout.
Select this option to preserve most formatting and layout information when exporting a table, query, form, or report.

☐ Open the destination file after the export operation is complete.
Select this option to view the results of the export operation. This option is available only when you export formatted data.

☐ Export only the selected records.
Select this option to export only the selected records. This option is only available when you export formatted data and have records selected.

Import data from external files

You can easily import data from a variety of different sources, including text, Excel, XML, HTML, ODBC data sources, and more.

You can also link to external data sources so that changes in the source file are reflected in your Access database.

Generally, when importing data, you have these options:

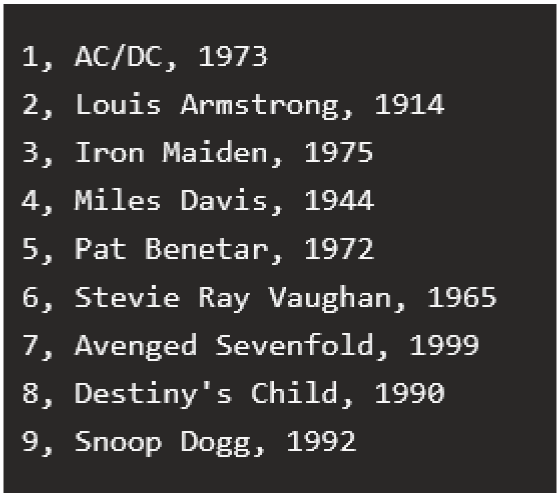
- Import the data into an existing table;
- Have Access to create a new table based on the data; or
- Establish a link to the external file, so that future updates are reflected automatically.

Import a CSV File into an Existing Table

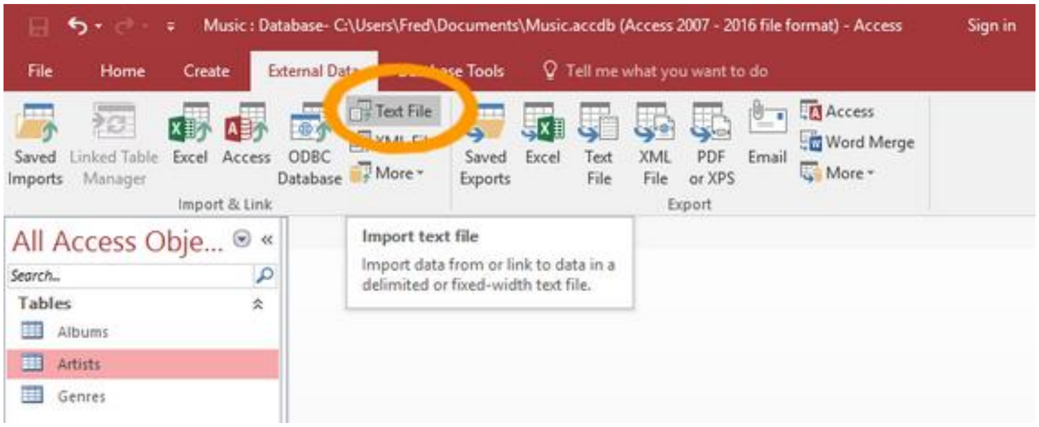
We will import the following CSV file into our table. the CSV file doesn't contain a header row, but that's fine. Access can deal with this too.

*Before starting, ensure that both the source file and the destination table are closed.

*To launch the wizard for a CSV file, click Text File from the External Data tab on the Ribbon.



```
1, AC/DC, 1973
2, Louis Armstrong, 1914
3, Iron Maiden, 1975
4, Miles Davis, 1944
5, Pat Benetar, 1972
6, Stevie Ray Vaughan, 1965
7, Avenged Sevenfold, 1999
8, Destiny's Child, 1990
9, Snoop Dogg, 1992
```

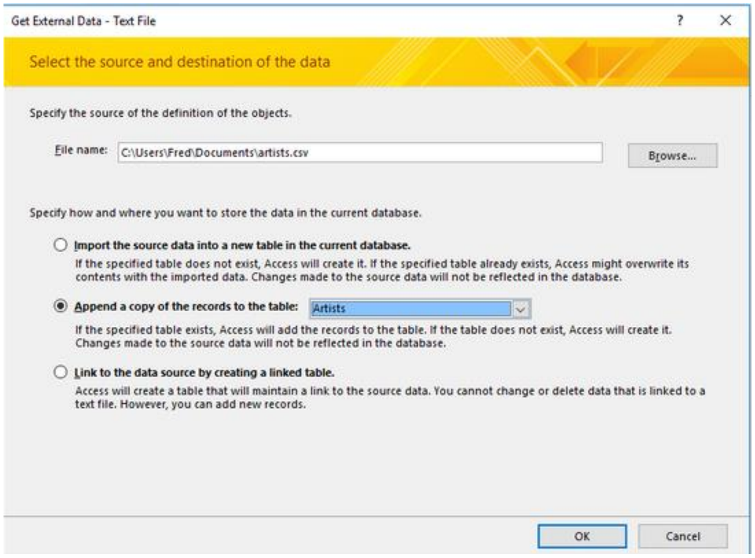


Music : Database- C:\Users\Fred\Documents\Music.accdb (Access 2007 - 2016 file format) - Access

Import text file

Import data from or link to data in a delimited or fixed-width text file.

Select the Source File & Destination Table
Use the Browse... button to navigate to and select the import file.
Select Append a copy of the records to the table and then select the table from the drop-down list of tables. Click OK to continue.



Get External Data - Text File

Select the source and destination of the data

Specify the source of the definition of the objects.

File name: C:\Users\Fred\Documents\artists.csv

Specify how and where you want to store the data in the current database.

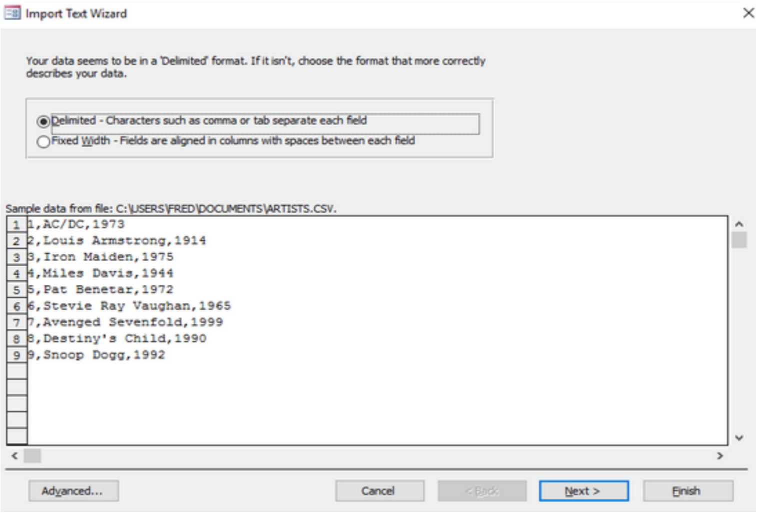
○ Import the source data into a new table in the current database.
If the specified table does not exist, Access will create it. If the specified table already exists, Access might overwrite its contents with the imported data. Changes made to the source data will not be reflected in the database.

● Append a copy of the records to the table: Artists
If the specified table exists, Access will add the records to the table. If the table does not exist, Access will create it. Changes made to the source data will not be reflected in the database.

○ Link to the data source by creating a linked table.
Access will create a table that will maintain a link to the source data. You cannot change or delete data that is linked to a text file. However, you can add new records.

This part of the Import Wizard will ask you to confirm the file's format. In our case, Access has (correctly) detected that our file is in a delimited format. If all looks OK, click Next >.

Now Access will ask you to select the file's delimiter. In our case, Access has (correctly) detected that our file uses a comma as its delimiter. If it was wrong, you can click another delimiter and see how the data updates to reflect the new delimiter.



Import Text Wizard

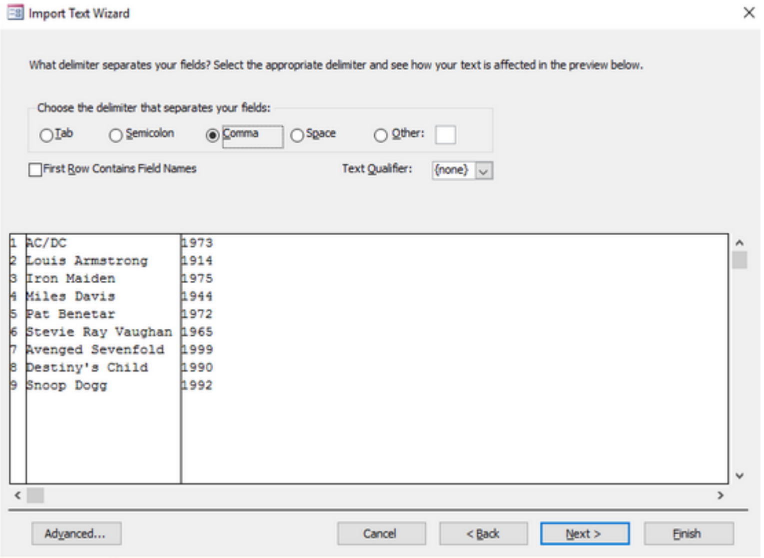
Your data seems to be in a 'Delimited' format. If it isn't, choose the format that more correctly describes your data.

● Delimited - Characters such as comma or tab separate each field
○ Fixed Width - Fields are aligned in columns with spaces between each field

Sample data from file: C:\USERS\FRED\DOCUMENTS\ARTISTS.CSV.

```
1 1,AC/DC,1973
2 2,Louis Armstrong,1914
3 3,Iron Maiden,1975
4 4,Miles Davis,1944
5 5,Pat Benetar,1972
6 6,Stevie Ray Vaughan,1965
7 7,Avenged Sevenfold,1999
8 8,Destiny's Child,1990
9 9,Snoop Dogg,1992
```

You can also select whether or not the file contains header rows. If the first row of your file contains headers, click First Row Contains Field Names. Otherwise, leave it unchecked. Click next, then finish.



Import Text Wizard

What delimiter separates your fields? Select the appropriate delimiter and see how your text is affected in the preview below.

Choose the delimiter that separates your fields:
○ Tab ○ Semicolon ● Comma ○ Space ○ Other: □

□ First Row Contains Field Names   Text Qualifier: {none}

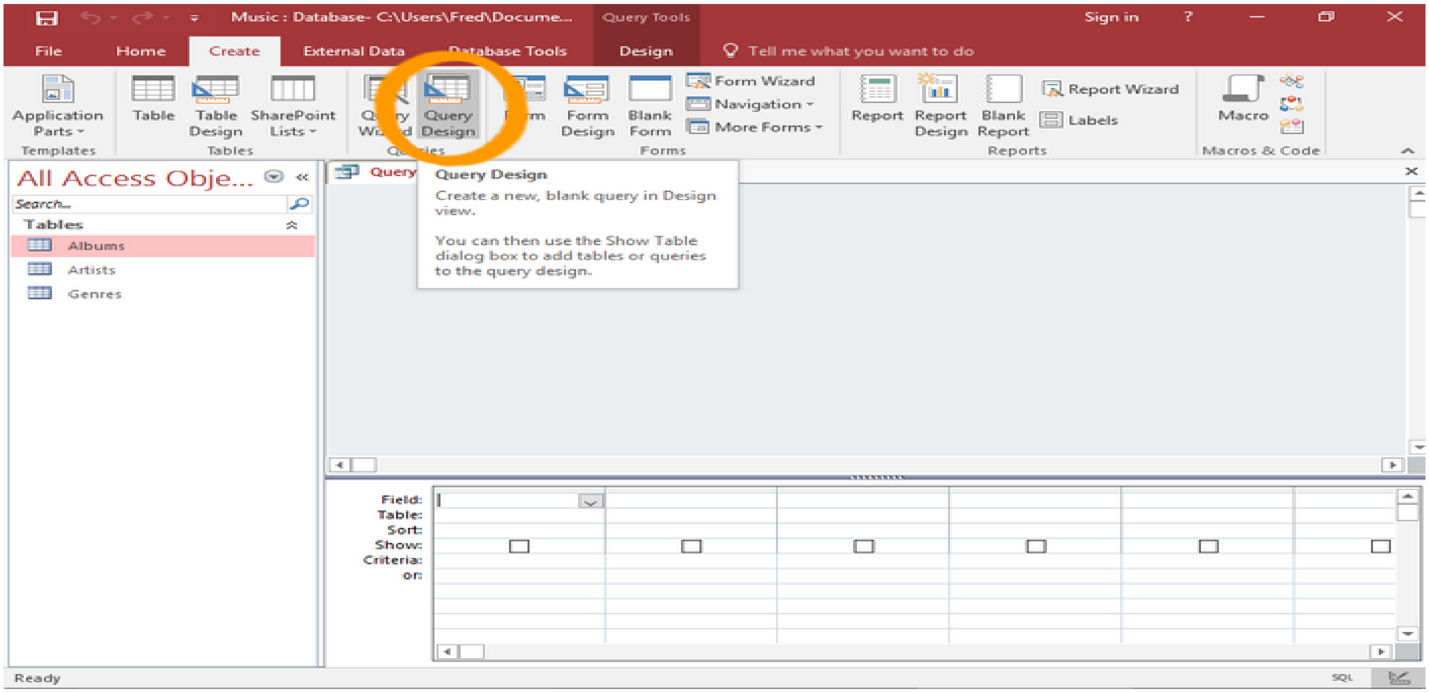
```
1 AC/DC              1973
2 Louis Armstrong    1914
3 Iron Maiden        1975
4 Miles Davis        1944
5 Pat Benetar        1972
6 Stevie Ray Vaughan 1965
7 Avenged Sevenfold  1999
8 Destiny's Child    1990
9 Snoop Dogg         1992
```

2. Query

An object that provides a custom view of data from one or more tables. Queries are a way of searching for and compiling data from one or more tables.

- Running a query is like asking a detailed question about your database.
- When you build a query in Access, you are defining specific search conditions to find exactly the data you want.
- In Access, you can use the graphical query by example facility or you can write Structured Query Language (SQL) statements to create your queries.
- You can define queries to Select, Update, Insert, or Delete data.
- You can also define queries that create new tables from data in one or more existing tables.

We're going to use Query Design to create a query. First, Ensure Data is in your Tables.

