

# A Basic introduction to Microsoft Access

*By Ojango J.M.K*

*Department of Animal Sciences, Egerton University, Njoro, Kenya and  
International Livestock Research Institute, Nairobi, Kenya*

Ms Access is a database management tool that enables one to have good command of data collected. The programme enables one to **retrieve, sort, summarize and report** results speedily and effectively. It can combine data from various files through creating relationships, and can make data entry more efficient and accurate.

Microsoft Access (MS Access) enables one to manage all important information from a single database file. Within the file, one can use:

- **Tables** to store your data.
- **Queries** to find and retrieve specific data of interest.
- **Forms** to view, add, and update data in tables.
- **Reports** to analyze or print data in a specific layout.
- **Data access pages** to view or update, the data.

In MS Access, data is stored once in one table, but can be viewed from multiple locations. When the data is updated in a Table, Query or Form, it is automatically updated everywhere it appears

## ***Establishment of Ms Access database***

All Ms Access databases files are saved with extension **.mdb**

A database should have a separate table for every major subject, such as **pedigree records**, **Production data** or **Treatment information**. Data should not be duplicated in multiple tables.

Microsoft Access provides three methods to create a database

- **Database Wizard** (though easy, the wizard offers limited options to customize the database)
- Using a **template** (This method works best if one can find and use a template that closely matches the specific requirements)
- **Creating a database directly** (**This is the most flexible method**, but it requires one to define each database element separately).

Create a new Access database **[Exercise 1.A]**

After creating a new database, this should be saved by the name which reflects the content of the database. Upon saving the database, the Ms Access database window opens with the Tables tab-active (i.e in such a way that the next activity should be to create a table).

## Creating a table

Tables are the **data storage facilities** in Ms Access. Each table contains rows called **records** and columns called **fields**.

**A record** is a collection of facts about a particular animal or event. Each record in a table should be unique. To distinguish one record from another, tables can contain a **primary key** field.

**A field** is a single kind of fact that may apply to each animal or event. For example, date of birth is a field in a table on animal information.

The fields in a database have **settings** that determine the

- type of data they can store,
- how the data is displayed,
- what can be done with the data.

For example, field settings can ensure that birth dates are entered with two numbers for the month, two numbers for the day, four numbers for the year, and slashes in between: 01/04/2006.

One important setting for fields is the **data type**, which could be **a number, text, currency, and date/time**. The data type limits and describes the kind of information in the field. The data type also determines the actions one can perform on a field and how much memory the data will use.

Fields also have **properties** that control the details of information inside them, including a **character length, a default value, and a validation rule** that makes sure the data meets certain criteria.

Tables may be created by either:

- Table wizard,
- Design view
- Entering data in a spreadsheet.

### Creating Tables using design view

Creation of a table by design view is **a user customized way** of making data storage tables. Each field in the design view of a table corresponds to a column in the datasheet view of a table.

Designing a table involves:

- Entering unique names of the columns of the table in the "**field name**" column of the design view. Names of fields and objects in Microsoft Access can be up to 64 characters long. They **can include** any combination of letters, numbers, spaces, and special characters except a period (.), an exclamation point (!), an accent grave (`), and brackets ([ ]). They also can't begin with leading spaces
- In the **Data Type** column, declare the type of data that will be entered in that column. The default is **Text**; or click in the **Data Type** column, click the arrow, and select the data type.
- In the lower portion of the window **Field properties**, can be further specified. Microsoft Access allows one to distinguish between two kinds of blank values: Null values and


zero-length strings The **Required** property determines whether users can leave a field blank, resulting in a **Null** value. The **AllowZeroLength** property determines whether Text or Memo fields can contain a zero-length string.

- Define a **Primary Key** field (You don't have to define a primary key, but it's usually a good idea. If you don't define a primary key, Microsoft Access asks if you want Access to create one for you when you save the table)
- Save the table by a name of your choice, (related to the information the table)
- Finally, close the design view to return to the **database window** where the columns of the table that you have just created are displayed.

#### Note:

To insert new fields within the table, in design view, click in the row below where you want to add the field, and then click **Insert Rows** on the toolbar.

To add the field to the end of the table, click in the first blank row.

To change from one view to another, make a selection on the **View** menu or click the arrow next to the **View** button  and choose from the list that appears.

To **view, enter, and change data** easily and directly in a table, create a 'form'.

#### Exercise 1. B

### Creating data entry forms

A **form** is a type of a database object that is primarily used to enter or display data in a database. Most forms are bound to one or more **tables** and **queries** in the database. A form's record source refers to the **fields in the underlying tables and queries**.

