Database Management System using LibreOffice Base

Introduction

Management of data is of prime importance for any organisation. The ease of entering data, its efficient storage, accurate and fast retrieval is the primary objective of any database management system. In this unit you will learn about various methodologies of data management using LibreOffice Base, which is an open source application.

The first chapter starts with the general concepts of database management system, different database model with the focus on relational database. It explains the concept of primary key and foreign key and the various database objects of RDBMS.

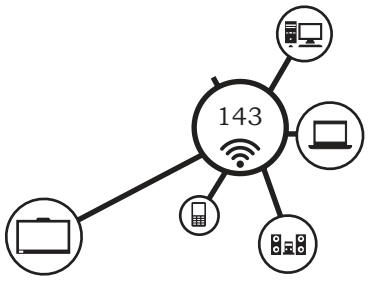
In the second chapter you will learn about LibreOffice Base application interface, how to start and use it, how to create database and tables, enter and edit the data in the table. Different data types used by LibreOffice Base have also been discussed.

After creating a table, you can edit or delete the table and set up relations between the tables to control data redundancy and inconsistency for proper maintenance of a database. In the third chapter, you will learn how

the changes are reflected in the related tables after setting up relations.

In the fourth chapter, you will learn to retrieve the data from the database as per your requirement by providing the desired specifications, which is called as Query. Depending upon given specifications, the specific records are searched form the database and then displayed in the desired manner.

In the last chapter, you will learn to create Forms and Reports using LibreOffice Base. Form is the user friendly data entry screen that allows to enter the data in the table easily by any user. The Report feature helps to present the retrieved data in a user friendly, understandable and formatted manner.



CHAPTER 8

Introduction to Database Management System

INTRODUCTION

Databases and database systems are essential parts of our life. We have been interacting with databases since a time. Recall the process of looking for a word in a dictionary or finding the telephone number of a friend from the telephone directory. With the advancement in ever changing technology, computerized databases are being used to store, manipulate, and manage the database. Today, we use databases in almost all spheres of life. When we go to book railway tickets, to search for a book in a library, to get the salary details, to get the balance amount after withdrawal of money from the ATM and so on, this list can run into several pages. Data being stored in databases can be of varied types like text, images, audio and video. This data is then stored and/or processed so as to get meaningful information.

Data and Information

The raw facts constitutes **data**. The facts may be related to any person, place, activity or things. It may be stored in the form of text, graphics, audio or video. This data must be processed by any computing machine in a proper way to generate the useful and meaningful information. The examples of data are marks scored by the students, weights, prices, costs, numbers of items sold, employee names, product names, addresses, tax codes, registration, marks etc.

Information is the processed or organized form of data. If data is not correct or accurate, the information obtained by processing such data may not be correct. For example, marks obtained by students and their roll numbers is the data, while the report card/sheet is the

information. Other forms of information are pay-slips, schedules, worksheet, bar charts, invoices, account returns etc. Similarly the temperature recorded is data which can be processed to find out the maximum or minimum temperature of day and night. Also this data is generally organised in tabular form as shown in the Table 8.1.

Table 8.1: Data Stored in Tabular Form

Day	Day Temperature	Night Temperature
Sunday	35	15
Monday	37	17
Tuesday	36	16
Wednesday	34	15
Thursday	35	17
Friday	35	16
Saturday	33	14

Databases and DBMS

A database is a collection of logically related data items stored in an organised manner. The information being stored in a database can be added, modified, deleted or displayed according to the requirements of the user. The software that is used to create, update and retrieve data is known as database management system (DBMS). It facilitates planning and maintenance of the database for the user. Some of the common examples of DBMS are MS Access, Open Office or LibreOffice Base, Oracle, Ingress, MySQL. In this book, we will use LibreOffice Base version 6.4 to learn various aspects of DBMS.

We all know that it is easy to remember names of our friends, but it is difficult to retain their phone numbers in our memory. If details like names and phone numbers are stored in a diary or mobile phone in an orderly manner, then it becomes easy to retrieve the phone number of a required person. Thus, the phone book can be considered a simple database and to manage this database electronically, we will require a database management system or a DBMS.

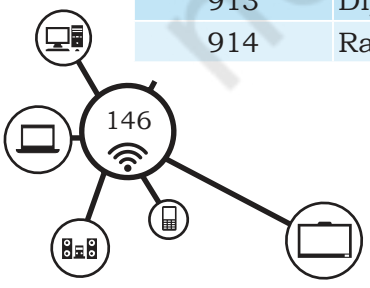
Let us discuss some of the advantages of DBMS:

- Organised Storage – The data in the database is stored in an organised manner, so that retrieval of the required data is fast and accurate.
- Data Analysis – A database helps in analysis of data based on certain criteria. It is easy to find out maximum or minimum value, average or mean using a database.
- Data Sharing – If the same data set is required for different applications then the database can be shared with other applications. Hence using a database means making once and using it repeatedly for multiple applications.
- Minimal Data Redundancy – In the event of requiring the same data field in several tables the data field might get repeated in number of tables. This is called as data redundancy. This can be reduced by using DBMS tools.
- Data Consistency – By minimising data redundancy, chances of inconsistent data being stored is reduced. For example, it should not happen that the name of the student is changed in one table and not in another. Such inconsistency is reduced by using a DBMS.

Let us consider a situation where, for example, Murugan looks after the data management of ABC School. There are two tables in his database– Admission table and Library table. 'Ram Lal Kumar' wants to change his name to 'Ram Kumar' has recently shifted his house. So, he hands over the application for the same to Murugan. In the absence of a DBMS, Murugan has to change the name of the student separately in the admission table and also in the library table. But as we can set a relation between the two tables using a DBMS, any change in one table is automatically reflected in all the related tables. There is no need to add another record with the changed name. The same record will be updated. Hence, this reduces the chances of data redundancy and inconsistency.

Admission Table

Roll Number	Student Name	Class	Date of Birth	Date of Admission
913	Dipak Kumar	10	06/04/2004	25/06/2018
914	Ram Lal Kumar	10	01/03/2004	24/06/2018



Library Table

Student Name	Name of the Book	Date of Borrow	Date of Return
Dipak Kumar	IT	10/12/2018	25/12/2018
Ram Lal Kumar	Science	11/11/2018	24/11/2018
Sham Lal	Mathematics	12/11/2018	30/12/2018

- Increases Efficiency – Since database tables are properly organised, saving, reading and searching data can be carried out efficiently.
- Increases Accuracy – Since data redundancy and inconsistency can be minimised in a database, the data is retrieved accurately from the database.
- Increases Validity – Properties of different data fields can be assigned when a database is planned. So whether or not valid data is being entered can be checked at the data entry stage. This increases the validity of the database. For instance, we may set up a restriction while creating the table that the amount of fees being entered should be more than 1000.
- Security – Unauthorised access can be controlled by assigning passwords to the users. The data might be translated in such a manner that unauthorised users are not able to read it. This is known as encryption. Both these measures increase the security of the database.

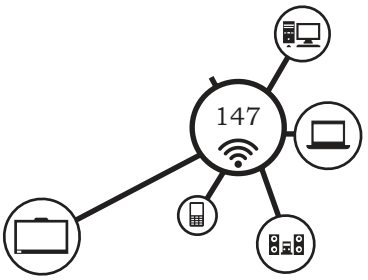
Data Models

A database can be designed in different ways depending on the data being stored. This structure of database is known as **data model** that describes the manner in which data will be stored and retrieved.

A data model consists of components for describing the data, relationships among them and the constraints that hold data. There are different data models such as hierarchical data model, network data model and relational data model.

Hierarchical Data Model

In this model the data is organized into a tree like structure. The data is stored in the form of records. A record is a collection of fields and its data values.



All these records are linked to each other at various levels, thereby forming a hierarchy. For example, in Fig. 8.1 the data of a company is stored using a hierarchical data model.

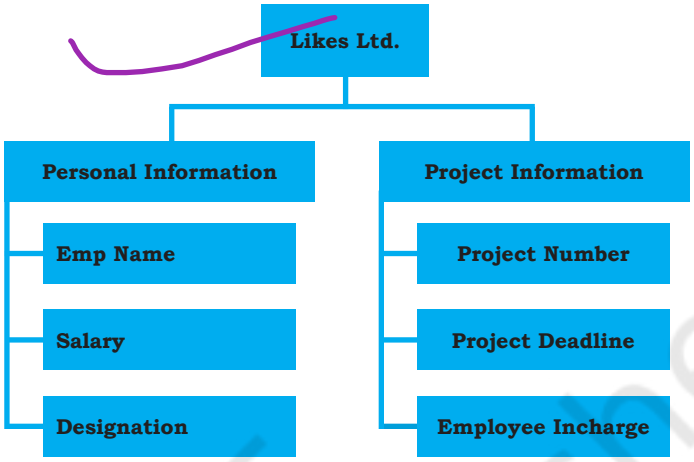


Fig. 8.1: Hierarchical Data Model

Network Data Model

In this model, multiple records are linked to same master file. It is also considered as an inverted tree where master is present in the bottom of the tree and the branches contain information linked to the master. In Fig. 8.2, the data of the company is represented using the network data model.

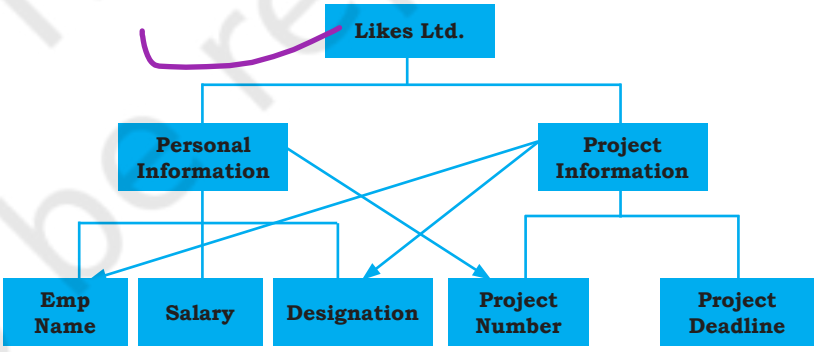


Fig. 8.2: Network model

Relational Data Model

This data model is based on the principle of setting relationships between two or more tables of the same database. It is the most commonly used database model. Let us study about this model in detail.

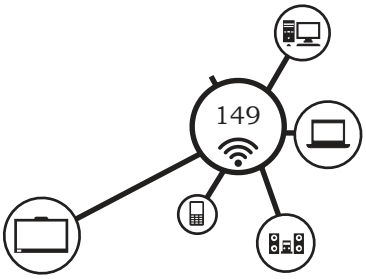
Relational Database Model

The Relational Database Model was proposed in 1970 by E. F. Codd. Relational database model is the most common type of database model. The data elements are stored in different tables made up of rows and columns. The data in different tables are related through the use of common fields. So relations are set between tables based on common fields. That is why this model is termed as relational database model.

Relational Database Terminology

Let us get familiar with some of the common terms used in RDBMS.

- **Entity –** It is a real world object about which information is to be stored in a database. For example, if we want to store information about an entity Student in a school, then we need to have his admission number, roll number, name, father's name, date of birth, etc. These details associated with the entity are called **attributes**. Each entity is a collection of these attributes associated with it. So roll number, name, admission number, etc., are attributes associated with the entity student. These attributes are represented in the form of columns.
- **Table –** A table is a collection of logically related records. It is organised as a set of columns, and can have any number of rows. For example, the Employee table can have columns, namely name, designation, department and have records or rows having data of 100 employees.
- **Field or Columns or attributes –** A field is the smallest entity in the database. A collection of fields make a record, a collection of records make a table and a collection of tables make a database. Fields are individual record characteristics and are presented as columns within a table. Data values are stored in a database as fields. A field holds the data values of one type of data for several persons. For example, in the Employee table the field "Emp Name" will hold the names of employees of an organisation.
- **Data Values –** Data values are the raw data represented in numeric, character or



alphanumeric form. Examples of data values are 'Abhinav Bindra', '26' 'shooting', "Chandigarh", "10-12-2018", etc.