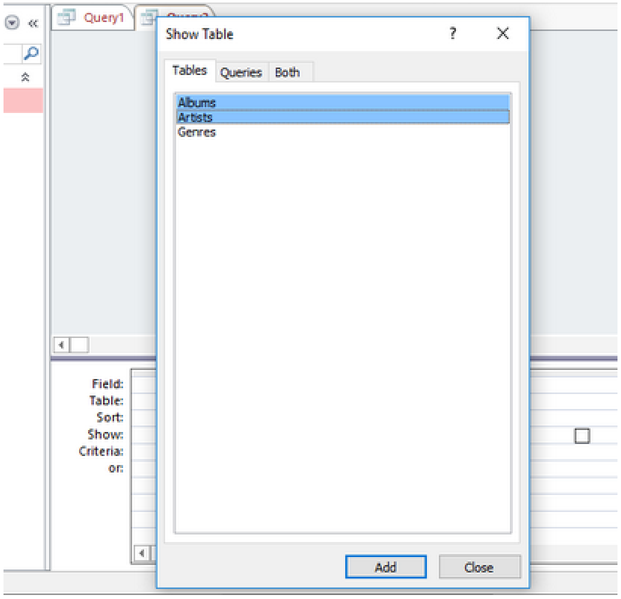


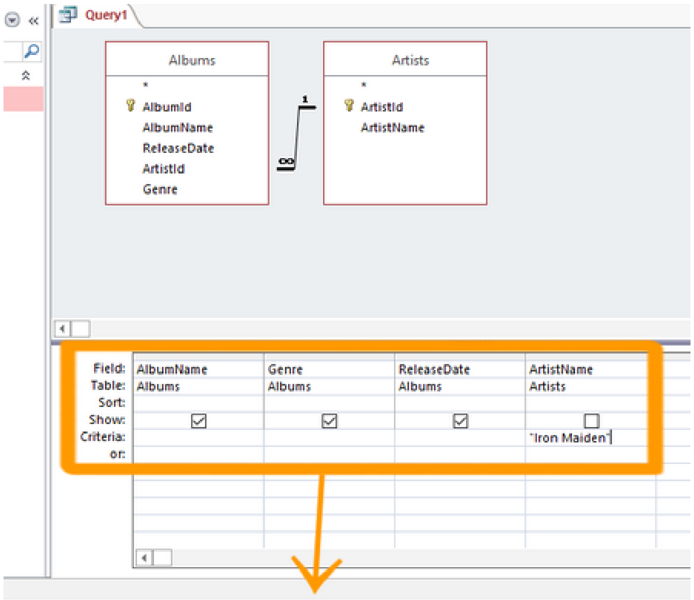
Query Design

Create a new, blank query in Design view.

You can then use the Show Table dialog box to add tables or queries to the query design.

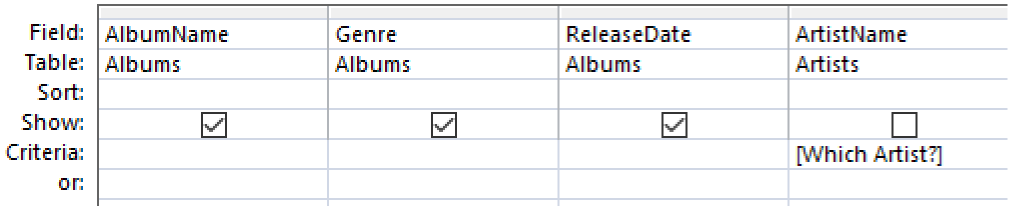
Click Query Design from the Create tab on the Ribbon. The Show Table dialog box will appear. Select the table and click Add. Click Close to close the dialog box.





Field:	AlbumName	Genre	ReleaseDate	ArtistName
Table:	Albums	Albums	Albums	Artists
Sort:				
Show:	☑	☑	☑	□
Criteria:				"Iron Maiden"
or:				

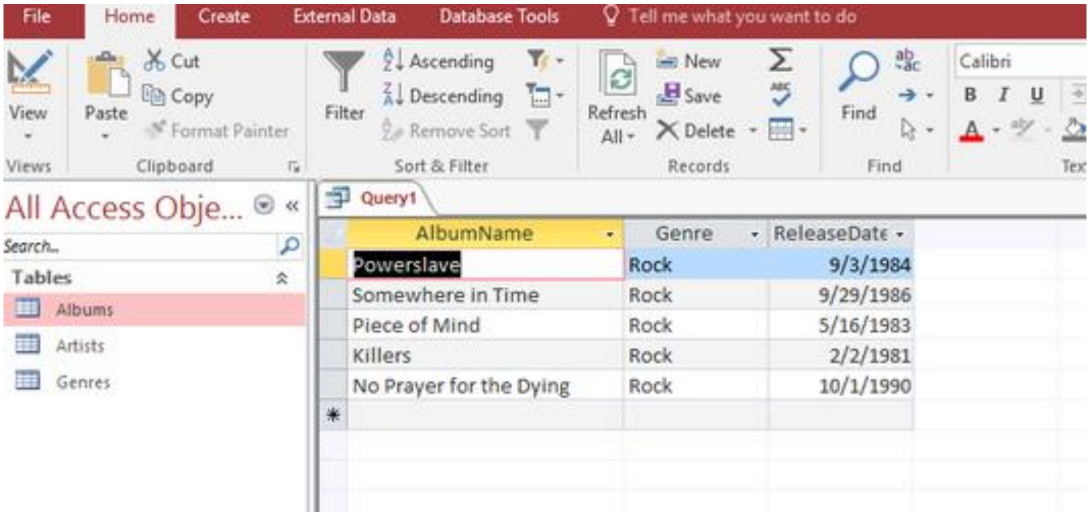
The query results are displayed in Datasheet View. Notice that the results only display the fields for which we selected Show against.



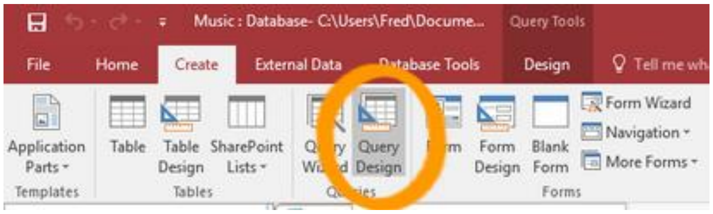
Field:	AlbumName	Genre	ReleaseDate	ArtistName
Table:	Albums	Albums	Albums	Artists
Sort:				
Show:	☑	☑	☑	□
Criteria:				[Which Artist?]
or:				

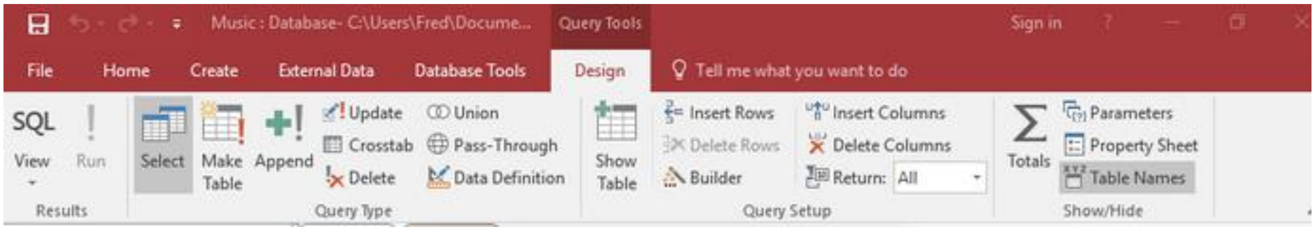
The square brackets will cause a parameter dialog box to open when the user runs the query. The text inside the square brackets will be displayed to the user, and an input field will be displayed for their input.

Now run the query (either by clicking Run or View in the Ribbon, or clicking the Datasheet View icon in the bottom right corner of the screen).



AlbumName	Genre	ReleaseDate
Powerslave	Rock	9/3/1984
Somewhere in Time	Rock	9/29/1986
Piece of Mind	Rock	5/16/1983
Killers	Rock	2/2/1981
No Prayer for the Dying	Rock	10/1/1990



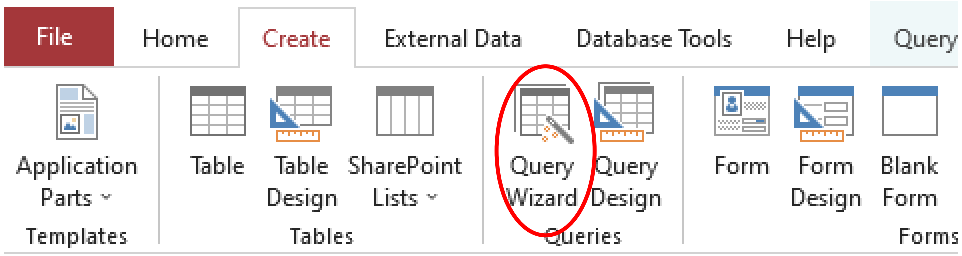


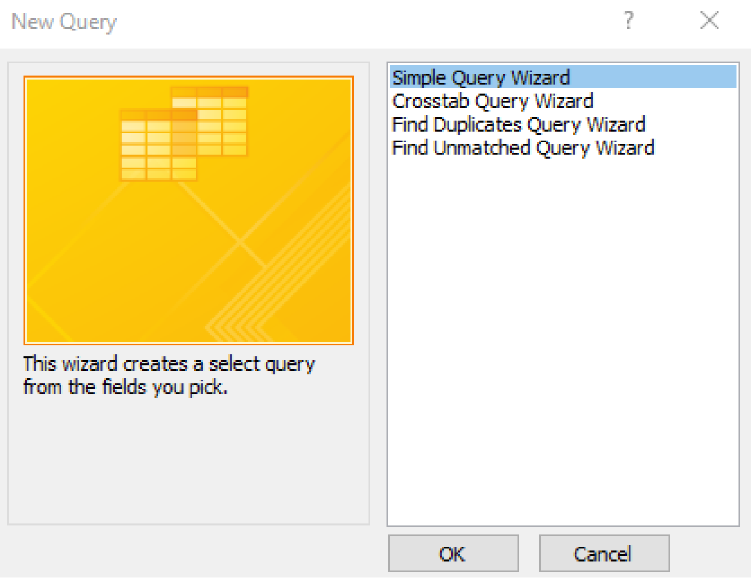
Make Table	To create a new table for the query: click the make table first, you need to enter the table name, then write your criteria and run.
Append	To add a new record for the table depends on the query criteria: after a click on the append you need to enter your table to add the record to it
Update	To update the data in the table depending on the criteria
Delete	To delete the data in the table depending on the criteria

*If you want to add a new column to a table and this column takes its data from another column, you need to write **name-of-new-column** followed by "**:**" then **[column-name]** as an example

New salary: [salary] + 1000

• If we are going to use Query Wizard, will need to choose one of four queries type wizard





This wizard creates a select query from the fields you pick.

- The first one use to create queries automatically with a simple design.
- The second type is used to create a new table with a new heading for rows and columns. And this needs to detect the fields to be row and column
- The third type is used to detect the data that is repeated in the same column.
- finally: this type is used to detect the data found in the first table and not found in the second table

Each text or command that wants to run, should be written inside square brackets. There are some commands necessary such as:

Criteria for Text, Memo, and Hyperlink fields

To include records that...	Use this criterion	Query result
Exactly match a value, such as China	"China"	Returns records where the CountryRegion field is set to China.
Do not match a value, such as Mexico	Not "Mexico"	Returns records where the CountryRegion field is set to a country/region other than Mexico.
Begin with the specified string, such as U	Like U*	Returns records for all countries/regions whose names start with "U", such as the UK, USA, and so on.

		Note: When used in an expression, the asterisk (*****) represents any string of characters — it is also called a wildcard character. For a list of such characters, see the article.
Do not begin with the specified string, such as U	Not Like U*	Returns records for all countries/regions whose names start with a character other than "U".
Contain the specified string, such as Korea	Like "*Korea*"	Returns records for all countries/regions that contain the string "Korea".
Do not contain the specified string, such as Korea	Not Like "*Korea*"	Returns records for all countries/regions that do not contain the string "Korea".
End with the specified string, such as "ina"	Like "*ina"	Returns records for all countries/regions whose names end in "ina", such as China and Argentina.
Do not end with the specified string, such as "ina"	Not Like "*ina"	Returns records for all countries/regions that do not end in "ina", such as China and Argentina.
Contain null (or missing) values	Is Null	Returns records where there is no value in the field.
Do not contain null values	Is Not Null	Returns records where the value is not missing in the field.
Contain zero-length strings	"” (a pair of quotes)	Returns records where the field is set to a blank (but not null) value. For example, records of sales made to another department might contain a blank value in the CountryRegion field.
Do not contain zero-length strings	Not ""	Returns records where the CountryRegion field has a nonblank value.
Contains null values or zero-length strings	"" Or Is Null	Returns records where there is either no value in the field, or the field is set to a blank value.
Is not empty or blank	Is Not Null And Not ""	Returns records where the CountryRegion field has a nonblank, non-null value.
Follow a value, such as Mexico, when sorted in alphabetical order	>= "Mexico"	Returns records of all countries/regions, beginning with Mexico and continuing through the end of the alphabet.
Fall within a specific range, such as A through D	Like "[A-D]*"	Returns records for countries/regions whose names start with the letters "A" through "D".
Match one of two values, such as USA or UK	"USA" Or "UK"	Returns records for USA and UK.

Contain one of the values in a list of values	In("France", "China", "Germany", "Japan")	Returns records for all countries/regions specified in the list.
Contain certain characters at a specific position in the field value	Right([CountryRegion], 1) = "y"	Returns records for all countries/regions where the last letter is "y".
Satisfy length requirements	Len([CountryRegion]) > 10	Returns records for countries/regions whose name is more than 10 characters long.
Match a specific pattern	Like "Chi??"	Returns records for countries/regions, such as China and Chile, whose names are five characters long and the first three characters are "Chi". **Note:** The characters **?** and **_**, when used in an expression, represent a single character — these are also called wildcard characters. The character **_** cannot be used in the same expression with the **?** character, nor can it be used in an expression with the ***** wildcard character. You may use the wildcard character **_** in an expression that also contains the **%** wildcard character.