A form:

- Focuses on one record at a time
- Can display fields from more than one table
- Can also display pictures and other objects
- Can contain a button that prints, opens other objects, or otherwise automates tasks

Data entry forms can be created either using a form wizard or in the design view. Once a table with fields is available, it is **easy to create a form using the wizard**:

- o Click on the forms tab
- o Double click on the create form by using a wizard.
- o This will lead you through a series of steps until you finish creating the form.

The form created is linked to the table and information entered in the fields of the form is stored in the table.

A form **need not contain all the fields** from each of the tables or queries that it is based on.

When you open a form, Microsoft Access retrieves the data from one or more tables, and displays it on the screen with the chosen layout in the Form Wizard, or with the layout that you created on your own in design view.

## Exercise 1.C

### Queries

A **query** is a derived item in the database meant to answer specific questions that relate to the information in the database. Queries are handy during data processing.

To **find and retrieve just the data that meets conditions** that you specify, including data from multiple tables, **create a query**. A query can also update or delete multiple records at the same time, and perform predefined or custom calculations on your data.

**A query requests data from the database.** At its simplest, a query merely fetches all data from a single table. But as you create more complex (and more typical) queries, you can assemble exactly the data you want (*i.e. unique sets of data that you require at any given time*). Queries can also be used to execute mathematical and logical functions to obtain certain information in the database.

Queries are **derived from and linked to tables or other queries**. (Due to these linkages, they tend to largely inflate the size of the database and should thus only be used to execute the intended functions, and stored only if updated information is to be retrieved)

There are various types of queries for different uses:


- **Select queries**- used for extracting specific information from a large multi-information table. They can also be helpful in merging related information from different tables.
- **Make-Table queries**- used for making sub tables from the main table(s) and queries.
- **Update queries**- important in adding information in the fields of a Table.
- **Append queries**- used to copy records from one table/ query to another.
- **Delete query**- to **PERMANENTLY** remove unwanted content from the table. **NOTE:** delete query should not be used unless one surely will not require the information to be deleted.

To run a simple **Select query**: In the Database window, click **Queries** under **Objects**. Click the query you want to open. Click **Open** on the Database window toolbar.

**Caution:** It's a good idea to make a copy of the data you are changing or moving in an **action query**, in case you need to restore the data to its original state after running the action query

In Design view,

- Choose the tables or existing queries that contain the fields to use
- Select and drag those fields to a grid (The fields can come from just one table, or from multiple tables, the fields specified for a query control the data that the query retrieves)
- Specify criteria and other settings, such as whether to sort the results (

To test a new query in Design view. Just click the **Run** button  on the **Query Design** toolbar. This will switch you to the results view.

If you don't see the results you want, click the **Design** button  to return to Design view.

**Note:** To stop a query in progress (after you start it), press “CTRL+BREAK”

## **Exercise 2.A**

A query result can include **data from multiple tables**. To combine data from tables, you use the Join operation

The **Join operation** matches rows of one table with rows of another table, based on values in those rows.

## ***Relational Databases***

A relational database is **one whose components (tables, forms, queries etc) are related (linked)**. The linkages between database components are created by making relationship links between them. The relationship can be between:

- One component and another (one-to-one relationship),
- One component related to several other components (one-to-many)
- Several database components (many-to-many).


Creation of relationships between database components reduces data redundancy and enhances ease of access of the information.


### ***Creation of relationships between database components***

When you create a relationship between tables, the related fields **don't necessarily have to have the same names**. However, related fields **must have the same data type**. UNLESS the **Primary key** field is an AutoNumber field.

An AutoNumber field can be matched with a Number field only if the **FieldSize** property of both of the matching fields is the same. For example, you can match an AutoNumber field and a Number field if the **FieldSize** property of both fields is **Long Integer**. Even when both matching fields are Number fields, they must have exactly the same **FieldSize** property setting.