- **Record or Row –** The data values for all the fields related to a person or object is called a record. It is presented as rows within a table. A record holds the data values of all the fields for a single person or object in a table. For example, in the Employee table with the field names as Name, Designation, Department, the data values of all the fields for an employee may be ('Abhinav', 'Manager', 'Finance') and this forms one record.
- **Primary Key –** A primary key or simply a key is a field that uniquely identifies a row in a table. The key identifier can be the value of a single column or of multiple columns. The primary key is a unique identifier for the table. The column or combinations of columns that form the primary key have unique values. At any time, no two rows in the table can neither have same values for the primary key nor can data value for such field be left blank. For example, in a student table, each student has a unique roll no., which forms the primary key. If, in a table we use more than one fields to identify a record, it is known as a composite key. For example, we may form a composite key consisting of fields roll no. and name.
- **Relational Database –** A relational database is a collection of related tables. For example, in Fig. 8.3, the database contains two related tables.

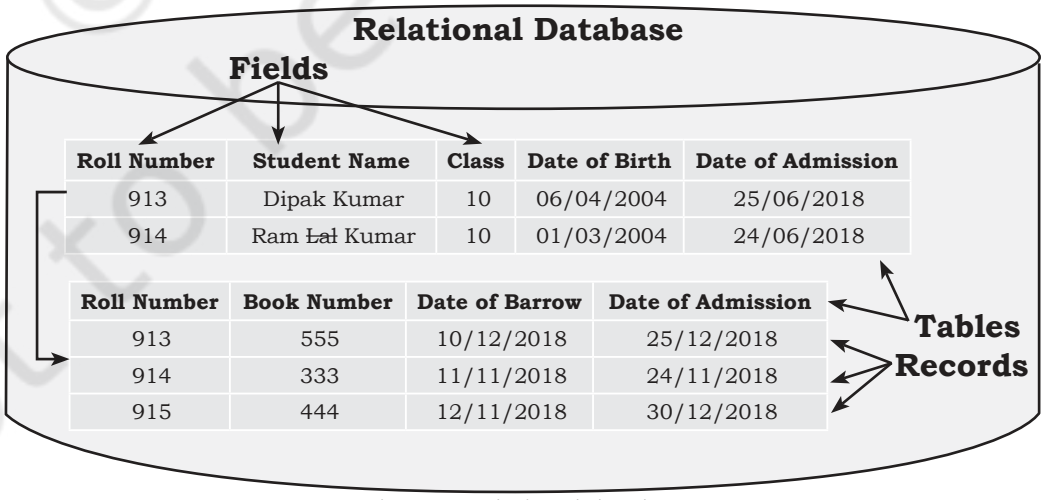


Fig. 8.3: Relational database

- **Foreign Key –** If a field or a combination of fields of one table can be used to uniquely identify records of another table, then that particular field is known as the foreign key. This foreign key helps to build a relation between two tables. Consider the example given below.

Student Registration Table

Enrolment Number	Student Name	Class	Date of Birth	Date of Admission
XX1234567890	Dipak Kumar	10	06/04/2004	25/06/2018
XX1234567891	Ram Kumar	10	01/03/2004	24/06/2018

Primary key: Enrollment Number

Student Marks Table

Roll Number	Maths	Science	Vocational	Enrolment Number
44983	87	75	80	XX1234567890
44990	74	45	75	XX1234567891

Primary key – Roll Number, **Foreign key –** Enrollment Number

In Student Registration Table, 'Enrolment_Number' is the primary key and in the Student Marks Table, 'Roll_Number' is the primary key, whereas 'Enrollment_Number' is the foreign key. This foreign key can be used to set a relation between two tables.

- **Candidate Key –** All the field values that are eligible to be the primary key are the candidate keys for that table. Such fields can neither be left blank nor can have duplicate values. So in the table Student Marks, Enrollment Number and Roll Number both are candidate keys.
- **Alternate Key –** Out of the candidate keys, one or two are made as primary keys. The others are the alternate keys. Hence, if Roll Number is made as the primary key, Admission Number is the Alternate key.

Objects of an RDBMS

An object in a database is a structure or a feature that is used to store, represent or retrieve data. In fact a database is a collection of these objects that work on multiple sets of data related to each other. Various objects in a database are as discussed below:

- **Table** – As mentioned before, a table is the basic unit of any DBMS. The data is first stored in tables

in row and column format. A column represents a field or an attribute while a row represents a record.

- **Forms –** A form is a feature of a database using which we can enter data in a table in an easy and user friendly manner. A form consists of text boxes, labels, radio buttons, list boxes, check boxes etc. that give a user friendly interface for entering data. The data entered through the forms is stored in tables.
- **Queries –** A query is used to retrieve the desired information from the database. In simple terms, it is a question asked from the database. For example, if we want to view the names of only those students who have scored more than 50 marks, then we post a query. The data set matching the given criterion is retrieved from the table and displayed on the screen.
- **Reports –** The output of a query may be displayed in the form of reports. The usual result of the query is in the form of rows and columns. But if we want the report to be formal and in proper layout, then we can use the Reports feature of RDBMS.

Let's Practice

Consider the following table and answer the questions that follow.

Item No.	Name	Price	Quantity	Discount (in%)
A001	Pen	20	12	0
A003	Pencil	15	5	1
A010	Notebook	50	25	5

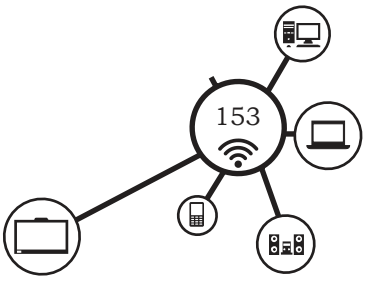
From the above table, identify the primary key. Justify your choice.

How many fields and how many records does the table have?

SUMMARY

- The raw facts constitutes data.
- Information is the processed or organised form of data.
- A database is a collection of logically related data items stored in an organised manner.

- The software that is used to create, update and retrieve data is known as database management system (DBMS).
- Some of the common examples of DBMS are MS Access, Open Office or LibreOffice Base, Oracle, Ingress, MySQL.
- Data Model is the structure of database and it describes the manner in which data will be stored and retrieved.
- There are different data models, such as hierarchical data model, network data model and relational data model.
- In Hierarchical Data Model, the data is organised into a tree like structure. The data is stored in the form of linked records.
- In Network Data model, multiple records are linked to same master file.
- The Relational data model is based on the principle of setting relationships between two or more tables of the same database.
- Entity is a real world object about which information is to be stored in a database.
- The details associated with the entity are called attributes.
- A table is a collection of logically related records. It is organised as a set of columns, and can have any number of rows.
- A field is the smallest entity in the database. These are individual record characteristics and are presented as columns within a table.
- Data values are the raw data represented in numeric, character or alphanumeric form.
- The data values for all the fields related to a person or object is called a record. It is presented as rows within a table.
- A primary key is a field that uniquely identifies a row in a table.
- This foreign key helps to build a relation between two or more tables in a database.
- All the field values that are eligible to be the primary key are the candidate keys for that table.



- Out of the candidate keys, one or two are made as primary keys. The others are the alternate keys.
- An object in a database is a structure or a feature that is used to store, represent or retrieve data. The various objects in a database are tables, forms, reports and queries.
- A form is a feature of a database using which we can enter data in a table in an easy and user friendly manner.
- A query is used to retrieve the desired information from the database.
- The output of a query may be displayed in the form of reports.

Check Your Progress

A. Multiple choice questions

1. Which of the following can be considered as an example of a database?
   (a) Dictionary
   (b) Telephone directory
   (c) Marks Register
   (d) Newspaper
2. Which of the following is NOT a DBMS?
   (a) MS Access
   (b) Open Office Base
   (c) MS Excel
   (d) MySQL
3. DBMS stands for _____.
   (a) Data and Books Management System
   (b) Database Management System
   (c) Duplicate Books Management System
   (d) Data Management Multi System
4. Which of the following data models sets a relation between the two or more tables?
   (a) Relational Data Model
   (b) Network Data Model
   (c) Hierarchical Data Model
   (d) Connection Data Model
5. The details associated with an entity are called _____.
   (a) Table
   (b) Attributes
   (c) Records
   (d) Primary key

6. A _____ is represented as rows in a table.
   (a) field
   (b) attribute
   (c) record
   (d) candidate key
7. In which of the following forms can a data value be represented?
   (a) Numeric
   (b) Character
   (c) Alphanumeric
   (d) All of the above
8. Which of the following uniquely identifies a row in a table?
   (a) Primary key
   (b) Alternate key
   (c) Foreign key
   (d) Candidate key
9. A _____is a feature of a database using which we can enter data in a table in an easy and user friendly manner.
   (a) query
   (b) report
   (c) form
   (d) field
10. A _____ is a question asked from a database.
    (a) query
    (b) report
    (c) form
    (d) field

B. State whether the following statements are True or False

1. A database cannot be organised.
2. Data is the collection of raw facts.
3. A table can be created without a primary key.
4. Two tables can be related in a network data model.
5. MS Access is an example of a database.

C. Fill in the blanks

1. The raw facts constitutes _____.
2. An _____ is a real world object about which information is to be stored in a database
3. The output of a query may be displayed in the form of _____.
4. The data values for all the fields related to a person or object is called a _____.
5. All the field values that are eligible to be the primary key are the _____ keys for that table

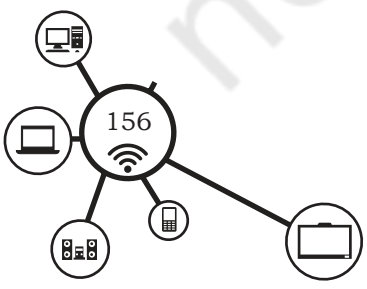
D. Answer the following questions

1. Define the terms
   (a) Database
   (b) Data redundancy
   (c) Report
2. Give one point of difference between
   (a) Data and Information,
   (b) Form and Query,
   (c) Network and hierarchical data model
3. Give any four advantages of a DBMS.
4. Consider the table given below and answer the questions that follow

Table: Library

Book_Id	Book Name	Author Name	Price	Publisher
F001	Pride and Prejudice	Jane Austen	550	ABC
S004	Amazing Astronomy	E. Shane	1050	ABC
C005	**IT and Mankind**	**MHA Diwaan**	**2500**	**HYM**

   (a) Name the fields in the given table.
   (b) Which field should be made the primary key?
   (c) Is there any alternate key in the table?
   (d) How is primary key different from foreign key? Explain with example.



CHAPTER 9

Starting with LibreOffice Base

Introduction

In the previous chapter we have learned about databases and database management system (DBMS). LibreOffice Base is a free and open source DBMS. It can be downloaded from www.libreoffice.org and is available for both Linux and Windows operating systems. Data has to be stored in an organised manner using a DBMS. Also, the data being stored can be a text, number, date or in any other form. So, we need to understand different types of data that can be stored in a table. The data types of the fields have to be specified while creating tables in a database. Thereafter valid data is entered and stored in a table.

Data Types

The nature of data to be entered for various fields are of different types. For example, names are stored in the form of text, age in numbers, fees in decimal numbers, date of birth in date format and so on. A data type refers to the type of data that will be stored in that particular field. The memory size of a field varies according to its data type.

Some commonly used data types are described below.

Text Data Type – The text data is a combination of letters, numbers or special characters. No arithmetic calculations can be performed on text data. Examples of text data type is PAN Card Number, Name, Marks, etc.

The table below lists various data types that can store textual data.

Table 9.1 Text Data Types

Name	Data type	Description
Memo	LONGVARCHAR	Stores up to the maximum length indicated by user. It is used to store some descriptive data having more than 255 characters. Memo data type allows to store text data up to 64,000 characters.
Text (fix)	CHAR	Stores exactly the length specified by user. Character data type is used to enter fixed number of characters. It can be used for license number, passport number as they have fixed number of characters.
Text	VARCHAR	Stores upto the specified length. The number of bytes allocated depends on the number of characters entered by the user. For example, the address is defined as varchar (50), and if the address entered by the user is of 20 characters then only 20 bytes will be occupied in the database.

- **Numeric Data Type –** Numeric data types consists of numbers. The numbers can be integer or real numbers on which any type of arithmetic calculations can be performed. For example, 10, -34.8, 90.6789 , -86 are of numeric data type. Table 9.2 lists different numeric data types along with the number of bits/bytes it uses and its range.