Criteria for Number, Currency, and AutoNumber fields

To include records that...	Use this crit erion	Query Result
Exactly match a value, such as 100	100	Returns records where the unit price of the product is $100.
Do not match a value, such as 1000	Not 1000	Returns records where the unit price of the product is not $1000.
Contain a value smaller than a value, such as 100	< 100 <= 100	Returns records where the unit price is less than $100 (<100). The second expression (<=100) displays records where the unit price is less than or equal to $100.
Contain a value larger than a value, such as 99.99	>99.99 >=99.99	Returns records where the unit price is greater than $99.99 (>99.99). The second expression displays records where the unit price is greater than or equal to $99.99.
Contain one of the two values, such as 20 or 25	20 or 25	Returns records where the unit price is either $20 or $25.
Contain a value that falls with a range of values	>49.99 and <99.99	Returns records where the unit price is between (but not including) $49.99 and $99.99.

	-or- Between 50 and 100	
Contain a value that falls outside a range	<50 or >100	Returns records where the unit price is not between $50 and $100.
Contain one of many specific values	In(20, 25, 30)	Returns records where the unit price is either $20, $25, or $30.
Contain a value that ends with the specified digits	Like "*4.99"	Returns records where the unit price ends with "4.99", such as $4.99, $14.99, $24.99, and so on. **Note:** The characters * and **%**, when used in an expression, represent any number of characters — these are also called wildcard characters. The character **%** cannot be used in the same expression with the * character, nor can it be used in an expression with the **?** wildcard character. You may use the wildcard character **%** in an expression that also contains the _ wildcard character.
Contain null null (or missing) values	Is Null	Returns records where no value is entered in the UnitPrice field.
Contain non-null values	Is Not Null	Returns records where the value is not missing in the UnitPrice field.

Criteria for Date/Time fields

To include records that ...	Use this criterion	Query result
Exactly match a value, such as 2/2/2006	#2/2/2006#	Returns records of transactions that took place on Feb 2, 2006. Remember to surround date values with the # character so that Access can distinguish between date values and text strings.
Do not match a value, such as 2/2/2006	Not #2/2/2006#	Returns records of transactions that took place on a day other than Feb 2, 2006.
Contain values that fall before a certain date, such as 2/2/2006	< #2/2/2006#	Returns records of transactions that took place before Feb 2, 2006.

		To view transactions that took place on or before this date, use the **<=** operator instead of the **<** operator.
Contain values that fall after a certain date, such as 2/2/2006	> #2/2/2006#	Returns records of transactions that took place after Feb 2, 2006. To view transactions that took place on or after this date, use the **>=** operator instead of the **>** operator.
Contain values that fall within a date range	>#2/2/2006# and <#2/4/2006#	Returns records where the transactions took place between Feb 2, 2006 and Feb 4, 2006. You can also use the **Between** operator to filter for a range of values, including the end points. For example, Between #2/2/2006# and #2/4/2006# is the same as >=#2/2/2006# and <=#2/4/2006# .
Contain values that fall outside a range	<#2/2/2006# or >#2/4/2006#	Returns records where the transactions took place before Feb 2, 2006 or after Feb 4, 2006.
Contain one of two values, such as 2/2/2006 or 2/3/2006	#2/2/2006# or #2/3/2006#	Returns records of transactions that took place on either Feb 2, 2006 or Feb 3, 2006.
Contain one of many values	In (#2/1/2006#, #3/1/2006#, #4/1/2006#)	Returns records where the transactions took place on Feb 1, 2006, March 1, 2006, or April 1, 2006.
Contain a date that falls in a specific month (irrespective of year), such as December	DatePart("m", [SalesDate]) = 12	Returns records where the transactions took place in December of any year.

Contain a date that falls in a specific quarter (irrespective of year), such as the first quarter	DatePart("q", [SalesDate]) = 1	Returns records where the transactions took place in the first quarter of any year.
Contain today's date	Date()	Returns records of transactions that took place on the current day. If today's date is 2/2/2006, you see records where the OrderDate field is set to Feb 2, 2006.
Contain yesterday's date	Date()-1	Returns records of transactions that took place the day before the current day. If today's date is 2/2/2006, you see records for Feb 1, 2006.
Contain tomorrow's date	Date() + 1	Returns records of transactions that took place the day after the current day. If today's date is 2/2/2006, you see records for Feb 3, 2006.
Contain dates that fall during the current week	DatePart("ww", [SalesDate]) = DatePart("ww", Date()) and Year([SalesDate]) = Year(Date())	Returns records of transactions that took place during the current week. A week starts on Sunday and ends on Saturday.
Contain dates that fell during the previous week	Year([SalesDate])* 53 + DatePart("ww", [SalesDate]) = Year(Date())* 53 + DatePart("ww", Date()) - 1	Returns records of transactions that took place during the last week. A week starts on Sunday and ends on Saturday.
Contain dates that fall during the following week	Year([SalesDate])* 53+DatePart("ww", [SalesDate]) = Year(Date())* 53+DatePart("ww", Date()) + 1	Returns records of transactions that will take place next week. A week starts on Sunday and ends on Saturday.
Contain a date that fell during the last 7 days	Between Date() and Date()-6	Returns records of transactions that took place during the last 7 days. If today's date is 2/2/2006, you see

		records for the period Jan 24, 2006 through Feb 2, 2006.
Contain a date that belongs to the current month	Year([SalesDate]) = Year(Now()) And Month([SalesDate]) = Month(Now())	Returns records for the current month. If today's date is 2/2/2006, you see records for Feb 2006.
Contain a date that belongs to the previous month	Year([SalesDate])* 12 + DatePart("m", [SalesDate]) = Year(Date())* 12 + DatePart("m", Date()) - 1	Returns records for the previous month. If today's date is 2/2/2006, you see records for Jan 2006.
Contain a date that belongs to the next month	Year([SalesDate])* 12 + DatePart("m", [SalesDate]) = Year(Date())* 12 + DatePart("m", Date()) + 1	Returns records for the next month. If today's date is 2/2/2006, you see records for Mar 2006.
Contain a date that fell during the last 30 or 31 days	Between Date( ) And DateAdd("M", -1, Date( ))	A month's worth of sales records. If today's date is 2/2/2006, you see records for the period Jan 2, 2006. to Feb 2, 2006
Contain a date that belongs to the current quarter	Year([SalesDate]) = Year(Now()) And DatePart("q", Date()) = DatePart("q", Now())	Returns records for the current quarter. If today's date is 2/2/2006, you see records for the first quarter of 2006.
Contain a date that belongs to the previous quarter	Year([SalesDate])*4+DatePart("q",[SalesDate]) = Year(Date())*4+DatePart("q",Date())- 1	Returns records for the previous quarter. If today's date is 2/2/2006, you see records for the last quarter of 2005.
Contain a date that belongs to the next quarter	Year([SalesDate])*4+DatePart("q",[SalesDate]) = Year(Date())*4+DatePart("q",Date())+1	Returns records for the next quarter. If today's date is 2/2/2006, you see records for the second quarter of 2006.
Contain a date that falls during the current year	Year([SalesDate]) = Year(Date())	Returns records for the current year. If today's date is 2/2/2006, you see records for the year 2006.
Contain a date that belongs to the previous year	Year([SalesDate]) = Year(Date()) - 1	Returns records of transactions that took place during the previous year. If today's date is 2/2/2006, you see records for the year 2005.

Contain a date that belongs to next year	Year([SalesDate]) = Year(Date()) + 1	Returns records of transactions with next year's date. If today's date is 2/2/2006, you see records for the year 2007.
Contain a date that falls between Jan 1 and today (year to date records)	Year([SalesDate]) = Year(Date()) and Month([SalesDate]) <= Month(Date()) and Day([SalesDate]) <= Day (Date())	Returns records of transactions with dates that fall between Jan 1 of the current year and today. If today's date is 2/2/2006, you see records for the period Jan 1, 2006 to to 2/2/2006.
Contain a date that occurred in the past	< Date()	Returns records of transactions that took place before today.
Contain a date that occurs in the future	> Date()	Returns records of transactions that will take place after today.
Filter for null (or missing) values	Is Null	Returns records where the date of transaction is missing.
Filter for non-null values	Is Not Null	Returns records where the date of transaction is known.

Criteria for Yes/No fields

Field value	**Result**
Yes, True, 1, or -1	Tested for a Yes value. A value of 1 or -1 is converted to "True" in the Criteria row after you enter it.
No, False, or 0	Tested for a No value. A value of 0 is converted to "False" in the Criteria row after you enter it.
No value (null)	Not Tested
Any number other than 1, -1, or 0	No results if it's the only criteria value in the field
Any character string other than Yes, No, True, or False	Query fails to run due to Data type mismatch error

3. Form

A form is an object in a desktop database designed primarily for data input or display or for control of application execution. You use forms to customize the presentation of data that your application extracts from queries or tables.

- Forms are used for entering, modifying, and viewing records.
- The reason forms are used so often is that they are an easy way to guide people toward entering data correctly.
- When you enter information into a form in Access, the data goes exactly where the database designer wants it to go in one or more related tables.

Create a Form from a Table

1. Create the form
Select the table in the left menu that you want to use for the form and click Form from the Create tab on the Ribbon. Then choose "**design form**".

Form Views:

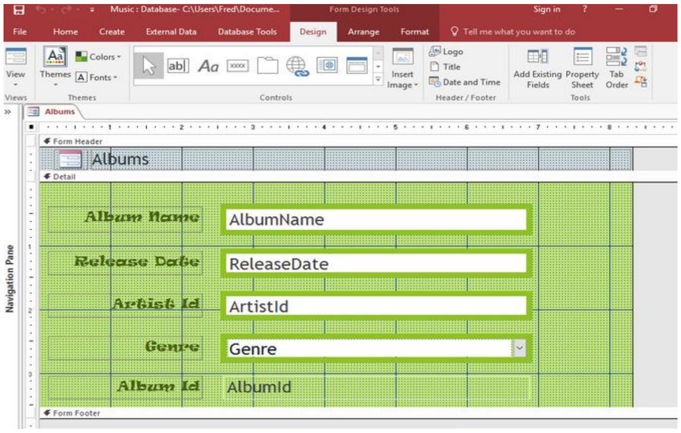
There are three different views for working with forms:

- Design View
- Layout View
- Form View

When you first create a form from a table, Access opens it in Layout View. You can customize the form however you like. You can switch between views as often as you like while customizing your form.

Switching views can be done by using the View button on the Ribbon, or the three icons at the bottom right of the Access window.

Here's an explanation of each view.



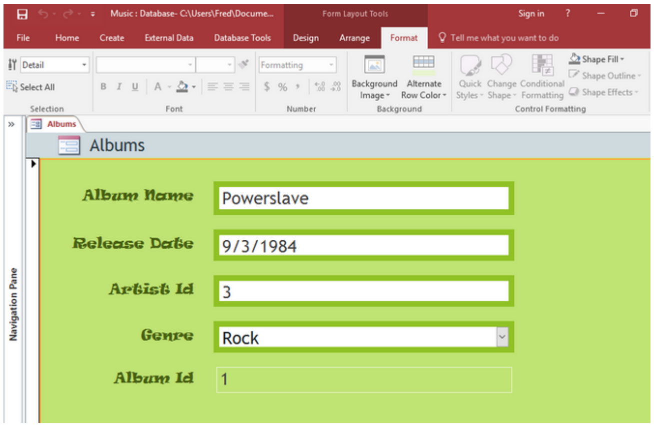
Albums

Album Name	AlbumName
Release Date	ReleaseDate
Artist Id	ArtistId
Genre	Genre
Album Id	AlbumId

Design View:

provides a more detailed view of the form's structure than Layout View. Design View also allows you to modify the form without having any data getting in the way.

Most tasks can be done in either Design View or Layout View, but there are some form properties that can only be changed in Design View.



Albums

Album Name	Powerslave
Release Date	9/3/1984
Artist Id	3
Genre	Rock
Album Id	1

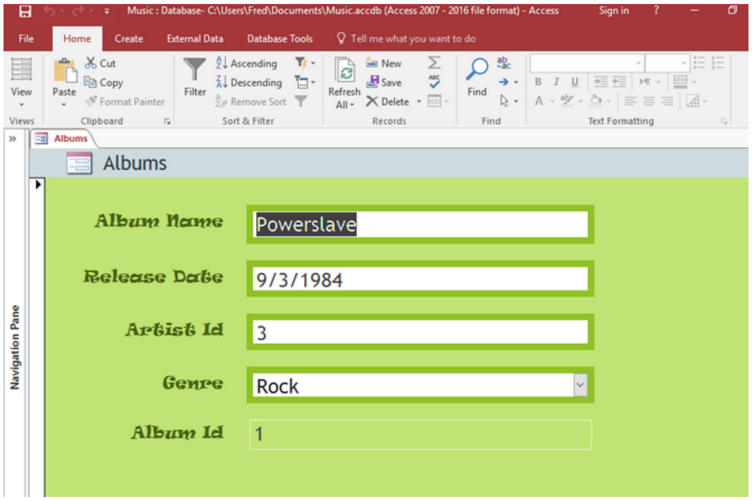
Layout View

Layout View provides a more visual layout for editing the form. It bears a close resemblance to the actual form that the user will see. Layout View allows you to edit the form properties while it has real data.

As mentioned, there are some things that can't be done in Layout View. In some cases, Access will tell you that you need to switch to Design View if you want to make a certain change.

Form View

Form View is how the user will see the form. You can't edit the form's properties, but you can use the form just as a user would.



Albums

Album Name	Powerslave
Release Date	9/3/1984
Artist Id	3
Genre	Rock
Album Id	1

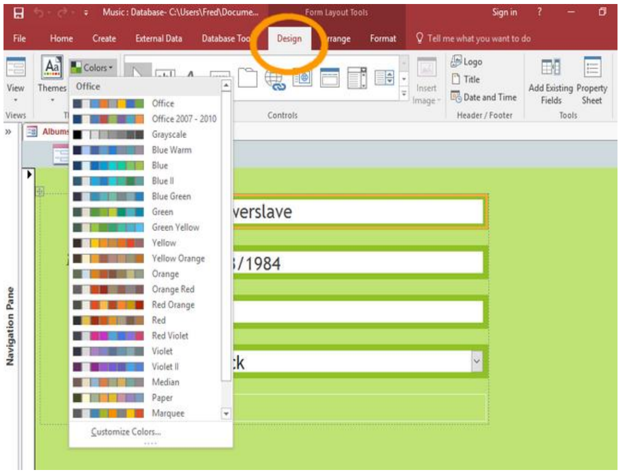
The Form Tabs

When you edit a form in either Layout View or Design View, three tabs appear on the Ribbon with specific options for editing forms.

Here's an overview of these tabs.