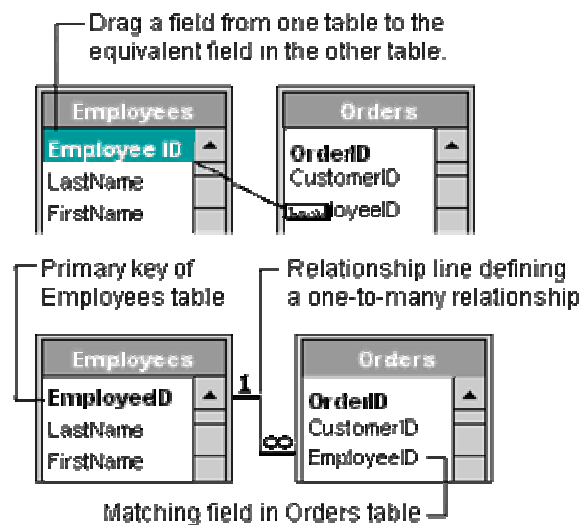
### **Defining a one-to-many or a one-to-one relationship**

1. Close any tables you have open. You **can't** create or modify relationships between open tables.
2. Press **F11** to switch to the Database Window.
3. Click **Relationships**  on the toolbar.
4. If you haven't yet defined any relationships in your database, the **Show Table** dialog box is automatically displayed.

If you need to add the tables you want to relate and the **Show Table** dialog box isn't displayed, click **Show Table**  on the toolbar.

5. Double-click the names of the tables you want to relate, and then close the **Show Table** dialog box. To create a relationship between a table and itself, add that table twice.
6. Drag the field that you want to relate from one table to the related field in the other table.

In most cases, you drag the primary key field (which is displayed in bold text) from one table to a similar field (often with the same name) called the foreign key in the other table (see *the illustration below*).



7. The **Edit Relationships** dialog box is displayed. Check the field names displayed in the two columns to ensure they are correct. You can change them if necessary.

Set the relationship options if necessary.

8. Click the **Create** button to create the relationship.
9. Repeat steps 5 through 8 for each pair of tables you want to relate.

When you close the **Relationships window**, Microsoft Access asks if you want to save the layout. Whether you save the layout or not, the relationships you create are saved in the database.

**Note:** You can create relationships using **QUERIES** as well as **TABLES**. However, **referential integrity** isn't enforced with queries.

### Exercise 2.B

#### .NOTE

Data can be imported into and exported out of access from various other programmes, eg ascii files, MS Excel, Dbase, .rtf etc.

## Reports

Reports provide a means of organizing and summarizing data. Reports are often used to present an overview highlighting main points and trends. A report can be a simple list, a status report or a monthly production report. With reports, one can prepare:

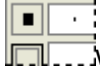
- Mailing labels for various producers
- Produce a directory
- Prepare invoices
- Present data summaries.

A report is made from the data available. There are several ways of preparing a report:

- **AutoReport:** the quickest way to create a report, but gives least control over the report's structure and appearance. This automatically contains all the fields in its data source, whether table or query.
- **Report Wizard:** asks you questions and creates a report based on your answers. The Report Wizard asks which tables or queries the report will be based on, and which fields to use from those data sources. It also asks whether the data is to be grouped, and how it should be sorted and summarized.
- **Creating a Report in Design view:** Gives one control, right from the start. Design view provides you with a toolbox from which you drag selected controls and arrange them on a grid.

To see the report as it will appear when printed, it should be viewed in Print Preview. Different properties may be set for a report, to change how it is viewed or printed.

For example, a report's Page Header property determines whether the page header is printed on the page that begins with the Report Header.

To view or change a report's properties, just double-click the **report selector**  where the rulers meet.

To see more details about a property, click that property and then press F1.

To view a control's properties, double-click the control.

Further information is available through the help section within access, and at <http://office.microsoft.com/en-gb/training/CR061829401033.aspx>



## Terms used in Access

**Action query:** A query that copies or changes data. Action queries include append, delete, make-table, and update queries. They are identified by an exclamation point (!) next to their names in the Database window

**Crosstab query:** A query that calculates a sum, average, count, or other type of total on records, and then groups the result by two types of information: one down the left side of the datasheet and the other across the top

**Database window:** The window that appears when you open an Access database or an Access project. It displays shortcuts for creating new database objects and opening existing objects

**Datasheet view:** A window that displays data from a table, form, query, view, or stored procedure in a row-and-column format. In Datasheet view, you can edit fields, add and delete data, and search for data

**Data type:** The characteristic of a field that determines what type of data it can hold. Data types include Boolean, Integer, Long, Currency, Single, Double, Date, String, and Variant

**Design view:** A window that shows the design of these database objects: Tables, queries, forms, reports, macros and data access pages. In design view one can create new database objects and modify the design of existing ones.