Table 9.2: Numeric data types

Name	Data type	Signed	Range
Tiny Integer	TYNYINT	No	0-255
Small Integer	SMALLINT	Yes	-32768 to 32768
Integer	INTEGER	Yes	-2.14×109 to 2.14×109
BigInt	BIGINT	Yes	-2.3×1018 to 2.3×1018
Number	NUMERIC	Yes	Unlimited
Decimal	DECIMAL	Yes	Unlimited
Float	FLOAT	Yes	
Real	REAL	Yes	5×10(-324) to 1.79×10(308)
Double	DOUBLE	Yes	

- **Currency Data Type –** The currency data type indicates the monetary values and can be stored using currencies of various countries. For example $100, £ 500 or Rs. 25.50.
- **Date Data Type –** This data type is used to indicate dates and time. For example 12/25/2019, 08:45 AM. The data and time can be stored in various formats. Table 9.3 list various forms of date data type.

Table 9.3: Various forms of Date data type

Name	Description
Date	Stores the year, month and day as it is stored in the system.
Time	Stores the time of the day as hour, minute and second.
Timestamp	Stores date and time information at once.

- **Boolean –** In boolean data type there can be only two values- True or False. This also can be given in multiple formats like Yes/No, True/False, On/Off.
- **Binary –** The Binary data type used to store digitized images and sounds that comes as long string of zeros and ones. It is possible to store photos of the products or employees, or sound snippets or voice messages in Base database.

Starting with LibreOffice Base

Annual Sports Day is being held in Ruhi's school in which various sports competitions will be conducted. Ruhi has been asked to create a database consisting of various sports activities and its players. She decides to create a database named, 'Sports Day' consisting of a table 'Events' that contains the following fields:

Event_Id	-	to store Event identification no like E001, E002, etc.
Event Name	-	to store name of the event
Date	-	to store date on which the event will be conducted
Winner 1 Name	-	name of the winner
Winner 1 Points	-	points earned y the winner

Let's Practice

What should be the data type of the fields of the table Events that has to be created by Ruhi?
Which field should be made the primary key?

Once the table with its fields has been decided, Ruhi decides to start creating the database using LibreOffice Base. Follow the following steps to create the database.

Step 1. Start the LibreOffice Base as per the standard process of starting the application in Windows or Linux. In Windows, click **Start > LibreOffice** or double click on the **LibreOffice** icon on the desktop or Select **Base Database** option from the bottom left panel.

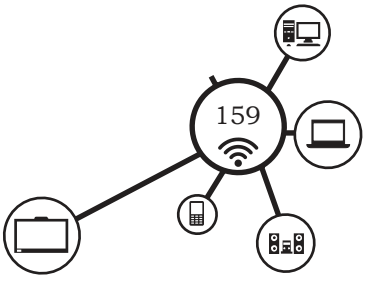


Fig. 9.1(a): Clicking on the Base icon on the left panel to open Base

Fig. 9.1(b): Clicking on Show Applications to search for installed packages

Fig. 9.1(c): Clicking on the searched Base icon to open Base

In Ubuntu Linux, click on the Base icon located on the left panel as shown in Fig. 9.1(a) or click on the ***Show Applications*** as shown in Fig. 9.1(b) to search for its icon. Type Base in the search box. The LibreOffice Base application icon will be displayed as shown in Fig. 9.1(c). Click on the Base icon to open the Base application.

Step 2. A **Database Wizard** opens as shown in Fig. 9.2. It allows you to create a new database or open an existing database.

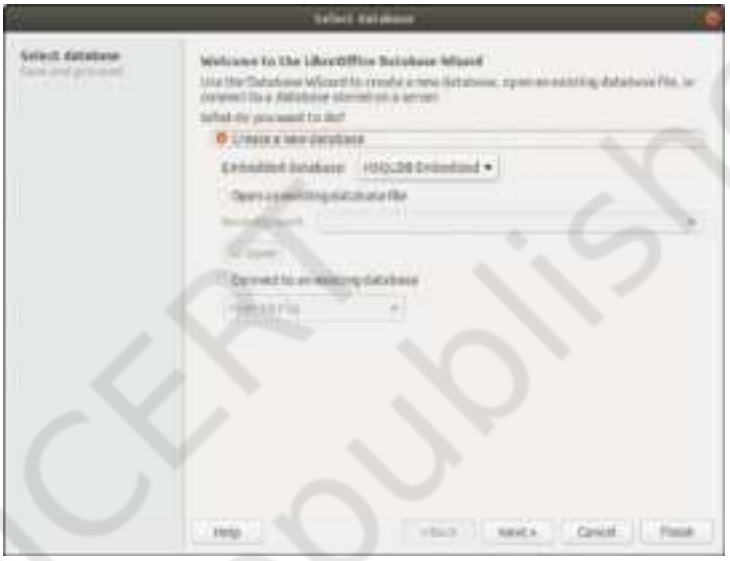


Fig. 9.2: LibreOffice Database Wizard

Tip: To open LibreOffice Base, you may search the application Base on your computer and click on the LibreOffice Base icon.

Step 3. Since we want to create a new database, so click **Create a new database** radio button. If any database that has already been created is to be opened, then select **Open an Existing Database** option to select the database to be opened. Thereafter click **Next** button.

Step 4. The Next step gives the option to register our database with LibreOffice.org. If we register the database, then our database is made public and hence can be accessed by other people. As of now, we would not like to register, so we click and select the radio button with option, **'No, do not register the database'**.

Fig. 9.3: Decide How to Proceed After Saving the Database

Step 5. This step also asks whether you want to **open the database for editing** or want to create a table using the wizard. The option Open the database for editing is already selected.

Step 6. Click **Finish** button to complete the database creation process. The **Save As** dialog box appears.

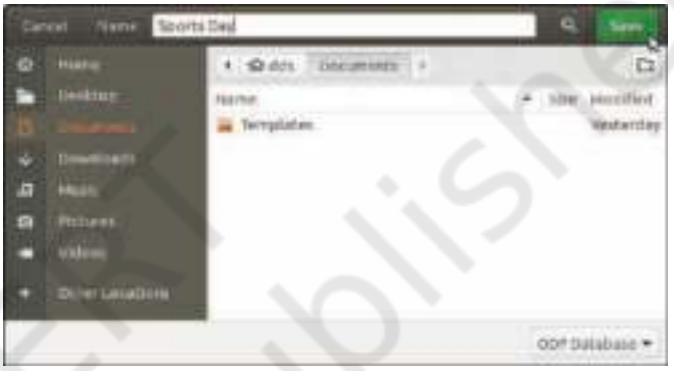


Fig. 9.4: Save As dialog box

Step 7. Browse for the drive and folder where you want to store your database.

Step 8. Type the name (Sports Day) in the **File name** text box.

Step 9. Click **Save** button. The database in Base is saved with an extension **.odb**. Also note that if no name is given to the database, then it is saved with a default name as NewDatabse.odb.

User Interface of Libre Office Base

Once a database is created, the screen as shown in Fig. 9.5 appears. This is the User Interface of Open Office Base.

The various components of the Base User Interface are discussed below:

Fig. 9.5: User Interface of LibreOffice Base

- **Title Bar –** The title bar displays the name of a database and an application in which it is made. The windows buttons to maximize, minimize or close the window are located on the right corner of the title bar.
- **Menu Bar –** The menu bar appears below the title bar. It consists of seven menu items – **File, Edit, View, Insert, Tools, Window and Help**. All these menu items contain commands that help to perform various operations on the database.
- **Standard Toolbar –** It is located below the menu bar. It is used to access frequently used tools.
- **Status Bar –** It is located at the bottom of the interface window. It displays information about the type of view of the object in the database. Database Pane – The database pane is located on the left side of the window. LibreOffice Base is the collection of related data objects known as Tables, Forms, Queries, Reports and application modules. Depending on the object that is selected, the respective Task Pane and Object Area displaying the created object appears.

Fig. 9.6: Opening database from menu option

Opening a Database

To open an already created database, click **File > Open**. The Open dialogue box appears as shown in Fig. 9.6. Browse for the folder where the database to be opened is stored. Select the desired database and click on Open button.

Tip: Use the keyboard shortcut key ***Ctrl+O*** to open an already existing database.

Creating a Table

Once the database is created, we can start working with objects of the database. First and foremost is the creation of the table and then entering data in the table. A table in LibreOffice Base can be created using a wizard or using the Design view.

Creating a Table using a Wizard

A table can be created at the time of creating a database or after creating database. To create a table

at the time of creating a database in database wizard, select the radio button with option **"Create tables using the table wizard"**, then click on the **Finish** button.

Since our database is already open, so we will select and click If the database is opened we can select **Use Wizard to create a table** option from the Tasks Pane. The **Table Wizard** dialog box will open as shown in Fig. 9.7. Follow the instructions in the wizard to create the table with desired fields.

The Table Wizard of Base consists of ready-made tables. Click the Sample tables list box and select select any one table, say Customer from the drop down list.

After selecting the Customer table, various fields appears under ***Available Fields***. The required fields from ***Available fields*** box can be shifted one by one to ***Selected fields*** box using > button as shown in Fig. 9.8. You can select all the fields from ***Available fields*** to ***Selected fields*** in one stroke by clicking on >> button.

Click on **Next** button. It will move you to step 2. Set types and formats as shown in Fig. 9.9.

Then click on **Next** button. It will move you to step 3. Set primary key. Let us set the CustomerID field to primary key as shown in Fig. 9.10.

Click on **Next** button, that will take you to the complete the process of creating table using wizard and display the screen as shown in Fig. 9.11. This screen will give you three choices. By default the "Insert data immediately" option selected.

Fig. 9.7: Table Wizard

Fig. 9.8: Fields shifted from Available fields to Selected fields

Fig. 9.9: Setting types and formats

Fig. 9.10: Setting primary key

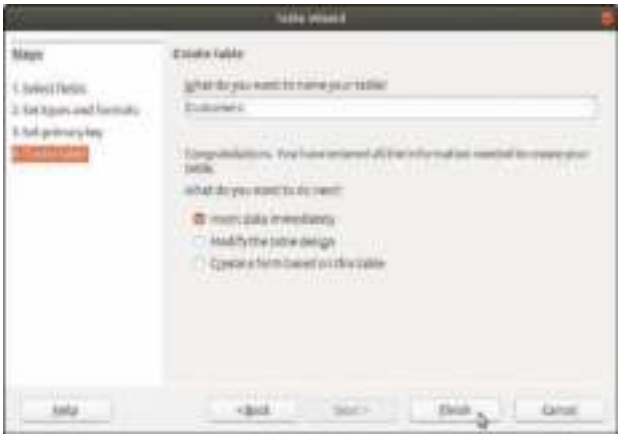


Fig. 9.11: Creating table using wizard

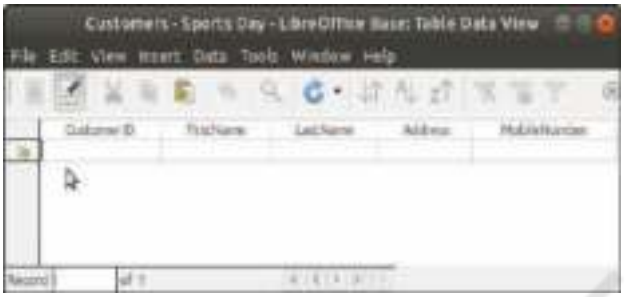


Fig. 9.12: Datasheet view of Customer table

Fig .9.13: Creating Table in Design view

Click on the **Finish** button to complete the process of creating the table and using wizard. The next screen will allow to enter the data as per the selected fields. The data sheet view interface as as shown in Fig. 9.12 will open that allow to enter the data in the Customer table.

Creating Table in Design View

Creating the table using wizard restricts us to use the same fields in the per-designed tables. It may not solve the purpose in real scenario, as we may require to create a table with the different fields for our purpose. Creating tables using Design View gives us more flexibility to do our work.

To create table using Design View, in main Database window, click on the option **Create Table in Design View** in the Tasks Pane and Table Design Window will be opened as shown in Fig. 9.13.

The screen is broadly divided into two sections or horizontal panes.

The upper half consists of a grid structure with three columns Field Name, Field Type and Description.