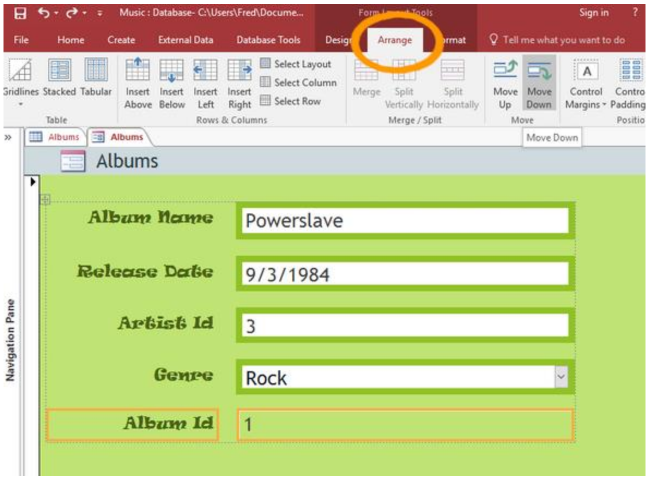
The Design Tab

You can use the Design tab to change the color of the form, change fonts, or change the whole theme. You can also upload a logo, add/edit a title, add the date and time, etc. The Design tabs also allow you to add form controls such as buttons, text input, combo box, etc.



The Arrange Tab

You can use the Arrange tab on the Ribbon to arrange individual elements. Select the elements and click an option to re-arrange them.



Albums

Album Name: Powerslave

Release Date: 9/3/1984

Artist Id: 3

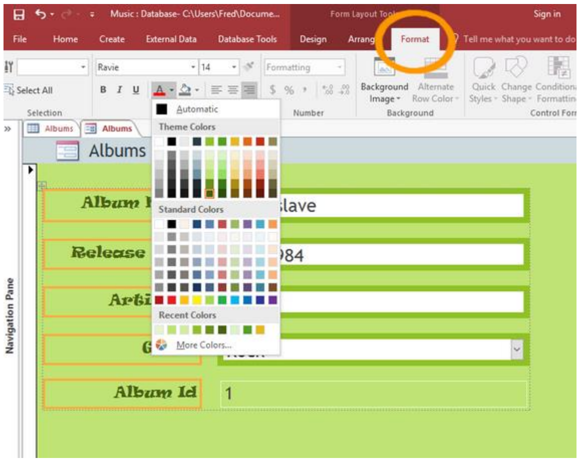
Genre: Rock

Album Id: 1

For example, you can move a field under the field beneath it by using the Move Down option. You can also set padding, and margins, add arbitrary spaces, etc.

The Format Tab

You can use the Format tab on Ribbon to format individual elements. To style an individual element, simply select the element and apply a style.



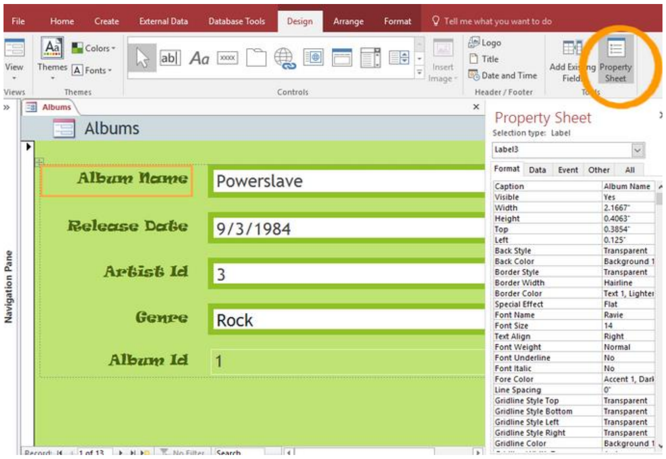
Album I	slave
Release	84
Arti	
G	
Album Id	1

Form Properties

Access provides the Property Sheet for setting properties against a form and its elements. Here's an overview.

The Property Sheet

You can use the Property Sheet to apply settings to individual elements, or the whole form. There are many different properties that can be applied across a form. These aren't just limited to formatting properties, they include data validation properties, event-based properties, and more.



Album Name	Powerslave
Release Date	9/3/1984
Artist Id	3
Genre	Rock
Album Id	1

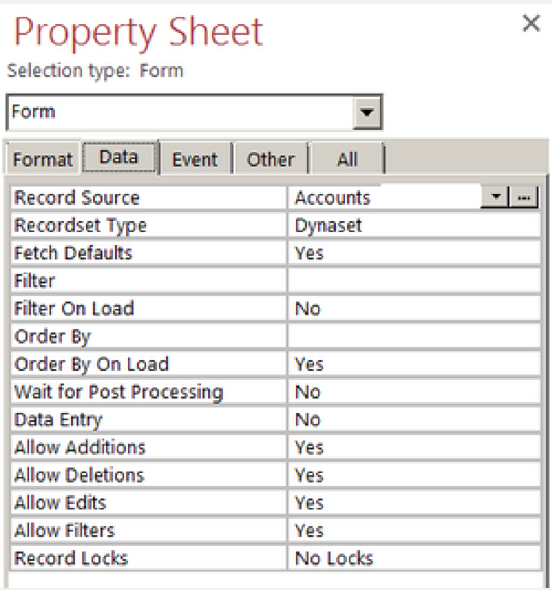
Property Sheet

Selection type: Label

Label3

Caption	Album Name
Visible	Yes
Width	2.1667"
Height	0.4063"
Top	0.3854"
Left	0.125"
Back Style	Transparent
Back Color	Background 1
Border Style	Transparent
Border Width	Hairline
Border Color	Text 1, Lighter
Special Effect	Flat
Font Name	Ravie
Font Size	14
Text Align	Right
Font Weight	Normal
Font Underline	No
Font Italic	No
Fore Color	Accent 1, Dark
Line Spacing	0"
Gridline Style Top	Transparent
Gridline Style Bottom	Transparent
Gridline Style Left	Transparent
Gridline Style Right	Transparent
Gridline Color	Background 1

The Property Sheet displays properties for the selected element. When you select another element, the properties in the Property Sheet will update to reflect the properties for that element. To open the Property Sheet, click Property Sheet from the Design tab in the Ribbon.



Property Sheet

Selection type: Form

Form

Record Source	Accounts
Recordset Type	Dynaset
Fetch Defaults	Yes
Filter	
Filter On Load	No
Order By	
Order By On Load	Yes
Wait for Post Processing	No
Data Entry	No
Allow Additions	Yes
Allow Deletions	Yes
Allow Edits	Yes
Allow Filters	Yes
Record Locks	No Locks

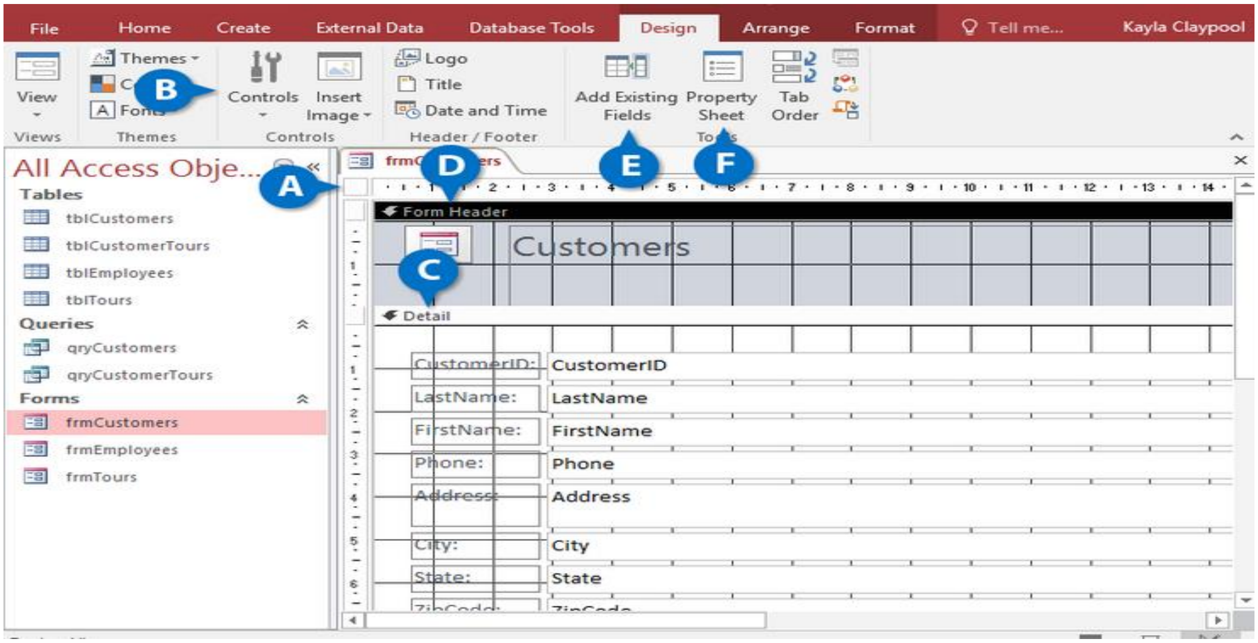
The **Data tab** displays properties of the form related to the data that is used to populate the form and how the data is to be displayed and manipulated.

Property	Description
Record Source	The source of data for the form. Forms may only have one source of data "bound" to the fields on the form. This source can be a query but most of the time it will be a table.
Recordset Type	The type of Recordset object created from the record source. In virtually all cases this should be set to Dynaset since a Dynaset maintains consistency with the underlying table.
Fetch Defaults	**Yes,** this indicates default values set in the underlying table will be populated in the fields when a new record is created.
Filter	Sets any filtering criteria for when the form is opened. By default, no filters are applied but can be added by the user once the form is displayed.
Filter on load	If a filter is set, **Yes** indicates the filter will be applied as the form is loading.
Order By	Specifies the order of the records as they are displayed in the form. For example, putting "ID" for this property will cause the form to display the Accounts records ordered by ID.
Data Entry	**Yes,** indicates the form can *only* be used to add new data records. **No** is the default.
Allow Additions	**Yes,** indicates this form can be used to add new records.
Allow Deletions	**Yes,** indicates this form can be used to delete existing records.
Allow Edits	**Yes,** indicates this form can be used to edit existing records.
Allow Filters	**Yes,** indicates users can add filters to display a subset of the records in the form.
Record Locks	Indicates how data should be locked when multiple users are working on the same data. Locks are used to prevent two users from overwriting the same data. **No Locks** indicate no locking is done. This is the default but may not be reasonable if you plan to share this data with multiple users. **All Records** indicates all records in the table are locked when a user edits one record. **Edited Record** indicates only the current record being edited will be locked.

The **Format properties tab** for a form controls the appearance of the form including scrollbars, headings, the default layout of fields, etc. Some handy properties include:

Property	Description
Caption	Form Caption – this is the name that appears on the top tab when the form is open.
Default View	The default view when the form is opened. Can be: Single Form, Continuous Form, Data Sheet, split Form
Allow Form View	**Yes,** indicates the regular form view is allowed.
Allow Datasheet View	**Yes,** indicates the form can be shown as a data sheet.
Allow Layout View	**Yes,** indicates the form can be switched to the Layout view.
Scroll Bars	Can be **Neither**, **Horizontal**, **Vertical** or **Both**
Close Button	**Yes,** indicates a Close button will appear in the upper right corner of the form.

While form-level properties apply to entire forms, field-level properties apply to specific fields. if the Property Sheet is no longer open, right-click on the specific field and select Properties from the pop-up menu. If the Property Sheet is already displayed then simply click once on this field. The Data tab for the field properties is shown below:

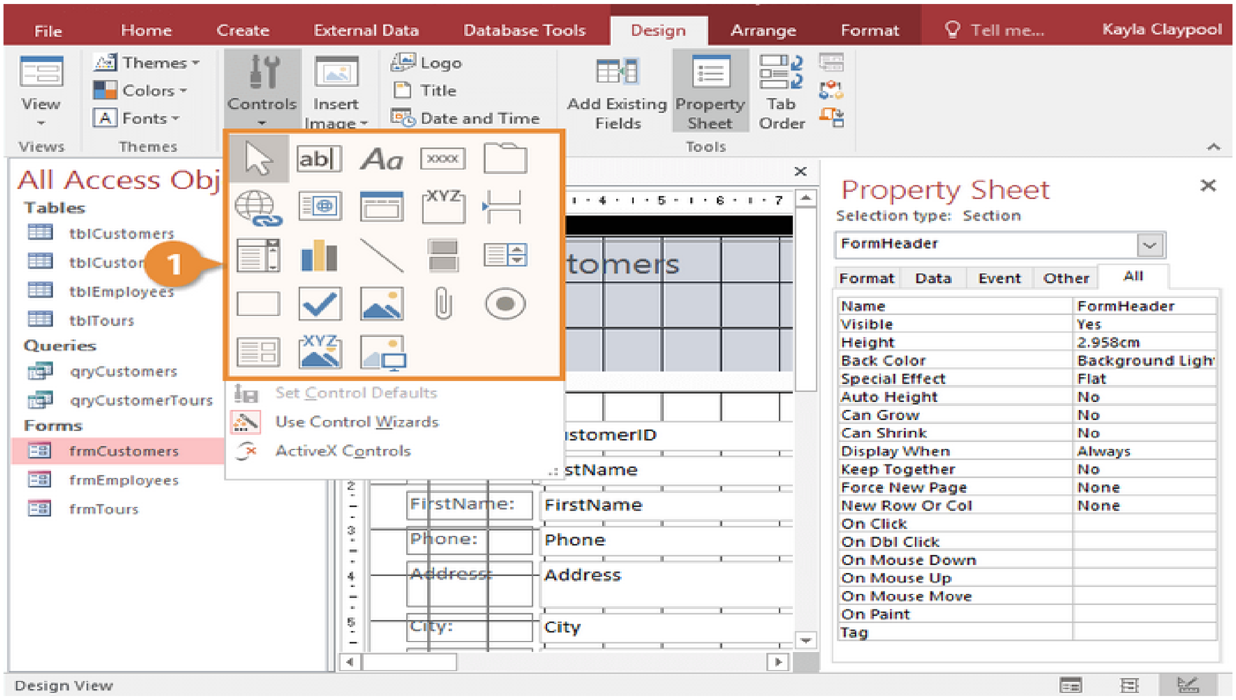


A	Form Selector: Click the Form Selector to select the entire form. Double-click it to display the form's properties.	D	Form header: The form header appears at the top of the form.
B	**Controls group**: The Controls group is where you can add controls to your form. Click a button in the Controls gallery to create a control on the form.	E	**Add Existing Fields button**: Click the Add Existing Fields button to display the Field List pane.
C	**Detail divider**: To enlarge the form's header, just click and drag the header down.	F	**Property Sheet button**: Click the Property Sheet button to display the form's property sheet.