**Field data type:** A characteristic of a field that determines what kind of data it can store. For example, a field whose data type is Text can store data consisting of either text or number characters, but a Number field can store only numerical data

**Field properties** are a set of characteristics that provide additional control over how the data in a field is stored, entered, or displayed. Which properties are available depends on a field's data type

**Foreign key:** One or more table fields (columns) that refer to the primary key field or fields in another table. A foreign key indicates how the tables are related

**Primary key:** One or more fields (columns) whose values ***uniquely identify*** each record in a table. A primary key ***cannot allow Null values*** and must always have a unique index. A primary key is used to relate a table to foreign keys in other tables

**Relationship:** an association that is established between common fields (columns) in two tables or other components of a database

**Relationships window:** A window in which you view, create, and modify relationships between tables and queries

**Referential integrity:** Rules that you follow to preserve the defined relationships between tables when you enter or delete records

**Select query:** A query that asks a question about the data stored in your tables and returns a result set in the form of a datasheet, without changing the data

# Microsoft Access Practical

## Exercise 1

1.A. Create a blank access database and save it under a name of your choice on the computer

1. Click **New**  on the toolbar.
2. In the **New File** task pane, under **New**, click **Blank Database**.
3. In the **File New Database** dialog box, specify a name and location for the database, and then click **Create**

1.B. Create **2 tables** in Design view, where you can add fields, define how each field appears or handles data that would be suitable for entry of the table of data given

- Table 1. Pedigree data
- Table 2. Production data

| 1 Pedigree |        |        |       |
|------------|--------|--------|-------|
| Record No  | AnimID | SireID | DamID |

| 2 Production data |        |       |      |         |        |     |         |      |
|-------------------|--------|-------|------|---------|--------|-----|---------|------|
| Record No         | AnimID | Breed | Herd | Dt Brth | BrthWt | Sex | Dt Wt 1 | Wt 1 |

1. C. Create **3 forms** based on the data given (page 3 of handout)
- Form number 1: To ease entry of Pedigree data
  - Form number 2: To enable entry of information on dates
  - Form number 3: To enable entry of performance information

**Key in data given using the forms designed**

| 1 Pedigree |        |        |       |
|------------|--------|--------|-------|
| Record No  | AnimID | SireID | DamID |
| 1          | 1E54   | 6C211  | 5C66  |
| 2          | 1E40   | 653    | 6C36  |
| 3          | 1E52   | 6C211  | 4C36  |
| 4          | 1E65   | 6C96   | 5C170 |
| 5          | 1E34   | 653    | 2C51  |
| 6          | 1E43   | 5C198  | 3C59  |
| 7          | 1E55   | 6C211  | 3C76  |
| 8          | 1E60   | 6C211  | 2C88  |
| 9          | 1E61   | 6C211  | 4C25  |
| 10         | 1E67   | 6C211  | 5C122 |

| 2 Production data |        |       |      |            |        |     |            |      |
|-------------------|--------|-------|------|------------|--------|-----|------------|------|
| Record No         | AnimID | Breed | Herd | Dt Brth    | BrthWt | Sex | Dt Wt 1    | Wt 1 |
| 1                 | 1E54   | B     | 2    | 11/07/2001 | 32     | M   | 27/11/2001 | 112  |
| 2                 | 1E40   | B     | 2    | 15/06/2001 | 32     | M   | 27/11/2001 | 150  |
| 3                 | 1E52   | B     | 2    | 06/07/2001 | 30     | F   | 26/11/2001 | 114  |
| 4                 | 1E65   | B     | 2    | 19/07/2001 | 28     | F   | 27/11/2001 | 115  |
| 5                 | 1E34   | B     | 2    | 12/06/2001 | 30     | M   | 30/10/2001 | 118  |
| 6                 | 1E43   | B     | 2    | 18/06/2001 | 28     | M   | 26/11/2001 | 100  |
| 7                 | 1E55   | B     | 2    | 11/07/2001 | 28     | M   | 26/11/2001 | 106  |
| 8                 | 1E60   | B     | 3    | 15/07/2001 | 30     | M   | 26/11/2001 | 120  |
| 9                 | 1E61   | B     | 3    | 15/07/2001 | 30     | M   | 26/11/2001 | 100  |
| 10                | 1E67   | B     | 3    | 21/07/2001 | 28     | M   | 26/11/2001 | 110  |

## **Exercise 2**

**2.A.** Create a **Query** to:

- Extract years from data
- Extract months
- Create a Herd-Year-Season effect

**Create a new table from the results of the query**

**2.B.** Create a query with **relationships** using information from table created in Exercise 2.A. and Table 2-Pedigree data.

Create a table with both production, calculated and pedigree information that can be used for data analysis

For More information, and an example database, use **Access help**, where there is a database called **Northwind**, given as an example.