- **Field Name –** It is the name of the field assigned at the time of creation of table.
- **Field Type –** It allows to assign a data type to the field.
- **Description –** It allows to describe the purpose of the field. It is not the part of database table, but it is meant for the user to understand the purpose of the field. We may or may not enter field description.

The Field Properties pane is located at the bottom half of the window. It displays the field properties assigned by the database designer. These properties can also be

changed as per the requirement and are used to control and validate the data that is to be entered.

Let us help Ruhi to create a table named Events using in the option Create Table in Design view in the Sports Day database. Follow the follow the following steps to do so.

Step 1. Type the first field name (EventId) in the **Field Name** column. Press **Tab** key. The cursor moves to the second column i.e. **Field Type**.

Step 2. The **Field Type** column contains a list box. As you click on the down arrow, it appears and we can select the desired data type from the list box. Select the datatype (Varchar).

Step 3. Observe that certain properties appear in the Field Properties Pane as the data type is selected. Some of the properties are Entry required, Length, Default value, Format example. Set the desired properties for the entered field.

Step 4. Press **Tab** key to move to the next column. Add any description if you want in the third column.

Step 5. Once the properties for the field are set, press **Tab** key to move to next row.

Step 6. Enter the next field by repeating steps 1,2 and 3. Repeat the process for adding all fields in the table. Fig. 9.15 shows the Design View with all the fields.

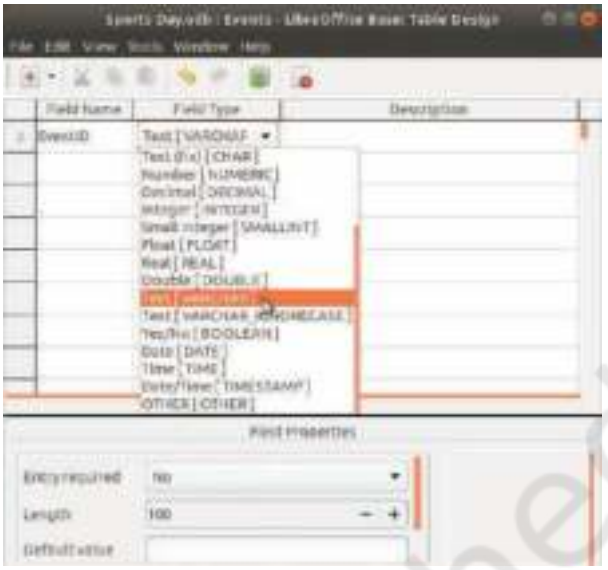


Fig. 9.14: Selecting data type for field

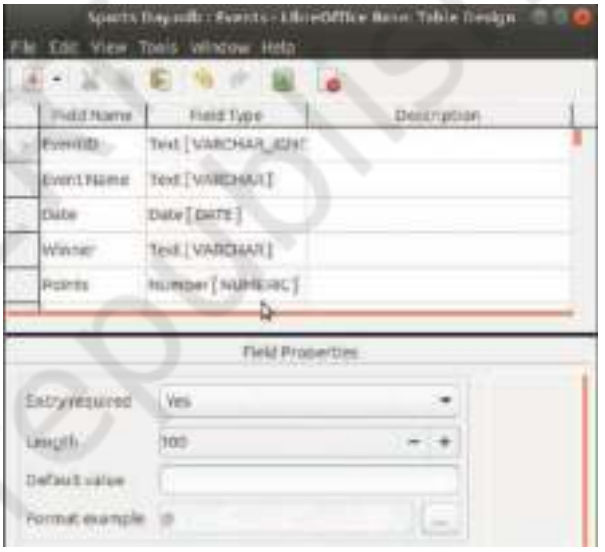


Fig. 9.15: Fields entered using Creating Table in Design View

Setting the Primary Key

As you have learned that every table must have a primary key that uniquely identifies a record in the table. To make a particular field as the primary key, place the mouse pointer before the field name, say Event Id in our above example and right click. A pop up menu appears.

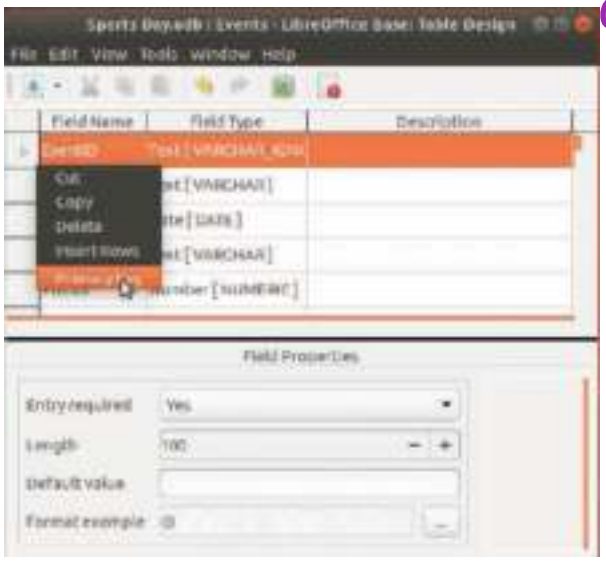


Fig. 9.16: Assigning primary key

Select the Primary Key option from pop up menu as shown in Fig. 9.16. A key icon appears before the field name indicating that it is a primary key.

Saving a Table

After creating the table you need to save it on the disk. To save the table click on the save button or follow menu option **File > Save As**. A Save As dialog box is displayed as shown in Fig. 9.17. Enter the name of table and click on **OK** button.

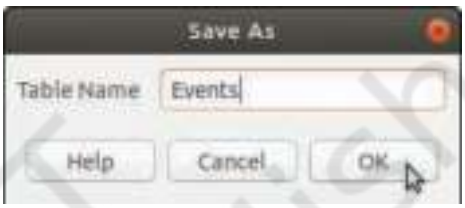


Fig. 9.17: Saving a table

Tip: Press **Ctrl + S** to save the table or click Save icon from the toolbar. If the table is being saved after making some changes, simply select File-> Save option.

Once the table design is complete, click on **Close** button on the toolbar to return to the Database screen. The name of the table will appear in the Tables Object Area as shown in the Fig. 9.18.

Fig. 9.18: Event table added to the database

More to know

To set a **composite key**, i.e. a primary key consisting of two fields, keep the ***Ctrl*** key pressed and then click on multiple fields to select them. Thereafter right click on selected fields and choose **Primary Key** option from the pop up menu.

Entering Data in a Table

To enter data in the table, double click on the created table Events icon in the Tables Object Area on the database screen. Alternatively, we can open the table by right clicking on the desired table and then selecting

the **Open** option from the drop down menu. The datasheet view of the table will appear as shown in Fig. 9.19. It displays the field names in the top row. These fields are displayed in the same order as they were added while creating the table.

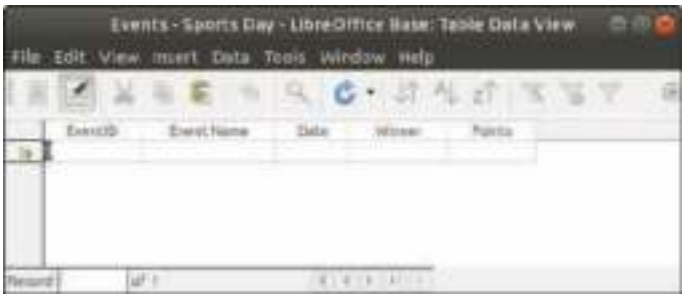


Fig. 9.19: Datasheet view of Events table

If the number of fields are more and cannot fit in the single row, you can use the horizontal scroll bar to view all the fields. The cursor will be blinking in the second row. Start typing the data value for each field. Use **Tab** to move to next field. Once all the data values are entered for a single record, the cursor moves to the next record. This process is called as data entry. Enter 4-5 records in the Events table as shown in Fig. 9.20.

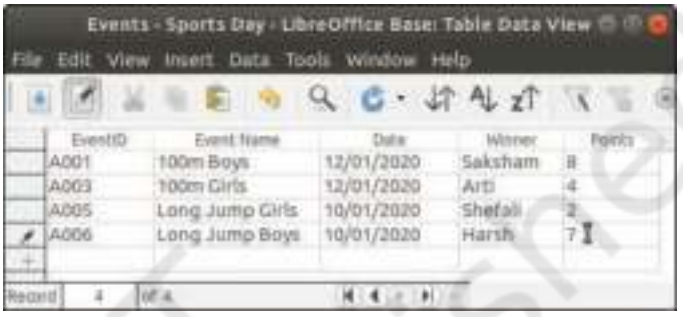


Fig. 9.20: Data entry in Event table

Navigating through the Table

The black pointing arrow (▶)just before the field name is the record pointer. To navigate through various records of the table, we use the navigation box present at the bottom of the datasheet window as shown in Fig. 9.21. It indicates a current record of the table at any given time.

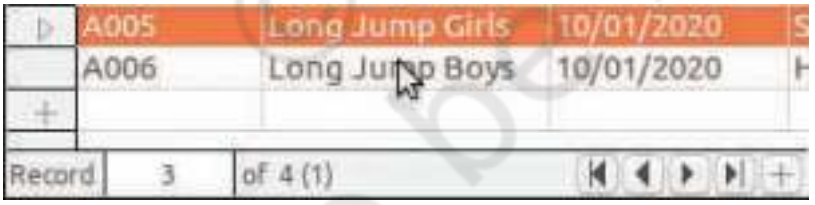


Fig. 9.21: Navigation box

The various components of Navigation Box are as follows:

- **Record Selector Box –** T his is the text box where the currently active record number is displayed. We may enter the record number that we want to see in this text box.
- **Navigation Buttons –** These are used to scroll vertically in the table.

Press ⏮ to move to the first record

Press ▶ and ◀ to move to the next and previous records respectively.

Press ⏭ to move to the last record

Fig. 9.20 shows the datasheet view after entering four records in the table. Note that the record pointer is on fourth record, the Navigation Bar shows the "Record 4 of 4".

Editing Data

To edit or modify the previously entered data simply place the cursor on the field value that has to be edited to edit and enter the new value.

The Edit icon (✎) appears before the record that is being edited. This icon is displayed till the table is saved after making the required changes. Press **Esc** key to cancel the corrections made and restore the original contents.

Deleting Records from Table

To delete any record from the table, open the table and select the record to be deleted. The record can be deleted by pressing the **Del** key from the keyboard or selecting the **Delete Record** option from the **Edit** menu. Alternatively a record can also be deleted by right clicking on the record and clicking on the **Delete Rows** option from the pop up menu.

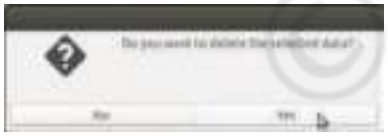


Fig. 9.22: Delete record alert

Attempting to delete the record will display the Confirmation box as shown in Fig. 9.22. Clicking on **Yes** button will finally delete the record, while clicking on the **No** button will not delete the record.

Sorting Data in the Table

Data in a table can be arranged in ascending or descending order. This process of arranging the records in particular order on any filed is called as **sorting**. Follow the following steps to sort the table.

Fig. 9.23: Sorting the Event table on Points field

Step 1. Open the *Event* table in datasheet view and select the field on which you want to sort. Fig. 9.23 shows that the field "*Points*" is selected to sort the records in ascending order.

Step 2. From the tool bar click ***Sort Ascending*** icon if the table has to be sorted in ascending order of selected field. Alternatively select ***Sort Descending*** icon if the table has to be sorted in descending order of selected field.

Step 3. The table will be sorted in the ascending order of points as shown in Fig. 9.24.

Fig. 9.24: Event table after sorting on Points field

Sometimes we may need to sort the table based on more than one fields. In such case, click ***Sort*** icon on the toolbar. The Sort Order dialog will be displayed as shown in the Fig. 9.25.

Fig. 9.25: Sort Order dialog box

Select the appropriate field name and field value according to which the records are to be sorted. Choose the order of sorting ascending or descending as required from the Order drop down box. Multiple fields may be selected in this dialog box. Click **OK** button once done.

Closing LibreOffice Base

To close the application window of LibreOffice Base, click on the **File > Close** or click on the cross (x) button of the LibreOffice Base window.