Add a Control

Any graphic object that appears on forms and reports is called a control. A text box used to enter and display information, a text label, and a button you click to print a report would all be examples of controls. Let's look at how to add controls in Design View.

1. In Design View, click the control button you want to add from the Controls group. Some controls, such as buttons or lists, have a wizard that helps you set them up.



Property Sheet

Selection type: Section

FormHeader

Format | Data | Event | Other | All

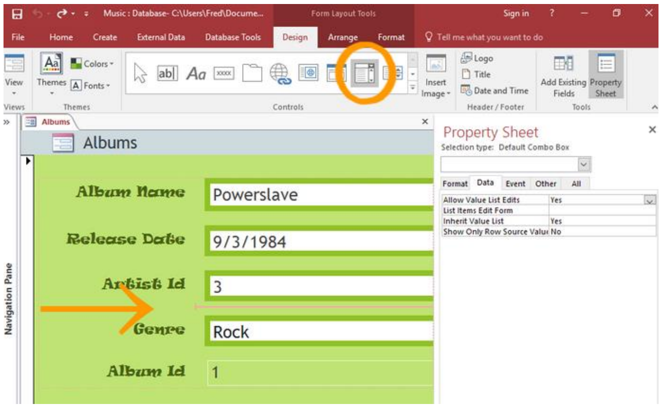
Name	FormHeader
Visible	Yes
Height	2.958cm
Back Color	Background Ligh
Special Effect	Flat
Auto Height	No
Can Grow	No
Can Shrink	No
Display When	Always
Keep Together	No
Force New Page	None
New Row Or Col	None
On Click	
On Dbl Click	
On Mouse Down	
On Mouse Up	
On Mouse Move	
On Paint	
Tag	

Controls	
Select	Click this button and then click the control you want to select. To select multiple controls, click this button and hold down the **Shift** key as you click each control, or drag a rectangle around all controls you want to select.
Text Box	Creates a text box that displays information from a table and query. You can also use text boxes to simply enter text.
Label	Creates a static text label that is the same for every record, such as a heading. Most controls already have a text label attached.
Button	Creates a button that runs a macro or Visual Basic function.
Tab Control	Enables you to create tabs (like the ones found in some dialog boxes) to include more than one page of controls on the form.
Hyperlink	Inserts a link to a webpage or file.
Web Browser Control	Inserts a control that allows the database user to access the computer's web browser.
Navigation Control	Creates a control that allows the form to be navigated.
Option Group	Creates a box around a group of option buttons so that the user is only allowed to make one selection from the group box.
Insert Page Break	Inserts a page break.
Combo Box	Creates a drop-down box that lets the user enter text or select an item from a list of options.
Chart	Inserts a chart.
Line	Enables you to draw a line.
Toggle Button	Creates a toggle button that allows you to display and enter data from a Yes/No field.
List Box	Creates a box that lets the user select an item from a list of options.
Rectangle	Enables you to draw a rectangle.
Check box	Creates a box that is checked or unchecked. Use to enter data from a Yes/No field.
Unbound Object Frame	Inserts an OLE object that is not bound to a field in the current database. Use an Unbound Object Frame to display information from an external source or program, such as a spreadsheet, graphic, or another file.
Attachment	Provides the option to include an attachment.
Option Button	Creates an option button (or radio button) that allows the user to make a single selection from two or more choices. Option buttons are normally used with the Options Group control.

Subform/Subreport	Inserts another form within the main form. Use it when you want to show data from a one-to-many relationship.
Bound Object Frame	Inserts an OLE object that is bound to a field in the database. Use Bound Object Frames to display pictures or other OLE information in the database.
Image	Displays a picture or graphic file that you specify.
Logo	To add the a in the form header
Title	To add a title in the form header
Date/time	To add the date and time in the form header

Add a Combo Box

Open the form in Layout View. Click the Combo Box icon from the Design tab on the Ribbon. Now click the form in the location where you want the combo box to be inserted. The Combo Box Wizard will open.



Albums

Album Name	Powerslave
Release Date	9/3/1984
Artist Id	3
Genre	Rock
Album Id	1

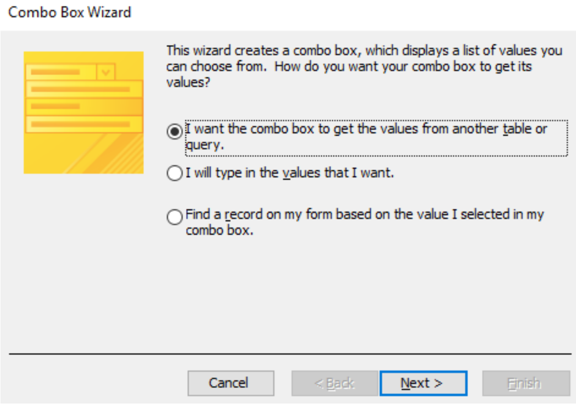
Property Sheet

Selection type: Default Combo Box

Format | Data | Event | Other | All

Allow Value List Edits	Yes
List Items Edit Form	
Inherit Value List	Yes
Show Only Row Source Valu	No

As mentioned, clicking on the form automatically opens the Combo Box Wizard. Select the first option: I want the combo box to get the values from another table or query and click Next >



Combo Box Wizard

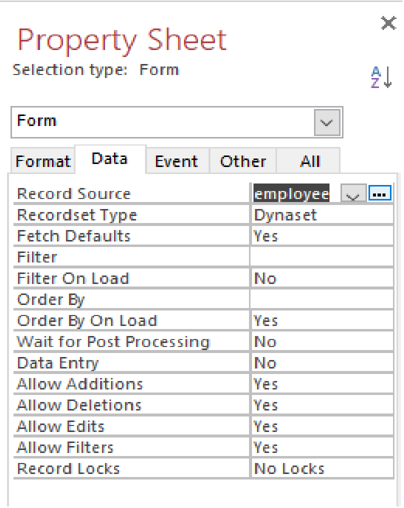
This wizard creates a combo box, which displays a list of values you can choose from. How do you want your combo box to get its values?

● I want the combo box to get the values from another table or query.

○ I will type in the values that I want.

○ Find a record on my form based on the value I selected in my combo box.

-The first selection is used to get values from another table.
-The second is used to type the values that you want.
-The third one is used to find my values form the selected field in the same table



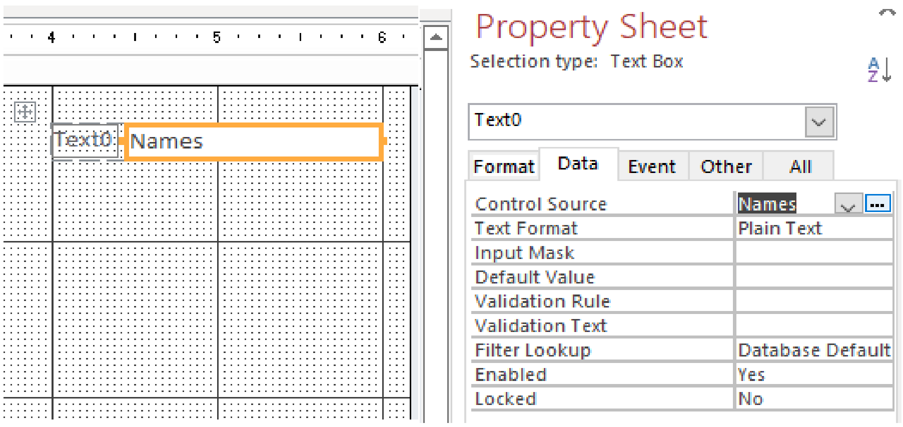
Property Sheet

Selection type: Form

Form

Format | Data | Event | Other | All

Record Source	employee
Recordset Type	Dynaset
Fetch Defaults	Yes
Filter	
Filter On Load	No
Order By	
Order By On Load	Yes
Wait for Post Processing	No
Data Entry	No
Allow Additions	Yes
Allow Deletions	Yes
Allow Edits	Yes
Allow Filters	Yes
Record Locks	No Locks



Property Sheet

Selection type: Text Box

Text0

Format | Data | Event | Other | All

Control Source	Names
Text Format	Plain Text
Input Mask	
Default Value	
Validation Rule	
Validation Text	
Filter Lookup	Database Default
Enabled	Yes
Locked	No

* First, you should click on the form, and from the property sheet tap data from the record source and select the table name.

* Second, after selecting the control tool, you can select the field name from the control source in the data tap of the property sheet.

4. Report

A report is an object in desktop databases designed for formatting, calculating, printing, and summarizing selected data.

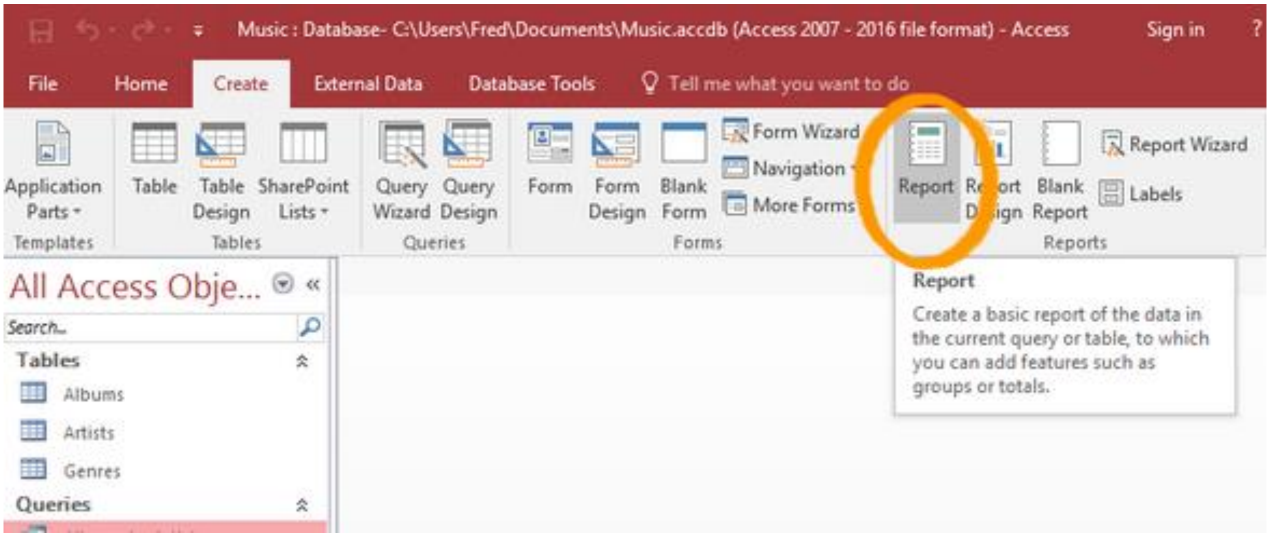
- You can view a report on your screen before you print it.
- If forms are for input purposes, then reports are for output.
- Anything you plan to print deserves a report, whether it is a list of names and
- addresses, a financial summary for a period, or a set of mailing labels.
- Reports are useful because they allow you to present components of your database
- in an easy-to-read format.
- You can even customize a report's appearance to make it visually appealing.
- Access offers you the ability to create a report from any table or query.

Access has several methods for creating reports. Any user can create a report quickly in Access, whether they're a novice or an expert.

you can create a report from a table or query with the click of a button. You can also use the Report Wizard to walk you through the process. And more experienced users can start with a blank report and build it exactly as they wish.

1. Create the Report using “Report”

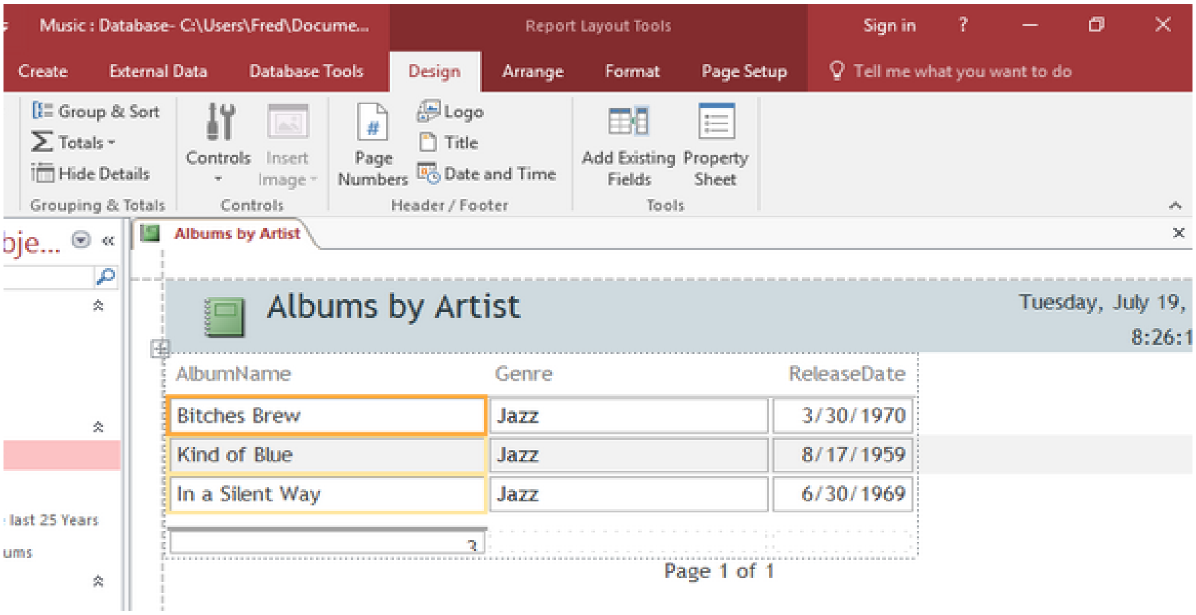
Ensure the report table is selected in the left navigation pane, click the Report button in the Ribbon from Create tap.



Report

Create a basic report of the data in the current query or table, to which you can add features such as groups or totals.

The Report displays the fields in columns, with the field names as column headers. It has basic formatting applied, and this can easily be changed using the various formatting options.