Let's Practice

There are 5 houses in Ruhi's school – Ganga, Yamuna, Satluj, Beas and Narmada. Once student can participate from each house in a particular event. Create a table Participants with following fields.

1. Event_Id
2. Event Name
3. Ganga
4. Yamuna
5. Satluj
6. Beas
7. Narmada

Also add minimum five records in the table.

Summary

- Base is a free and open source database component of LibreOffice suit.
- It can be downloaded from www.libreoffice.org
- A data type refers to the type of data that will be stored in that particular field.

- Various in Base can be categorized into Text , Numeric, Currency and Date.
- LibreOffice Base is the collection of related data objects known as Tables, Forms, Queries, Reports and application modules.
- Depending on the object that is selected, the respective Task Pane and Object Area displaying the created object appears on the User Interface Window.
- A table in LibreOffice Base can be created using a wizard or using the Design view.

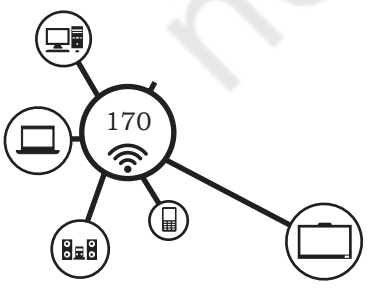
Practical Exercise

Ananthu wants to create a directory containing data – Serial No, Name, Mobile Number, Email id, Date of birth of his friends. Create a table using LibreOffice Base. Make Serial No as primary key. Enter minimum 5 records in the table. Sort the table in alphabetical order of name.

Check Your Progress

A. Multiple choice questions

1. Which of the following is NOT a type of text data type?
   (a) Memo
   (b) Varchar
   (c) Float
   (d) Char
2. A currency data type can only store monetary data that is in dollars.
   (a) True
   (b) False
   (c) Neither a nor b
   (d) Both a and b
3. Which of the following data can a date data type store?
   (a) Date
   (b) Time
   (c) Both date and time
   (d) Neither date nor time
4. Which of the following is true about LibreOffice Base?
   (a) It is a spreadsheet software
   (b) It is free and open source software
   (c) It can store only character data
   (d) It is a licensed software.



5. Which of the following methods can be used to create a table in Base?
   (a) Using a table wizard
   (b) Design View
   (c) Both a and b
   (d) Neither a nor b
6. The related objects of a database can be seen in _____ pane of the Base Database window.
   (a) Database
   (b) Task
   (c) Title Bar
   (d) Menu Bar
7. Which is the shortcut key to open an existing database?
   (a) Ctrl+ D
   (b) Ctrl+O
   (c) Ctrl+E
   (d) Ctrl+F
8. The Design view of Table Creation window in LibreOffice Base is divided into _____ sections or panes.
   (a) 2
   (b) 3
   (c) 4
   (d) 5
9. While entering records in a table, we can move to the next field by pressing the _____ key.
   (a) Tab
   (b) Ctrl
   (c) Enter
   (d) Shift
10. Which of the following is true about primary key of a table?
    (a) Every table must have a primary key
    (b) The data values in primary key field cannot be duplicated.
    (c) A primary key field cannot be left blank
    (d) All of the above

B. State whether the following statements are True or False

1. The text data can contain special characters.
2. Memo data type can be used to store descriptive data.
3. A Boolean data type can have two or more than two values.
4. We cannot store audio data in LibreOffice Base.
5. The properties of a field change according to the data type selected.
6. Field description may or may not be entered while designing a table.
7.  is pressed to move to the last record.
8.  appears when the record is being edited.

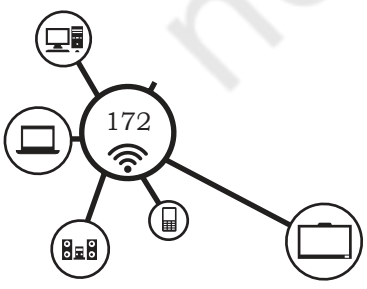
9. A table once created in a database cannot be edited.
10. Sort dialog box can only help to sort data in ascending order.

C. Fill in the blanks

1. A data type refers to the type of data that will be stored in that particular field.
2. The _____ data is a combination of letters, numbers or special characters.
3. _____ data type can be used to store Aadhar number.
4. The _____ data type used to store digitized images.
5. The shortcut key to save a table is _____.
6. _____ on the Base Interface Window displays information about the type of view of the object in the database.
7. A _____ icon appears before the field name indicating that it is a primary key.
8. The data can be entered in a table only in _____ view.
9. The black pointing arrow just before the field name in a table is called _____.
10. The process of arranging the records in particular order on any filed is called _____.

D. Answer the given questions

1. Differentiate between:
   (a) Memo and Varchar data type
   (b) Number and Decimal data type
   (c) Design View and Datasheet view of a table
2. Name the menu items present on the Base User Interface,
3. Label the components – Title Bar, Database Pane, Tasks Pane, Status Bar of the LibreOffice Base User Interface.
4. How can we define a primary key in a table?
5. Write steps to sort the table in descending order of primary key.
6. What is the use of navigation box with respect to tables in a database?



Chapter 10 Working with Multiple Tables

Introduction

We have learned to create tables in a database. Once the tables are added in a database, you may require to edit or delete the table. Also relations are set up between the tables to control data redundancy and inconsistency. This helps in proper maintenance of a database by checking that neither the records are duplicated nor there is variable data value for a particular field in two or more tables. If you set up relations between tables, then adding or updating a record in one table reflect the changes in all the related tables.

Editing and Deleting Tables

In the previous chapter, we have created a Customer table using a wizard in Sports Day database. It is possible to copy, rename, edit and delete the table of database by right clicking on the table name and using the appropriate option from the pop up menu.

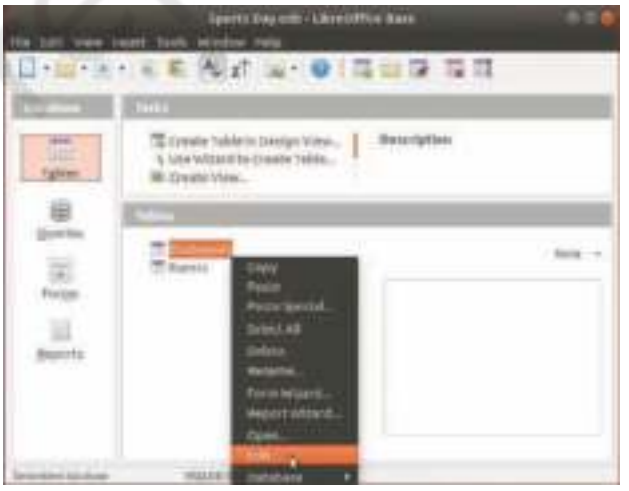


Fig. 10.1: Selecting Edit option from pop up menu

Editing a table involves the task such as adding a new field or removing any field in a table or to alter any of the field properties. To edit a table, open the **Database User Interface** window. Selecting the **Table** object in **Database Pane**, the list of tables will be displayed in the **Table Area**. Right click on the table name and select **Edit** option from the pop menu. In our example we have selected the Customer table as shown in Fig. 10.1.

The design view window of the table will be displayed as shown in Fig. 10.2. Do the required modifications

Fig. 10.2: Table in design view window

Fig. 10.3: Selecting Delete option from pop up menu

Fig. 10.4: Confirm Deletion Alert

and save the table. Also note that the changes so made will not affect the previous records entered in the table.

Similarly to delete a table, right click on the table to be deleted, say Customer table and select the **Delete** option from the pop up menu as shown in Fig. 10.3.

A confirmation box to confirm for deletion of the table will be displayed as shown in Fig. 10.4. Click on **Yes** button to finally delete the table.

To rename a table, right click on the table name in the Table Area and select **Rename..** option from the pop up menu. A cursor will appear. Type the new name and press the **Enter** key.

Relationships between Tables

While working with multiple tables, you need to check the redundancy and inconsistency of data. The record for a particular entity should neither be repeated nor different data values should appear for a single entity in the database.

This is done by setting relationship between the tables of a database. The most important prerequisite for setting a relationship is that there must be a common field(s) between the two tables to create a relationship.

Let us consider an example of a database containing following two tables–Student_Details and Student_Result

Table 10.1: Student_Details

Admission No	Name	Father's Name	Class	DOB
1001	Mampi	D K Bose	10	12/3/2004
1005	Harnoor	Tej Singh	9	2/5/2005
1110	Sanjeeva	B Reddy	10	13/11/2004
1002	Neeru	Rajesh Jain	7	14/11/2006
1134	Urjit	Mahesh Patel	10	7/7/2003

Table 10.2: Student_Result

Roll No	Admission No	Class	Aggregate_Marks
1	1001	10	78
2	1110	10	93
3	1134	10	46

In Table 10.1 (*Student_Details*), Admission No is the primary key. In table 2 (*Student_ Result*), *Roll No* is the primary key and *Admission No* is the foreign key. So each record in Table 10.2 has a value of Admission No that corresponds to a record in Table 10.1 with same value of Admission No. It is important to note that the data types of the common field in both the tables must be same. If they are not same then LibreOffice Base will display an error message and will not allow to set the relationship between the two tables.

Once the relationship between the two tables has been set, the integrity of data will be managed by the DBMS. That means once a student's record has been entered in the *Student_Details* table, only then that particular *Admission No* can be entered in the *Student_ Result* table. The record pertaining to *Admission No* in *Student_Details* table is considered as the **master record** while the corresponding record in the related table (*Student_Result*) is the **transaction record**. Therefore *Student_Details* is called the **master table** and *Student_ Result* is called the **transaction table**.

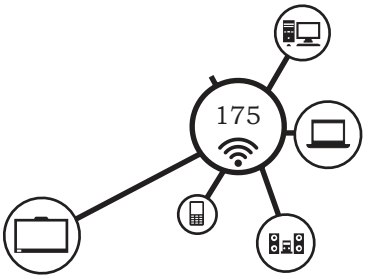
Types of Relationships

The type of relationship between any two tables in a database is based on the number of records that are present in the transaction table corresponding to the master table. Primarily three types of relationships can be set up between two tables in a relational database These are:

(i) One-to-One
(ii) One-to-many
(iii) Many-to-Many

One-to-One relationship

In this type of relationship, one specific record of a master table has one and only one corresponding record in the transaction table. For example, the record for *Admission_No* in the master table (*Student_Detail*) will have only one corresponding record of same value of *Admission No* in the transaction table of *Student_ Result*. This is because no two students will be given same admission number. Similarly one person can have



only one ticket to get entry into a stadium to view the match. So relationship between Student and Admission number and a person and his ticket number will be one-to-one relationship (Fig. 10.5).

Table 1: Student_Details

Admission No	Name	Father's Name	Class	DOB
1001	Mampi	D K Bose	10	12/3/2004
1005	Harnoor	Tej Singh	9	2/5/2005
1110	Sanjeeva	B Reddy	10	13/11/2004
1002	Neeru	Rajesh Jain	7	14/11/2006
1134	Urjit	Mahesh Patel	10	7/7/2003

Table 2: Student_Result

Only One corresponding record

Roll No	Admission No	Class	Aggregate_Marks
1	1001	10	78
2	1110	10	93
3	1134	10	46

Fig. 10.5: One–to-one relationship

One-to-Many relationship

This is one of the most common types of relationship between the tables in a database. As the name says, in this type of relationship, one specific record of the master table has more than one corresponding records in the related transaction table. For example, one teacher can teach multiple students or multiple classes, or one person can sell multiple products. So we can say that there is a one to may relationship between a teacher and class or teacher and student or seller and products (Fig. 10.6).

Table : Teacher Details

Teacher_Id	Name	Subject
T002	Venkateshwari	English
T123	Shikha	Chemistry
T009	Alam	Geography
T056	Vaishali	Maths
T908	Sandhya	English
T667	Puneet	Computers
T342	Onkar	Hindi
T765	Gajender	Maths

Table : Class allocation

Multiple corresponding records

Class	Section	Subject	Teacher_Id
10	A	English	T002
10	A	Hindi	T342
10	B	English	T002
10	A	Maths	T765
9	C	Maths	T765
8	A	Maths	T765

Fig 10.6: One–to–Many relationship

Many-to-Many relationship

In this type of relationship, there will be multiple records in the master table that correspond to multiple records in the transaction table as well. Generally this type of relationship is set when certain records have to be saved more than once in both the related tables. For example, a teacher in a school may hold multiple responsibilities such as class teacher, an activity in-charge or examination in-charge. For each responsibility the teacher might be attached with multiple students. So this type of relationship will be many to many relationship. Similarly a shopkeeper may sell multiple products to multiple customers. So many-to-many relationship exists between a product and a customer. Fig. 10.8 shows the corresponding records in master table and transaction table are in same color.

Table: Department_Incharges

Department	Teacher_Id
Examination	T002
Discipline	T765
Co-Curricular	T056
Time Table	T002
Website Update	T765

Table: Activity_Duty

Activity	Department	Teacher_Id
UT1	Examination	T002
Term1	Examination	T002
HomeWork Upload	Website Update	T765
Timetable Upload	Website Update	T765
Inter-house Dance	Co-Curricular	T056
Inter-class Debate	Co-Curricular	T056
Discipline duties	Discipline	T765

Fig. 10.8: Many–to-many relationship

Advantages of Relating Tables in a Database

There are various advantages of relating tables in a database. Few of them are as given below.

- A relationship can help prevent data redundancy.
- It helps prevent missing data by keeping deleted data from getting out of synch. This is

called **referential integrity**. We will study in detail about referential integrity later in the chapter.

- Creating relationships between tables restricts the user from entering invalid data in the referenced fields.
- Any updation in the master table is automatically reflected in the transaction tables.

Let's Practice

Name the type of relationship for the following:

- Citizen and his Driving License
- Customer and Product
- Student and Course
- Team and Match
- Player and Country
- Employee and Project

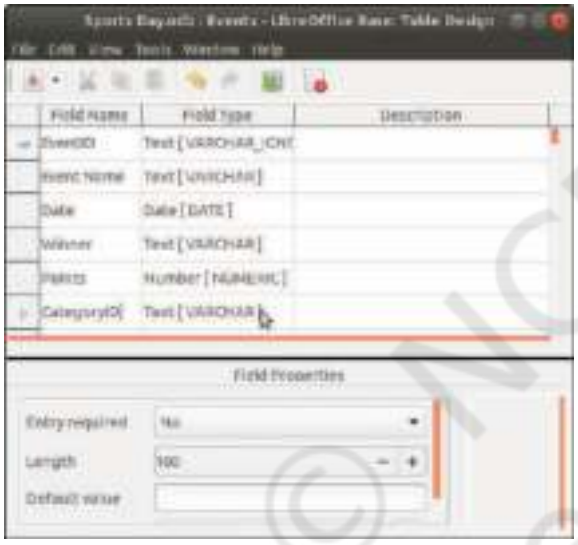


Fig. 10.9: Adding field CategoryID in Event table

Fig. 10.10: Records entered in Event table with added field CategoryID

Creating Relationships between Tables

Let us get back to Sports Day database created in the previous chapter. Let us add another field, CategoryID to the table *Events* using Edit option as shown in Fig. 10.9.

Enter 10 records in the Events table of various categories as shown in Fig. 10.10.

Create another table with the name EventCategory with the fields – CategoryID, CategoryName, TeacherIncharge and enter the records as shown in Fig. 10.11.

Hence, there are two tables in the database – Events and EventCategory with a common field as CategoryID.

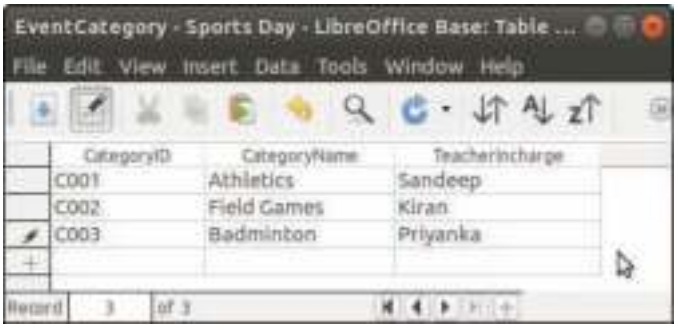


Fig. 10.11: Records entered in EventCategory table

In Event table, EventID is the primary key and CategoryID is the foreign key. In EventCategory table, CategoryID is the primary key. To set up relationship between these tables follow the following steps.

Step 1. From main menu of LibreOffice Base, click on Tools > Relationships…

Step 2. The Relationship Design screen will appear as shown in Fig. 10.12. In the middle of the screen there is Add Tables dialog box. Both the tables are listed in the dialog box.

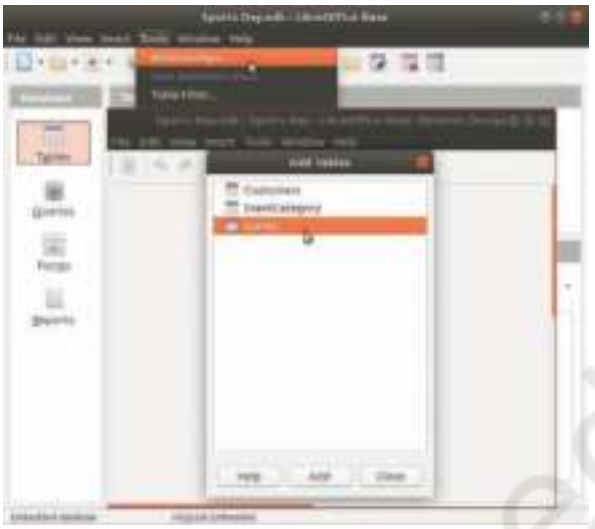


Fig. 10.12: Add Tables dialog box in Relationship Design Screen

Step 3. In the Add Tables dialog box, click Events table and then click Add button. Similarly add EventCategory table to the Relationship Area.

Step 4. Click Close button to close the Add Tables dialog box. Observe that the tables Events and EventCategory table added to the Relationship Area along with all its field list as shown in Fig. 10.13.

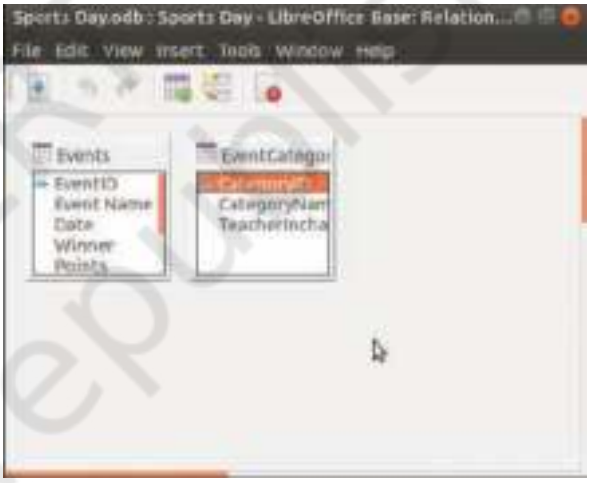


Fig. 10.13: Events and EventCategory Tables added in Relationship Design window

Step 5. As discussed before, CategoryID is the common field in the two tables. Hence it will be used to set a relationship between the two tables. To create a relation between the two tables, we just have to drag the common field CategoryID from the Events table and drop it in EventCategory table. A line connecting both the tables with the common field (CategoryID) appears on the screen as shown in Fig. 10.14.

Fig. 10.14: Relationship between two tables

The line is labeled as '1' on the primary key side and as 'n' on the foreign key side. Hence the CatgeoryID from EventCategory table as primary key will have unique values and is called referenced field. On the other hand,

Fig 10.15: CategoryID of Events table is referenced field and CategoryID of EventCategory table is referencing field

in the Events table, the values of CategoryID might be repeated. Here it is known as referencing field. Fig. 10.15 shows that the CategoryID of the Events table is referenced field and CategoryID of the EventCategory table is referencing field. This type of relationship where one value of a table is associated with multiple values in another table is a One-to-many relationship.

In this relationship EventsCategory is the master table. So the master record with a particular CategoryID has to be added in EventsCategory table first. Only then the corresponding record can be added in the Events table. Also, only one record with a specific CategoryID will exist in the master table but multiple values of CategoryID might exist in the transaction table, thereby forming a one-to-many relationship.

Referential Integrity

According to the principle of referential integrity, no unmatched foreign key values should exist in the database. That means if a record, say Admission No as 1001 is not present or deleted in the master table (Student_Details) of Student database, then there should be no record with Admission no as 1001 in the transaction table (Student_Result) as well. Similarly, in Ruhi's Sports Day database, if a particular category of sports, say C003 is deleted from the master table EventCategory, then there should be no record with Category as C003 in the transaction table. Likewise, if any student leaves the school and his record is deleted from Student_Details table, then there is no question of his appearing for exams and having a result. Hence corresponding record in the transaction table (Student_Result) should either have NULL value or should be deleted.

In Ruhi's Sports Day database as well, the CategoryID that exists in EventCategory table can only be entered in Events table. As mentioned before, once the relationship between the two tables has been set, the integrity of data will be managed by the DBMS. LibreOffice Base

will allow only that corresponding record to be entered in the transaction table which already exists in the master table.

LibreOffice Base gives us following four options to choose from to maintain referential integrity in such cases.

- **No action –** This is the default option. This option states that a user should not be allowed to update or delete any record in the master table if any related record exists in the transaction table.
- **Update cascade –** This option allows the user to delete or update the referenced field but along with it all the related records in any of the transaction tables will also be deleted or updated.
- **Set NULL –** This option assigns NULL value to all the related fields if the master record is deleted or updated.
- **Set default –** This option assigns any fixed default value to all the related fields if the master record is deleted or updated.

To set the relationship properties double click on the relation line joining the two tables, Events and EventCategory. A Relations dialog box will open as shown in Fig. 10.16.

By default the radio button with No action option will be selected. Choose any of the desired option and click OK to set the referential integrity between the two tables.

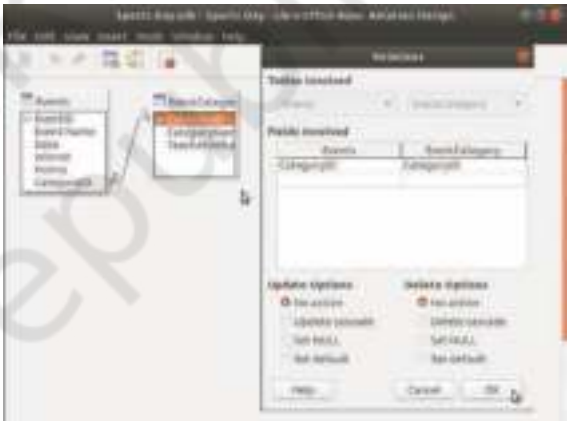


Fig. 10.16: Relations dialog box

Let's Practice

Form a table in Sports Day database in which one-to-one relationship can be established. Write steps to create such a relation in LibreOffice Base.

Summary

- Relations are set up between the tables to control data redundancy and inconsistency.
- The most important prerequisite for setting a relationship between the two tables is that there must be a common field(s) between the two tables.

- Three types of relationships can be set up between two tables in a relational database. These are One-to-One, One-to-Many and Many-to-Many.
- In One-to-One type of relationship, one specific record of a master table has one and only one corresponding record in the transaction table.
- In One-to-Many type of relationship, one specific record of the master table has more than one corresponding records in the related transaction table.
- In Many-to-Many type of relationship, there are multiple records in the master table that correspond to multiple records in the transaction table.
- According to the principle of referential integrity, no unmatched foreign key values should exist in the database.

Practical Exercises

Prakasan lives in Happy Home Society. He wants to create a database so as to store and manage the maintenance dues received from flat owners of the society. Create the following tables for his database.

Table: Residents Details

Flat no
Owner Name
Contact No
Flat Category

Table: Maintenance Dues

Receipt_Id
Flat No
Date
Amount

Relate the two tables and thereafter enter minimum five records in both the tables. Which type of relationship did you create? In which table did you enter the data first?

Uzair manages a boys hostel in an engineering college. He wants to manage the records of the people staying in his hostel by creating a database with the following tables and also set a relationship between the tables.