Albums by Artist

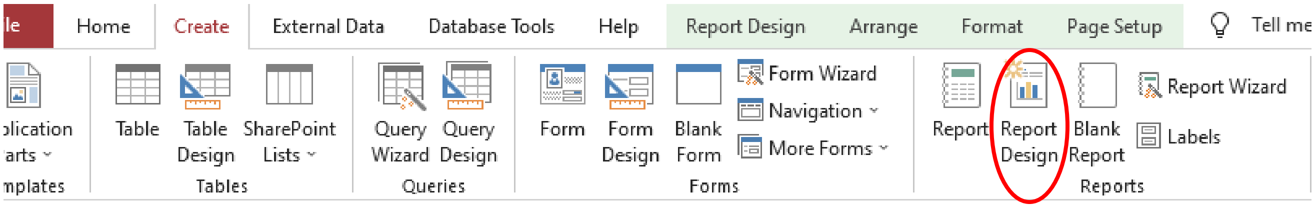
Tuesday, July 19,
8:26:1

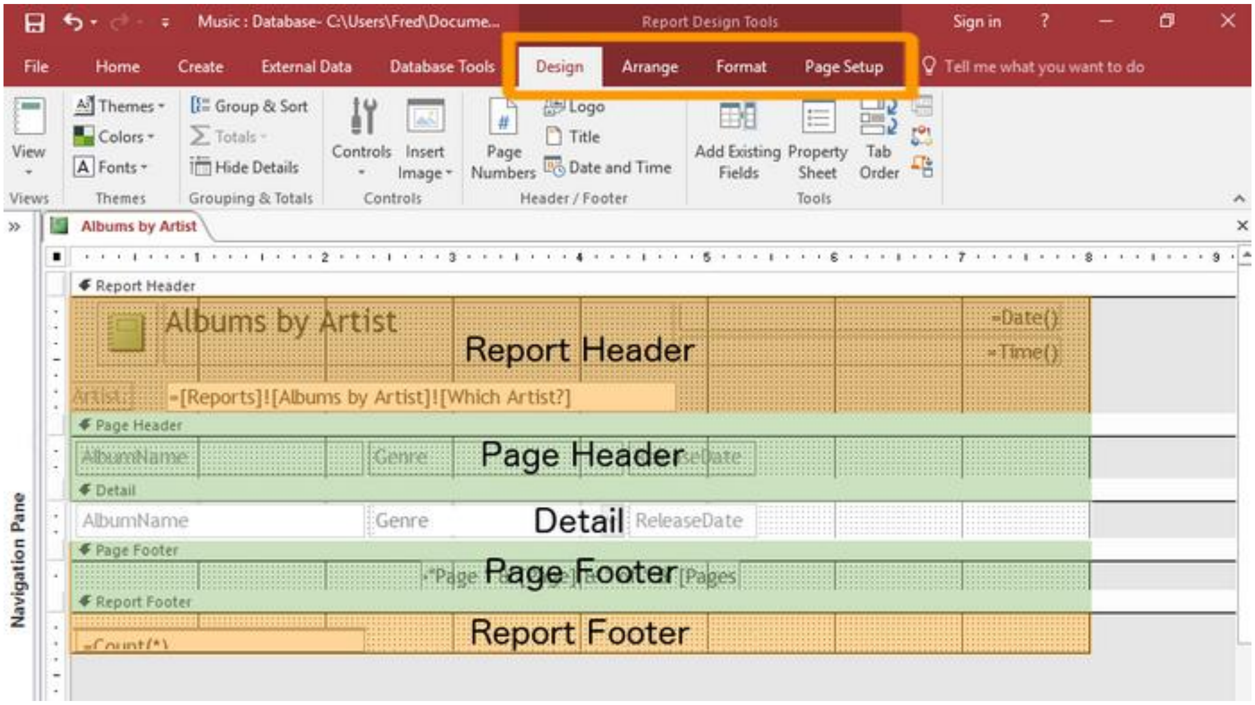
AlbumName	Genre	ReleaseDate
Bitches Brew	Jazz	3/30/1970
Kind of Blue	Jazz	8/17/1959
In a Silent Way	Jazz	6/30/1969

3

Page 1 of 1

2. Create the Report using "Report Design"





Reports can be designed and formatted in a similar way to forms. When you create a report, four new tabs appear in the Ribbon specifically for working with reports.

1- Report Header & Footer

The report header and footer are displayed only once in the report. The report header is displayed at the top of the first page, and the report footer is displayed at the bottom of the last page. So, in our example above, the artist's name will only be displayed once — at the top of the first page of the report.

2- Page Header & Footer

The page header and footer are displayed on every page of the report. If a report contains say, 10 pages, the page header, and page footer will be displayed on all 10 pages. So, in our example above, if we moved the artist's name to the page header, it would then be displayed on every page of the report.

3- Detail

The report detail section is where the actual data is displayed. This can run across many pages if required, and each page will continue where the previous left off.

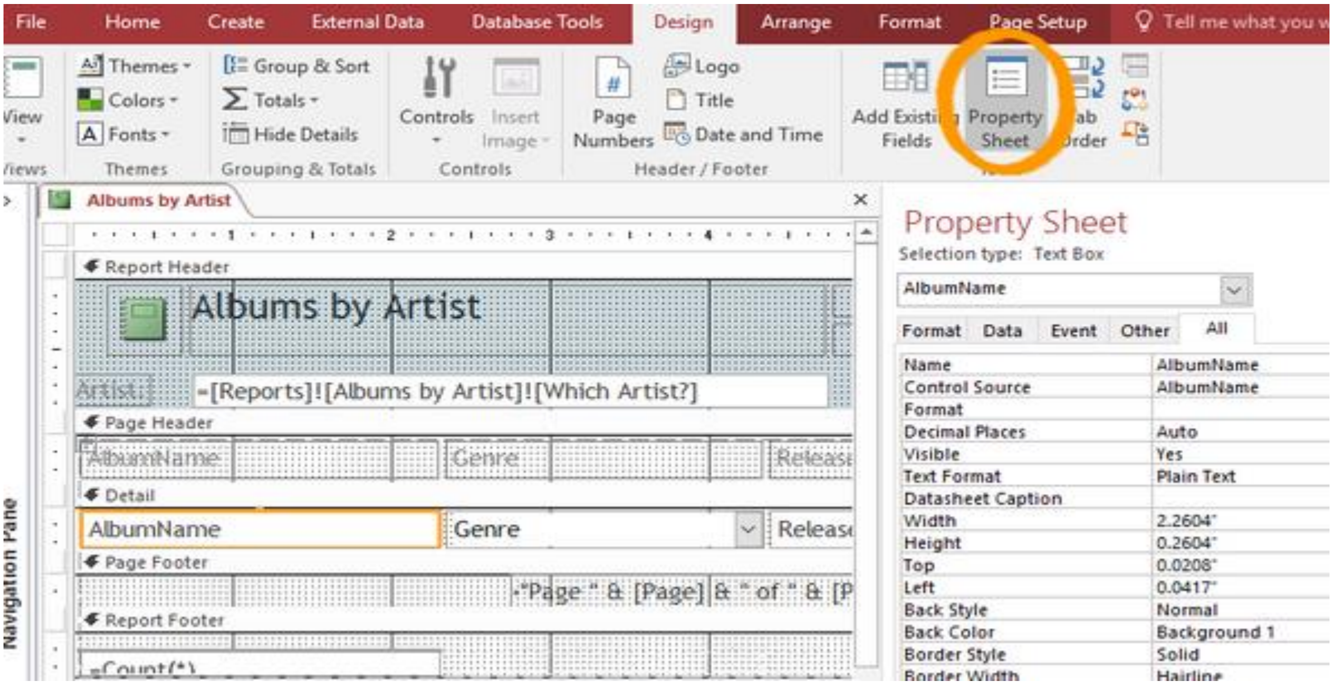
The Property Sheet

Screenshot of the Report in Design View with the Property Sheet open. As with other Access objects, you can use the Property Sheet to modify your report's properties. You can modify properties for the whole report, or just for selected elements.

You can change formatting properties (such as height, width, color, etc), data properties (such as where a control's data is sourced from), event properties (such as when a user clicks on the element), and more.

The Property Sheet can be accessed by clicking on Property Sheet on the Design tab of the Ribbon, or by right-clicking on an element and selecting Properties.

The Property Sheet only displays properties for the selected item.



Property Sheet

Selection type: Text Box

AlbumName

Format | Data | Event | Other | All

Name	AlbumName
Control Source	AlbumName
Format	
Decimal Places	Auto
Visible	Yes
Text Format	Plain Text
Datasheet Caption	
Width	2.2604"
Height	0.2604"
Top	0.0208"
Left	0.0417"
Back Style	Normal
Back Color	Background 1
Border Style	Solid
Border Width	Hairline

Report Views

Access provides four different views for reports.

As with all other Access objects, you can toggle the view using the buttons at the bottom right of the screen, or by using the View button on the Ribbon.

Design View

Allows you to view and design the report structure. Doesn't display live data. Instead, it displays any expressions that are used to generate content.

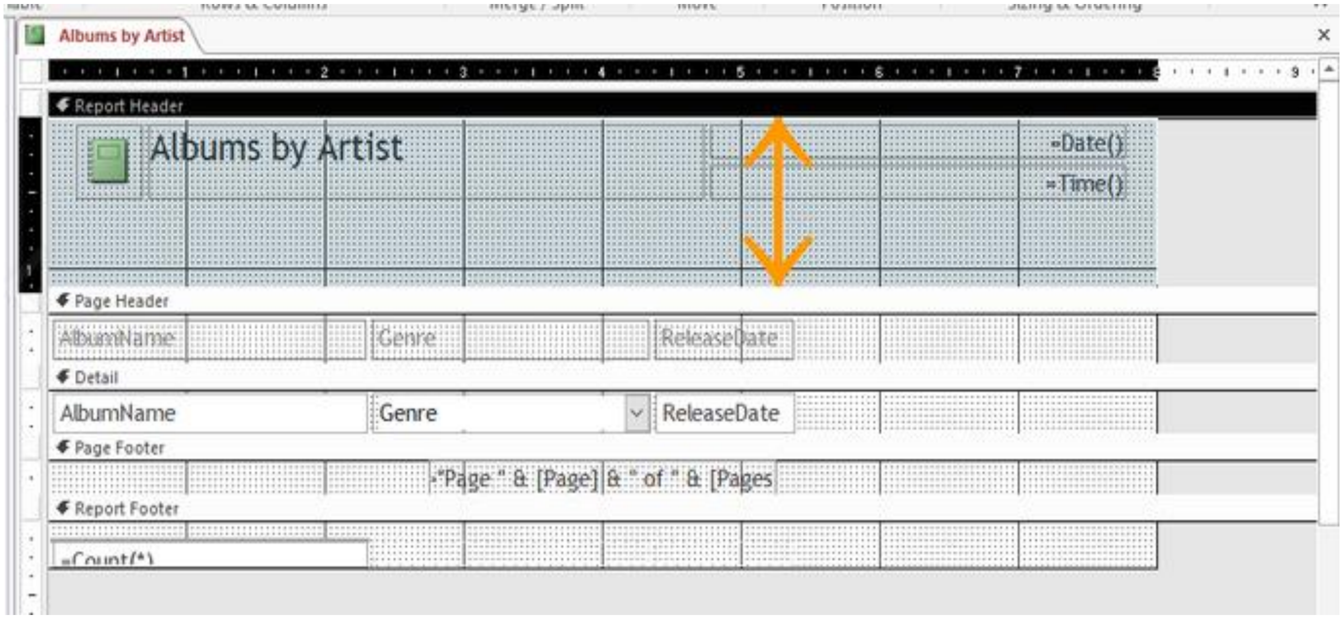
For example, when you look at the above report in Design View, you can see that various fields use in-built functions to display dynamic data. The Date() function is placed in the right corner of the header to display the current date (i.e. the date the report was generated). Similarly, the Time() function displays the current time. And the Count() function is used in the footer to display a count of the records returned in the report.

Report Design

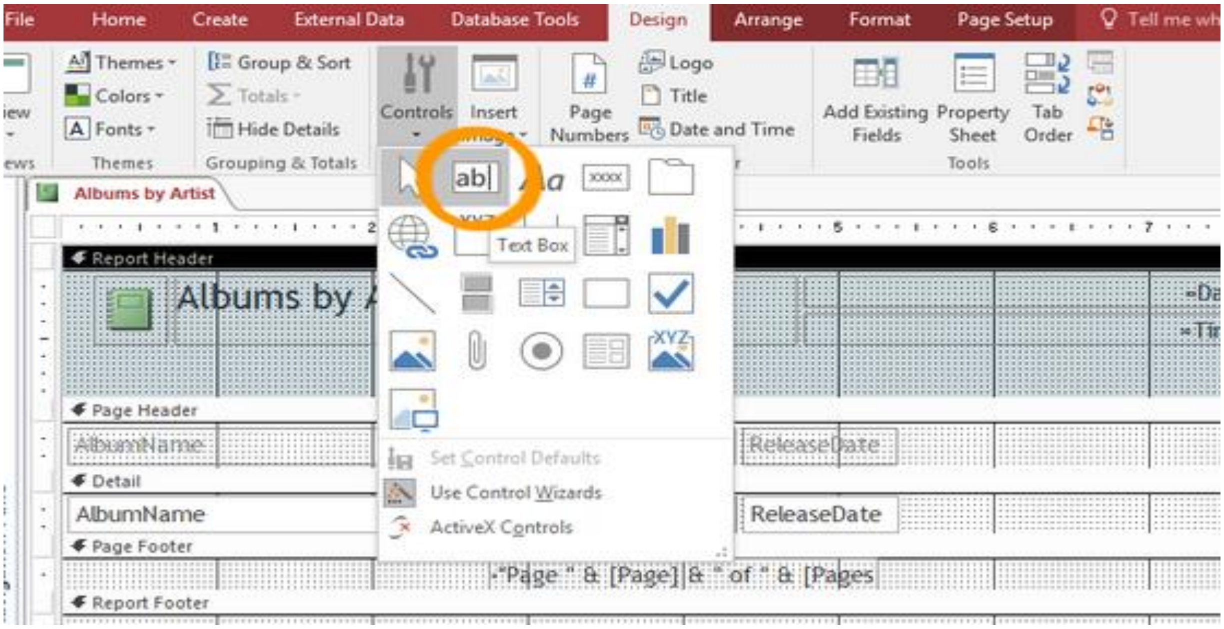
Select Report Design from create tap, then begin to design your report as you want.