Table: Student_Details
Student_Id
Name
DOB
Course
Father's Name
Contact No_student

Contact No_ guardian
Room No
Table: Room Details
Room No
Floor (data value can be first, second or third)
Category (data value can be AC / Non- AC)
Student Id
DOO (Date of occupancy)
Monthly Rent

Check Your Progress

A. Multiple choice questions

1. Which of the following actions can be performed once the tables are created in a database?
   (a) Add a field in a table
   (b) Rename a table
   (c) Delete a table
   (d) All of the above
2. Which of the following is checked by a DBMS?
   (a) Redundancy
   (b) Inconsistency
   (c) Both (a) and (b)
   (d) Neither (a) nor (b)
3. Which of the following is required to set a relationship between the two tables?
   (a) Both the tables must be in different databases
   (b) Both the tables must have a common field
   (c) Both the tables must have the same name
   (d) Both tables must be stored in documents folder only.
4. If a record is added in a master table, which of the following is NOT true for transaction table
   (a) The record in the master table is called the master record
   (b) The corresponding record in transaction table can only be entered once.
   (c) The record in the transaction table is called the transaction record.
   (d) It is possible to add a record in the master table
5. Which type of relationship exists between a student and the subjects studied by him/her?
   (a) One-to-one
   (b) One-to-many
   (c) Many-to-many
   (d) All of the above

6. Consider the following tables. Which type of relationship can be established between the two tables?
   (a) One-to-one
   (b) One-to-many
   (c) Many-to-many
   (d) None of the above

Table 1: Item

Item_Code
Item_Name
Price
Qty
Category_Code

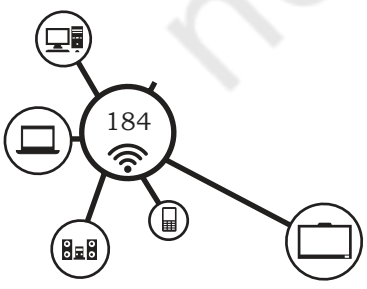
Table 2: Item_Category

Category_Code
Cat_Name
Item_Code
Cat_Disc

7. Which of the following menus contains the Relationship option?
   (a) Edit
   (b) File
   (c) Tools
   (d) View
8. The list of tables to be added is displayed in the _____ dialog box in the Relationship Screen.
   (a) Add Tables
   (b) Add Databases
   (c) Both (a) and (b)
   (a) Neither (a) nor (b)
9. In the relationship design screen, the relationship between the two tables is done using _____ operation.
   (a) Click
   (b) Double Click
   (c) Drag and Drop
   (d) Right click
10. Which of the following is NOT an option that can be used to maintain referential integrity in a database?
    (a) No Action
    (b) Set NULL
    (c) Set Default
    (d) Set Value

B. State whether the following statements are True or False

1. Redundancy is preferred in a database.
2. In a table, a record for a particular entity should not be repeated.
3. A single field should always have only one data value.
4. If a table is edited, the records already entered in it are deleted.
5. The record in master table should be entered before the corresponding record is entered in the transaction table.



6. In one-to-many relationship, one specific record of the master table has more than one corresponding records in the related transaction table.
7. The Relationship option is present in the Widows menu.
8. In a database, the referential integrity is maintained by the user.
9. A relationship is always set between the tables based on a common field.
10. If the master record is deleted, the transaction records will always be deleted.

C. Fill in the blanks

1. A table to be edited is displayed in _____ view.
2. The most important prerequisite for setting a relationship between the two tables is that there must be a _____between them.
3. In _____relationship, one specific record of a master table has one and only one corresponding record in the transaction table.
4. _____is one of the most common types of relationship between the tables in a database.
5. A record being entered in a _____ table must always exist in a _____ table.
6. The principle of _____helps prevent missing data by keeping deleted data from getting out of synch.
7. Creating _____ between tables restricts the user from entering invalid data in the referenced fields.
8. Data integrity is maintained by _____.
9. A relationship between customers and products is an example of _____ relationship.
10. The _____ window is used to set relationships between the tables.

D. Answer the following questions

1. Give any two advantages of relating a table in a database.
2. How is redundancy or inconsistency controlled in a database? Explain with an example.
3. Define referential integrity. Who maintains referential integrity in a database?
4. Differentiate between one to one relationship and one to many relationship. Give suitable examples to explain your answer.
5. Explain many to many relationship with an example.

CHAPTER 11

Queries in Base

Introduction

A database is used to store data in an organized manner so as to retrieve it easily and accurately from database. To search for the desired record and to retrieve the desired data, we have to give its specifications to DBMS. Such specifications are given to the database in the form of queries. For example, we may have to specify the fields that we want to display or any particular data value based on which the records are to be filtered from the table(s). Therefore, we can say that a query is a sort of question asked from a database. Depending upon specifications given in the query, the specific records are searched from the table(s) in the database and then displayed in the desired manner. Such information may be difficult to find by just looking at a single table or multiple tables. In fact as the number of records increase, finding the desired information becomes more and more difficult. By using queries, we are able to retrieve data without going individually through each record in the table(s) and also display them in desired format.

Queries

A query is one of the most important feature of any DBMS. Using a query, we can retrieve and display data from one or more tables in a database. This is done by giving specific search criteria to the DBMS so that we are able to view the exact information that we want.

LibreOffice Base allows us to create a query and even save it as an object in a database. This helps us to run the query multiple times as and when required.

Using a query, we can specify the fields that we want to display and also the criterion based on which the records to be filtered. The information may be retrieved from a single table or from multiple tables. Also the result of the query is displayed in tabular form with field names in columns and the records in rows.

For example, in a Student database, if we want to display the names and marks of the students in a particular class from the Result table, who have scored less than 320 aggregate marks, then we need to give the following information to the database:
Name of the table – Result
Fields to be specified – Name, Marks
Criterion – Aggregate marks should be less than 320

In this chapter you will learn to create a query, save it and then run it as and when required.

Creating a Query

A query can be created in three ways. In this chapter you will learn the first two methods to create a query.

(i) Using a Wizard
(ii) In Design View
(iii) In SQL view

Creating a Query Using a Wizard

Let us recall the *Sports Day* database created in the previous chapters with two tables – Events and EventCategory.

Let us create a query that will display the Event Name and Winner for each event with *CategoryID* as *C001*. The desired data has to be retrieved from the Events table. To create a *query using a wizard*, follow the following steps.

Step 1. Open the *Sports Day* database. In the **Database Design** window, click on **Queries** button present in the **Database Pane** on the left as shown in Fig. 11.3.

Step 2. In the **Tasks Area**, click on **Use Wizard to Create Query...** option. The **Query**

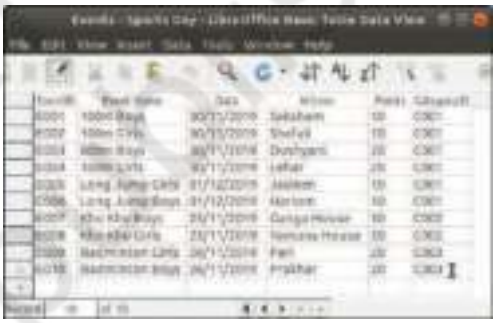


Fig. 11.1: Records in Events table

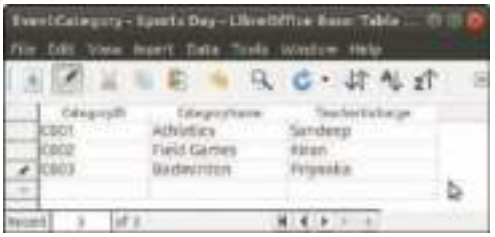


Fig. 11.2: Records in Event Category table

Fig. 11.3: Queries icon in Database Pane

Fig. 11.4: First step of Query wizard

Fig. 11.5: Fields of selected Events table displayed in the Available Fields list box

Fig. 11.6: Fields moved from Available fields to Fields in the Query list box

Fig. 11.7: Select sorting order

Wizard will start as shown in Fig. 11.4. It contains the **Steps Pane** on the left and the **Query Details Area** on the right.

Step 3. The first step of the wizard is to select fields from the respective tables. For our query, to display *Event Name* and *Winner* and the criterion to filter the records based on *CategoryID*. All these fields are present in *Events* table. Therefore select *Events* table from the **Tables** list box. The corresponding fields are displayed in the **Available Fields** list box as shown in Fig. 11.5.

Step 4. Select Event Name field from the list box and click the right arrow (>) button. The field name will appear in the **Fields in the Query** list box. Repeat the process for *Winner* and *CategoryID* fields as shown in Fig. 11.5. Observe that these three fields are moved to **Fields in the Query** list box.

Note that once the fields are added they can be moved up and down in order by clicking ∧ and ∨ buttons present on the extreme right of the wizard. Clicking on » button moves all fields to **Fields in the Query** area and « button moves all the fields back to **Available fields** area.

Step 5. Clicking on the **Next** button will display the screen to select the sorting order as shown in Fig. 11.7. It will set the sorting order. The result of the query can be displayed in ascending or descending order of any particular field of the table. Since we do not want to set in a particular order, so we click on **Next** button.

Step 6. The next step is to set the search conditions or the criteria on the basis

of which records will be filtered from the table. This is the step where actually the query is set up or the criterion is given to the database. As per our query, the criterion is to display the records of events with *CategoryID* as *C001*. By default, the radio button with option **Match all of the following** is selected.

Step 7. Select *Events.CategoryID* field from **Fields** drop down list, is equal to from **Condition** drop down list and type the value as C001 as shown in Fig. 11.8, and click on Click **Next** button.

Fig. 11.8: Selecting the Search condition and specifying value

Three search conditions can be given at the most in the wizard. Click and select the radio button with option **'Match any of the following'** if any one of the given condition are to be matched for filtering the records.

Step 8. Steps 4, 5 and 6 given in the **Steps Pane** deal with tasks like summarizing and performing numerical calculations. Such steps are not required if there is no numeric field involved in the query. So you can skip these steps and move directly to "*Step 7. Aliases*" of the wizard.

Step 9. The next step to give alias name i.e. the column header name will be displayed when we run the query. By default the field names will be displayed as column headers. Many times field names are not user friendly, so an alias name which is more readable, is chosen to be displayed in the query output. For example, the name of the field in the *Events* table is *Winner*. To display the name of winner as the column header, type '*Winner Name*' and click on **Next** button as shown in Fig. 11.9.

Fig. 11.9: Assigning Alias Names

Step 10. The last step of the Query wizard displays the entire overview of the query as shown in Fig. 11.10. All the steps performed till now are shown in a summarised manner.

Fig. 11.10: Overview of Query

It includes the following:

- **Name of the Query –** By default, the name of the query is Query_Events by default. If desired, type the new name in the text box.
- **The action to be performed after the wizard finishes –** By default Display Query option will be selected. Click and select the Modify Query radio button if the query has to be edited in the Design view.

Fig. 11.11: Records retrieved from Query with CategoryID C001

- **Complete detail of the query –** This section contains a summary about the query that has been created.

Step 11. Click on **Finish** button. The records with *CategoryID* as C001 will be displayed on the screen as shown in Fig. 11.11. Observe the column headings. Are all of them same as the field names in the *Events* table?

Once the query is created, it can be edited in Design view. We will study about Design View in the coming section.

Creating a Query in Design View

Another way to create a query is using the **Design** view. This is a more flexible method to create a query from either single or a multiple tables of a database. Let us create a query to display records of Athletics category. For this query, records have to be filtered from both *Events* and *EventCategory* tables. Open *Sports Day* database and follow the following steps to create a query in Design View:

Fig. 11.12: Create Query in Design View

Step 1. Click **Queries** icon on the **Objects Pane** in the **Database Window**.

Step 2. Click **Create Query in Design View...** icon in the **Tasks Pane**. The **Query Design Window** appears. In the middle of the window the **Add Table or Query** dialog box is displayed as shown in Fig. 11.12.

Step 3. Click on the Event table to be used in the query and then click on **Add** button. Alternatively double click on

the *Events* table. The table will be added to the Tables Pane present at the top of the **Query Design window**.

Step 4. Similarly add *EventCategory* table to **Tables Pane** of the **Query Design window**.

Step 5. Click **Close** button in the **Add Table or Query** dialog box to close it. The tables *Events* and *EventCategory* added will be displayed in the Table pane as shown in Fig. 11.13.