Increase the Height of the Report Header, Click on the bottom edge of the report header and drag it down to make it a little higher.

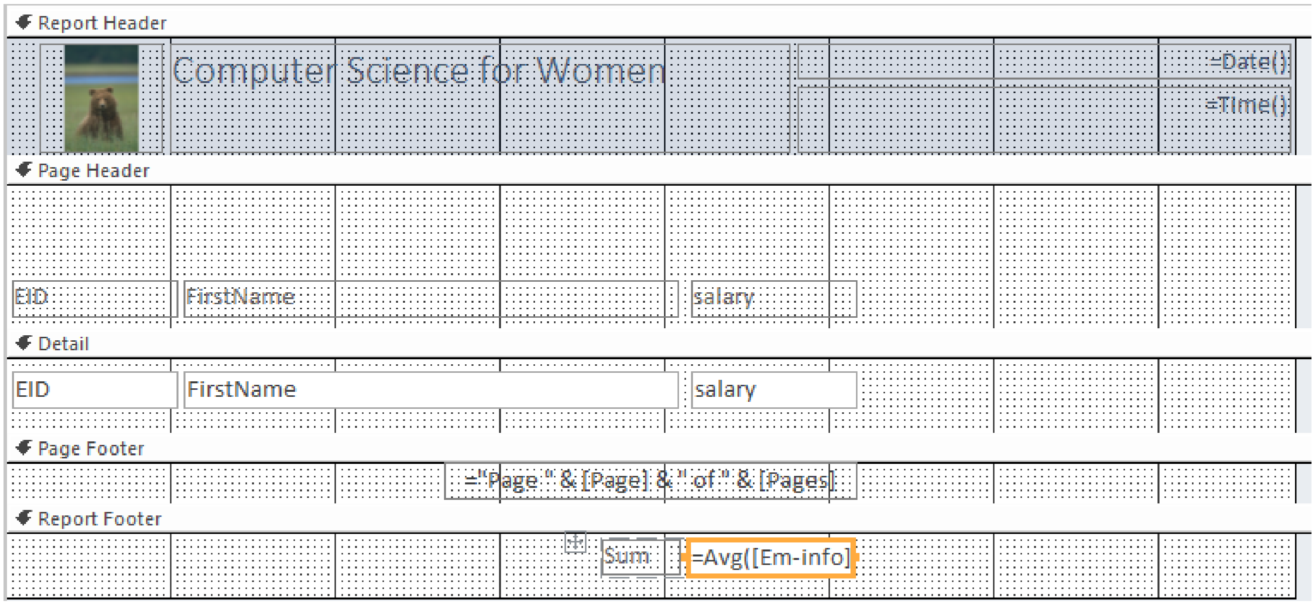
We're doing this so that we have enough room to display the the report title.



*Select a new Text Box Control: From the Design tab of the Ribbon, click the Controls button and select the text box option.



*Add the Text Box to the Report: You can click and drag to place the text box where you want, and at the size that you want. Click in the report header, under the title, and drag it so that it's wide enough to display the longest text that will be displayed. The text box will be inserted where you click.

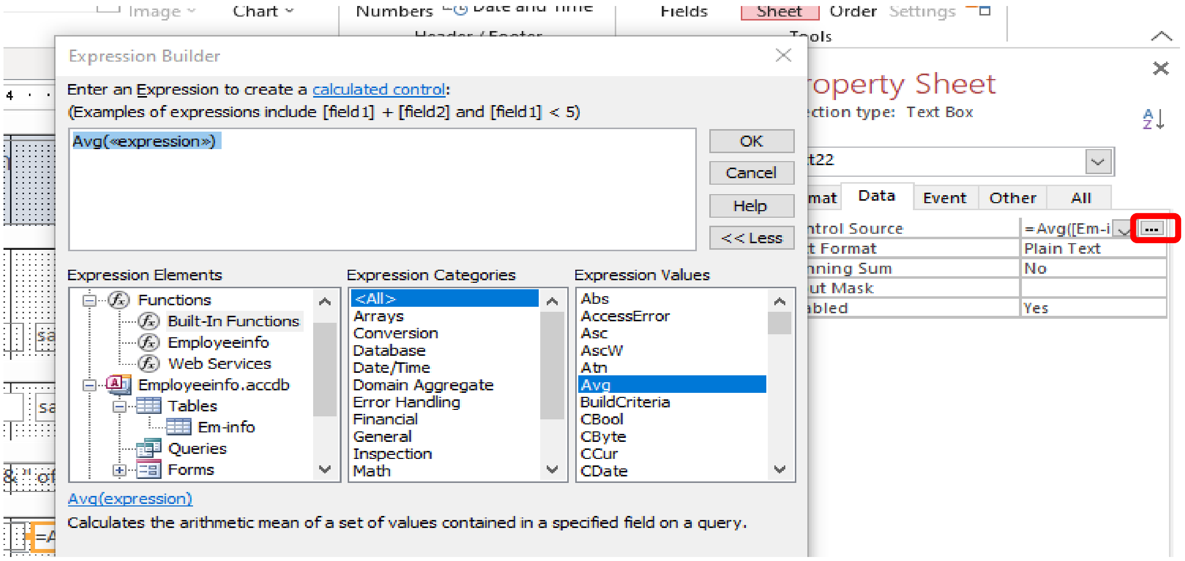


*We can add a function to our report, by clicking on the brows(….) as in the figure below.

*From functions select Bult-In-Functions to determine the function.

*Double-click on the expression in the function to add your field in that position as in the figure below.

*From the table choose the table name then select the field you want to apply the function on it.



Expression Builder

Enter an Expression to create a calculated control:

(Examples of expressions include [field1] + [field2] and [field1] < 5)

Avg(«expression»)

Avg(expression)

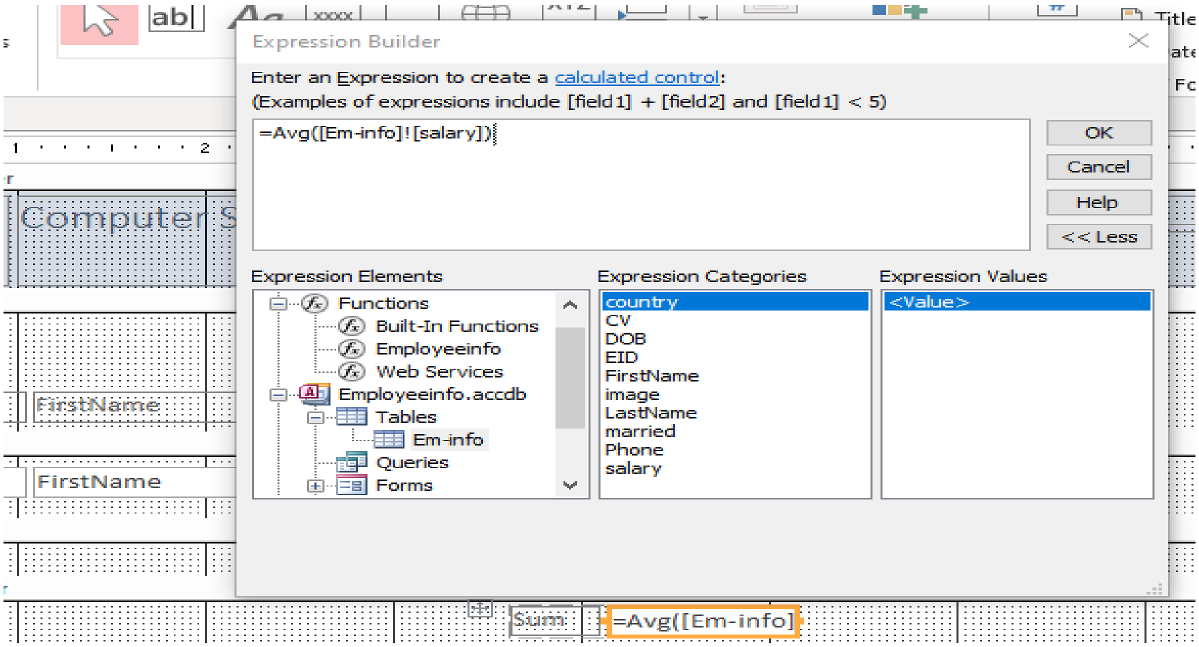
Calculates the arithmetic mean of a set of values contained in a specified field on a query.

Control Source: =Avg([Em-i

Text Format: Plain Text

Running Sum: No

Enabled: Yes



Expression Builder

Enter an Expression to create a calculated control:

(Examples of expressions include [field1] + [field2] and [field1] < 5)

=Avg([Em-info]![salary])

5. Macro

A macro is a set of actions that can be run automatically, and on demand. Macros can be used to automate repetitive tasks, which can save time and effort for the user.

Macros are usually configured to run whenever a particular event occurs. You specify what that event is when you create the macro.

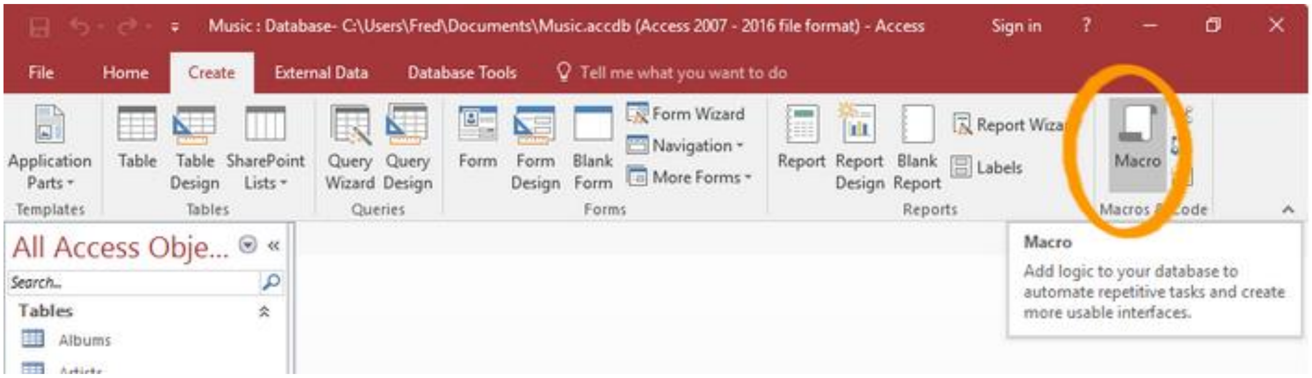
You can create macros for almost any task, or set of tasks. The great thing about macros is that they can run many tasks, one after the other, all at the click of a button. They can even be configured to run automatically as soon as the database is opened.

Create a Simple Macro

We'll now create a macro that opens a form automatically whenever the database is opened. We'll open the form that we created previously. This form enables the user to enter new albums into the database.

1.Create a new Macro

Click the Macro button from the Create tab on the Ribbon. This creates a blank macro, ready to be set up.

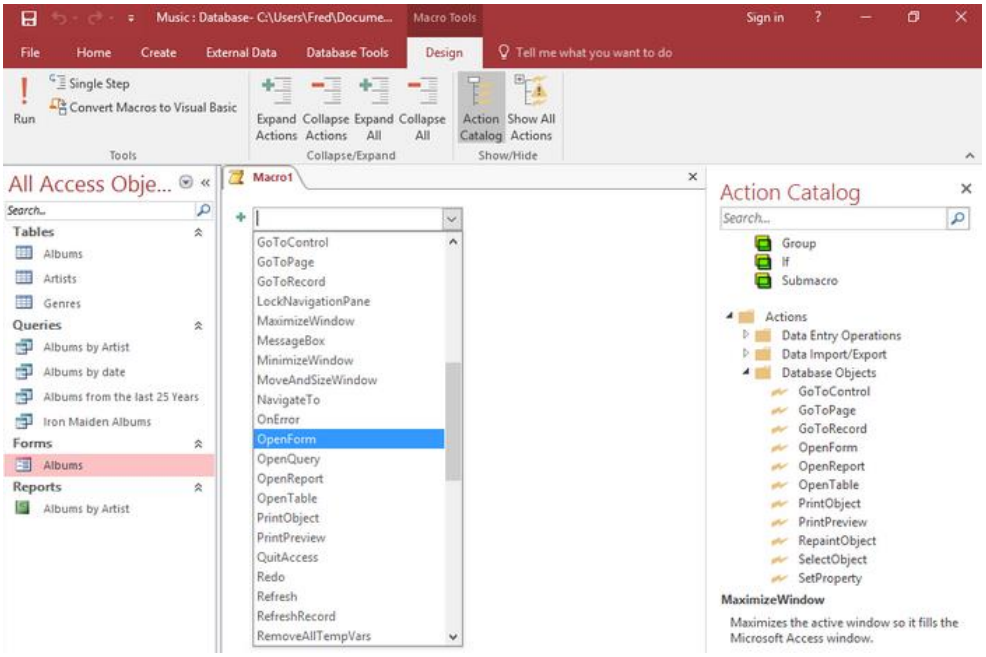


Macro

Add logic to your database to automate repetitive tasks and create more usable interfaces.

2.Select an Action

Select Open Form from the combo box.

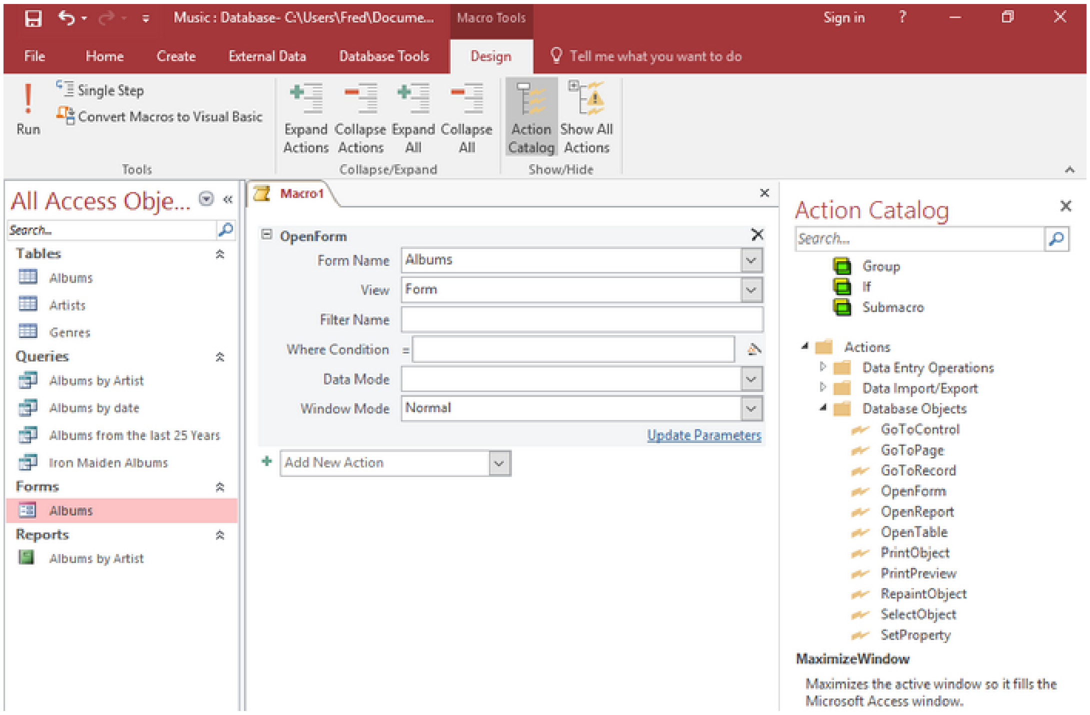


MaximizeWindow

Maximizes the active window so it fills the Microsoft Access window.

3.Configure the Action

Select Albums from the Form Name combo box. Set View to Form and Window Mode to Normal.



OpenForm

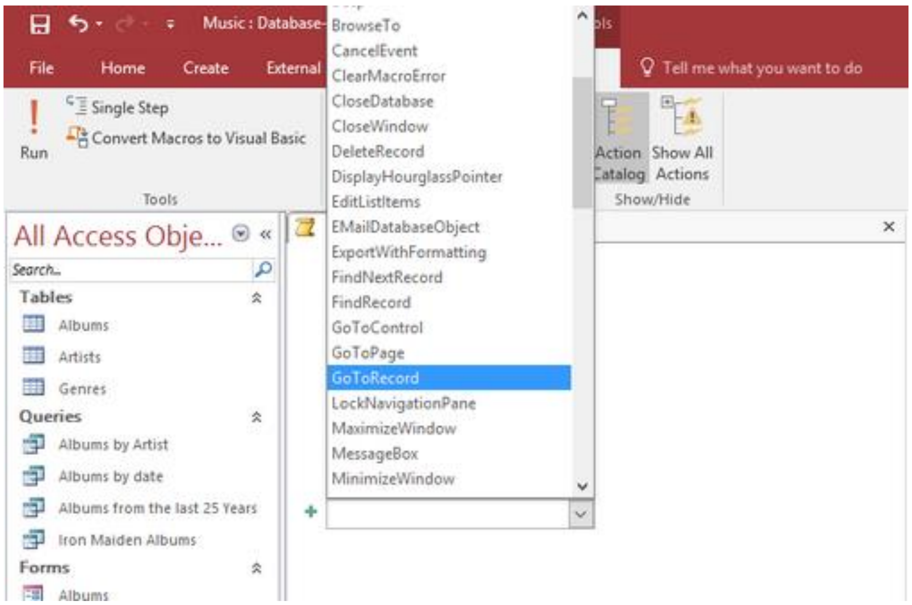
Form Name	Albums
View	Form
Filter Name	
Where Condition	=
Data Mode	
Window Mode	Normal

MaximizeWindow

Maximizes the active window so it fills the Microsoft Access window.

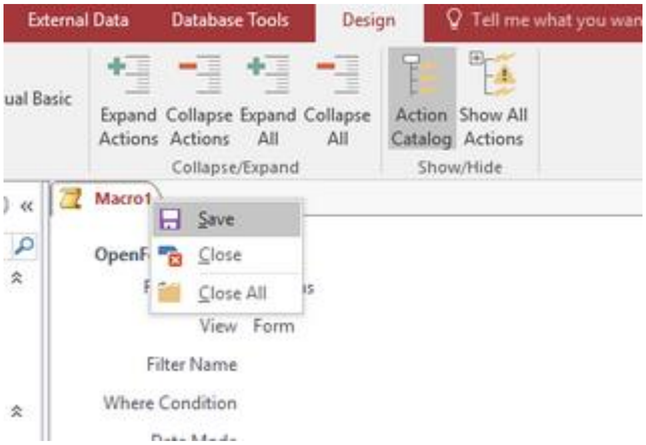
4.Select another Action

Select GoToRecord from the combo box.

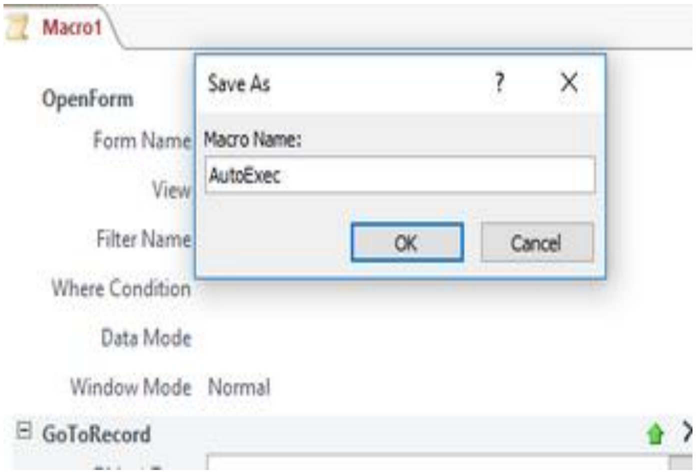


5. Save the Macro

Right-click on the macro's tab and select Save from the contextual menu. Name the macro AutoExec and click OK.



View Form



Macro Name: AutoExec

Window Mode Normal

6.Run the Macro

The only way to test the macro is to run it. One way to run it is to double-click on it in the left navigation pane.

But to test it properly, we need to make sure that it will run when the database is opened. Therefore, we'll need to close the database and open it.

6. Relationships

In relational database terms, a relationship is a situation where multiple tables can contain related data that is linked by a common field.

A relationship consists of a parent table and a child table. The child table references the parent table by having a field that matches a field in the parent table. The child's field is referred to as a foreign key. The parent's field is the primary key.

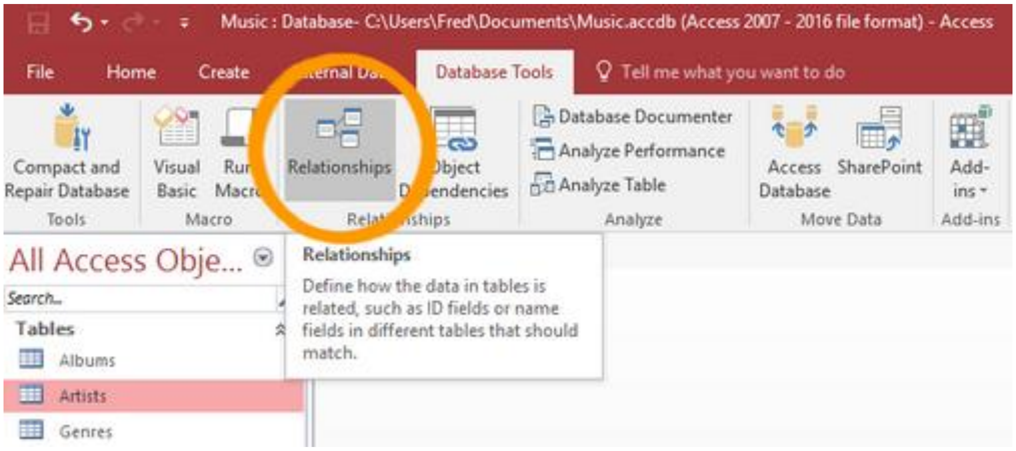
Any data entered into the child's foreign key field in a relationship must match a value from the parent's primary key field.

By ensuring that the foreign key's data matches the data in the primary key, we know that all records in the child table will have an associated record in the parent table.

So, we can create a one-to-many relationship between the class and the students' table. Our relationship will determine that an artist can have many albums, but an album can only belong to one artist.

1. Open the Relationship Dialog

Click Relationships from the Database Tools tab on the Ribbon. The Show Tables dialog box should appear. If it doesn't appear, click Show Tables.

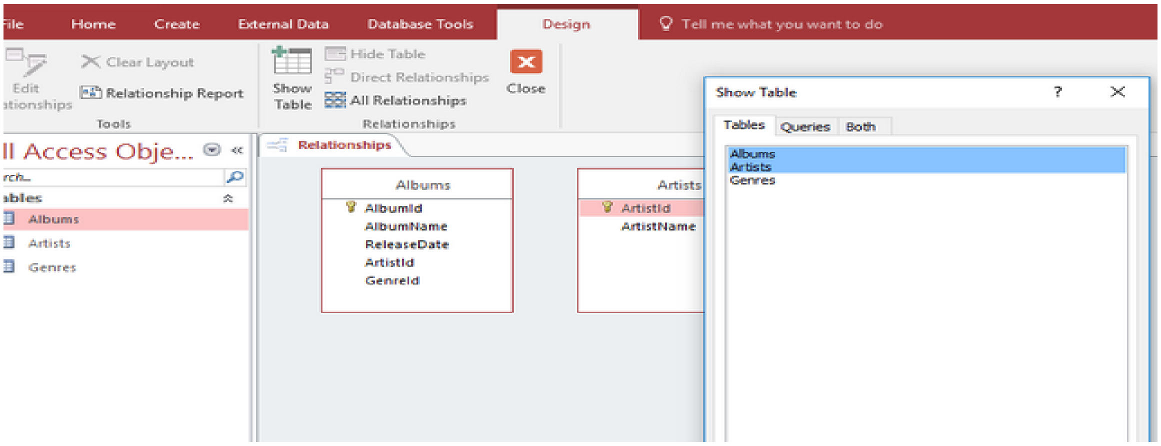


Relationships

Define how the data in tables is related, such as ID fields or name fields in different tables that should match.

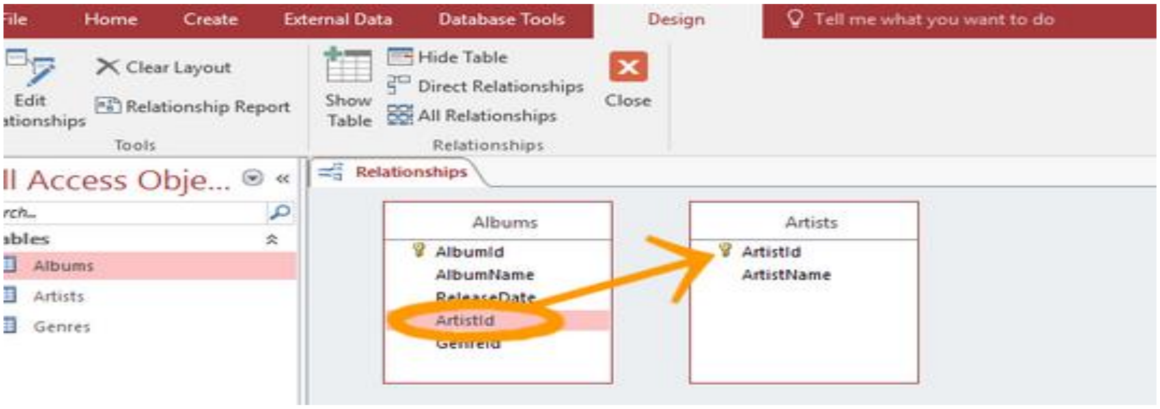
2.Select the Tables

Select both the Artists and Albums tables from the list and click Add. Click Close to close the dialog box.

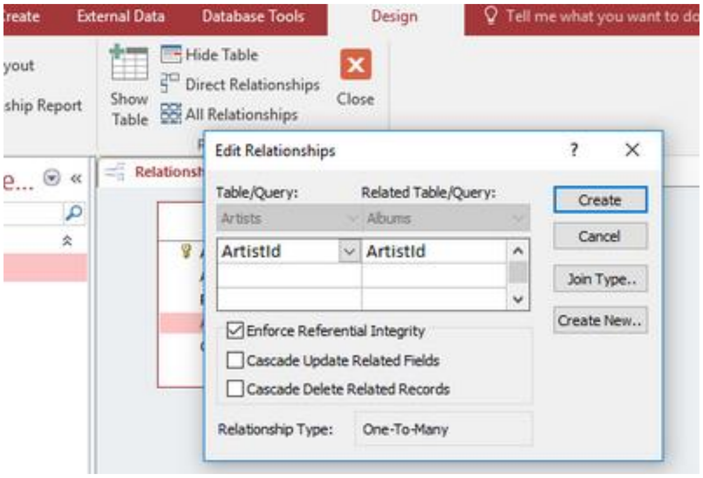


3.Create the Relationship

Click and drag the Albums.ArtistId field over the Artists.ArtistId field and release. The Edit Relationships dialog box appears.



4. Edit the Relationship



Edit Relationships

Table/Query: Artists | Related Table/Query: Albums

ArtistId | ArtistId

☑ Enforce Referential Integrity

□ Cascade Update Related Fields

□ Cascade Delete Related Records

Relationship Type: One-To-Many

Check Enforce Referential Integrity and click Create.:

1. The Enforce Referential Integrity option will ensure that child records cannot reference a non-existent parent. So, if a user tries to enter an album with an ArtistId that isn't in the Artists table, Access will prevent them from doing so. Access will also prevent the user from deleting an artist that has albums attached.

2. However, you can change how Access deals with deletes and data updates with the Cascade Update Related Fields and Cascade Delete Related Fields. Selecting these options will delete (or update) all related records whenever a primary record is deleted/updated.

The Relationship

A diagram now appears, representing the relationship. Save the relationship by pressing Ctrl+ S, right-clicking on the tab and clicking Save, or clicking the X icon to close the relationship.

So now that we've established the relationship, we can query data across both tables and get meaningful results. For example, we can now look up how many albums an artist has released. Or we could find out which artist released a given album. And more

Types of Relationships

There are three types of relationships:

One-to-One: A row in table A can have only one matching row in table B, and vice versa.

One-to-Many (or Many-to-One): A row in table A can have many matching rows in table B, but a row in table B can have only one matching row in table A.

Many-to-Many: A row in table A can have many matching rows in table B, and vice versa. This is achieved through the use of a third table (commonly called a junction table) that contains lookup data for both tables