Fig. 11.13: Table added in the Tables Pane

Step 6. Next step is to select the fields. For our query we want to display *Event Name* and *Winner* from the *Events* table and *Category Name* from the *EventCategory* table. So in the list box of *Events* table, double click on *EventName* and *Winner* field. Similarly add the field *CategoryName* form *EventCategory* table. The field name along with the table name is displayed in the **Design grid** present in the lower half of the **Query Design window** as shown in Fig. 11.14.

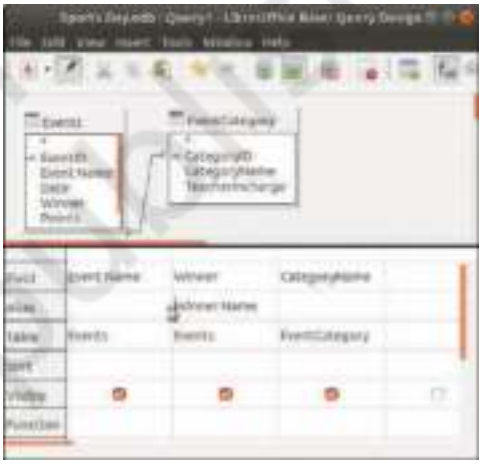


Fig. 11.14: Adding fields to query

Observe that the **Visible Check Box** is by default selected. This means that all these three fields will be visible when you run the query. If you do not want the data values for the particular field to be displayed, click to deselect the respective check box.

Step 7. In the grid, there is a row titled **Alias**. As mentioned before, it can be used to display meaningful names in the output. For example, instead of Winner, we would just like to display Winner Name. For this, type Winner Name in the **Alias** text box under Winner column as shown in Fig. 11.14.

Step 8. By default, the data that is displayed as a result of the query is not sorted. To sort the records in either ascending or descending order of a particular field, the **Sort** row is given in the grid. For example, to display the records in alphabetical order of *Event Name*, select *Ascending* from the drop down list box visible in the **Sort** row under Event Name column.

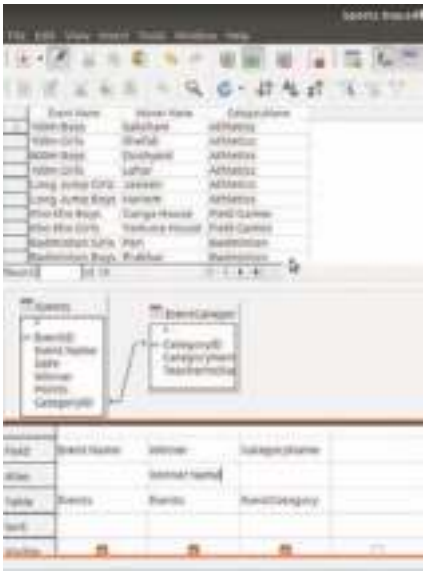


Fig. 11.15: Query Result in Tables Pane

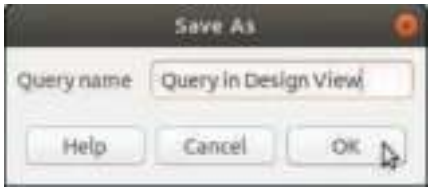


Fig. 11.16: Saving the query

Fig. 11.17: Selecting Edit option to edit query

Step 9. Once the query is designed, click **Run Query** ( ) button on the toolbar or press **F5** key. The query result will be displayed in the **Tables Pane** area as shown in Fig. 11.15.

Step 10. Click on **Save** button to save the query. The **Save As** dialog box will be displayed as shown in Fig. 11.16.

Step 11. By default, the Query Name as *Query1* will be displayed. Type a different name if required. Click on **OK** button to save the query.

The name of the query will be seen in the **Objects area** in the **Database window**. To run the query again and see the result s of the query, double click on the query name. The results of the query will be displayed in a separate window.

To close the Query window, click on on close button on the top right corner of the window.

Editing a Query

To edit any query, click on the **Query** icon of the query that has to be edited. The list of queries that have been created will be displayed in the **Objects Area**. Right click on the Query Name in the **Objects Area** of the **Database window**, as shown in Fig. 11.17.

Select **Edit** option from the drop down menu. The **Query Design window** will be displayed. Let us apply a criteria to display records of only Athletics category. For this type *Athletics* in the **Criterion row** under the *Category Name* column. Now, the query design is complete if we run the query, the record of only *Athletics* category will be displayed as shown in Fig. 11.18.

We can use the Criterion row to apply multiple conditions as well. Also note that we can apply all relational operators like <, >, <=, >=, != and = for all conditions that can be given in **Criterion** row. For example, if you want to see only those records where points scored are more than 10, then add Points field to the grid and then set the Criterion for it as >10 as

shown in Fig. 11.18. Now save and run the query. Observe the output as shown in Fig. 11.18.

Fig. 11.18: Query result in Design view after applying criteria to display records of Athletics category

Working with Numerical Data

Till now, we have been displaying data from tables in the query. But that is not all, we can even use certain mathematical functions to find the count, sum, minimum, maximum or average of data values. Let us design a query to display the average points for each category of events. In the **Database Window**, click **Create Query in Design View...** button to open the Query **Design Window**. Add Events and *EventsCategory* tables. Thereafter follow the following steps to display the average points for each category.

Step 1. Add Category Name field from *EventCategory* table and Points field from *Events* table.

Step 2. Under the *Category* Name field, in the **Function** row, click the down arrow. A drop down list will be displayed as shown in Fig. 11.19.

Step 3. Select **Group** option from the drop down list. This option forms the groups for each of the data values in the column of *Category Name*.

Step 4. Similarly, under the Points column, select the **Average** function from the drop down list.

Step 5. Press **F5** to run the query. The query result depicting average points in all the categories will be displayed as shown in the Fig. 11.20.

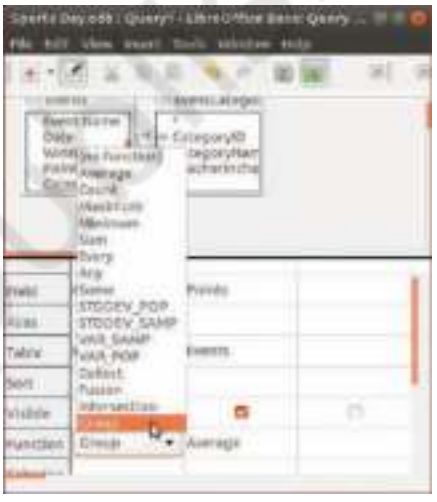


Fig. 11.19: Function drop down list

Fig. 11.20: Query result in Design view after applying Group function to Category Name and Average function to Points

Step 6. Further, we can also put conditions on groups using the **Criterion** row. Say, for example to display the average of only *Athletics* category, type *Athletics* in the **Criterion** row under the *Category Name* column. Now, if we run the query the average of only *Athletics* category will be displayed in the query result.

SUMMARY

- A query is used to retrieve and display data from one or more tables in a database.
- A specific search criteria is given to the DBMS to view the desired information.
- The result of the query is displayed in tabular form with field names in columns and the records in rows.
- A query can be created in three ways.
  - Using a Wizard
  - In Design View
  - In SQL view

Practical Exercises

Prakasan lives in Happy Home Society. He has created the following tables in his database so as to store and manage the maintenance dues received from flat owners of the society.

Table: Residents Details

Flat no
Owner Name
Contact No
Flat Category

Table: Maintenance Dues

Receipt_Id
Flat No
Date
Amount

Create a query to display the following information

Flat No	Owner	Receipt Number	Date	Amount

(Please note the change in column headings)

Uzair manages a boys hostel in an engineering college. To manage the records of the people saying in his hostel, he created a database with the following related tables.

Table: Student_Details

Student_Id
Name
DOB
Course
Father's Name
Contact No_student
Contact No_ guardian
Room No

Table: Room Details
Room No
Floor (data value can be first, second or third)
Category (data value can be AC / Non- AC)
Student Id
DOO (Date of occupancy)
Monthly Rent

Create a query to display the following tabular format

Room No	Student Name	Category	Monthly Rent

Create a query that displays total monthly rent collected for each category.

Project Work

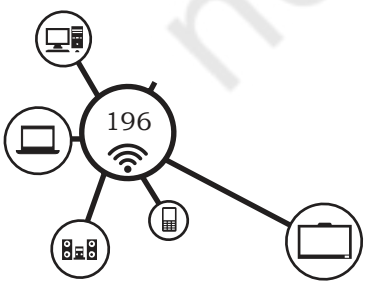
Use the internet to study the process of working of hostels. See, what all tables/fields/queries should be further added to the existing database so that it is able to fully automate the working of the boys hostel explained in the above exercise. Thereafter make changes in the database created by you in LibreOffice Base.

Check Your Progress

A. Multiple choice questions

1. Which of the following is refer to asking questions from the database?
   (a) Report
   (b) Table
   (c) Query
   (d) Database
2. Which of the following are the ways to design a query?
   (a) Wizard
   (b) Design View
   (c) SQL
   (d) All of the above
3. Which is a flexible way to create a query?
   (a) Wizard
   (b) Design View
   (c) Both (a) and (b)
   (d) Neither (a) nor (b)

4. Into how many parts is the query design window divided?
   (a) One
   (b) Two
   (c) Three
   (d) Four
5. Which of the following is NOT true about queries?
   (a) It can be created using multiple tables
   (b) Multiple queries can be created in a database
   (c) A query can run multiple times
   (d) A query once created cannot be edited
6. Which of the following is the shortcut key to run the query?
   (a) F3
   (b) F4
   (c) F5
   (d) F6
7. Which of the following functions can be performed on numerical data while designing a query?
   (a) Sum
   (b) Minimum
   (c) Maximum
   (d) All of the above
8. In a Query Design wizard, which of the following buttons is clicked to move a field from 'Available fields' list box to 'Fields in the query' list box?
   (a) >
   (b) < 9
   (c) ∨
   (d) ∧
9. Which of the following relational operators can be applied to set the criterion while designing a query in LibreOffice Base?
   (a) >
   (b) =
   (c) !=
   (d) Add Form
10. Which of the following dialog box is present when the Query Design window is opened for the first time to design a query?
    (a) Add Table
    (b) Add Query
    (c) Add Table or Query
    (d) None of the above
11. Which of the following step is not performed if there is no numerical data to be worked upon in a query?
    (a) Selection of fields
    (b) Giving Aliases
    (c) Summarizing
    (d) Selection of tables



B. State whether the following statements are True or False

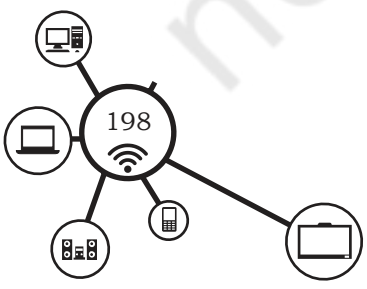
1. You can run a query only once
2. A query cannot be created from multiple tables
3. The shortcut key to run a query is F5.
4. LibreOffice Base provides us with two ways to create a query.
5. A query with numerical data cannot be saved.
6. By default the query result is not sorted.
7. A query can be used to display the average value of a numerical field.
8. While designing a query, the criterion can be set on only one field.
9. Alias is an alternative name for a field in a query.
10. In query Design window, the visible check box is selected by default.
11. A query once created using a wizard can only be edited in the Design view.

C. Fill in the blanks

1. A _____ is a sort of question asked from a database.
2. The result of the query is displayed in _____ form with field names in columns
3. A query can be created in _____ways.
4. The Query Design window is divided into _____ sections.
5. The shortcut key to run the query is _____.
6. The conditions to filter the records are set in the _____ row.
7. When a table is selected in a Query wizard, the corresponding fields are displayed in the_____list box.
8. The result of the query can be displayed in _____ or _____ order of any particular field of the table.
9. At the most _____ search conditions can be given in the query wizard.
10. The last step of the Query wizard displays the entire _____ of the query.
11. The _____ view is a more flexible method to create a query.
12. To edit any query, right click on the _____ icon of the query that has to be edited.
13. In the _____ row of the Query Design grid, we can type the column heading that will be displayed instead of field name when we run the query.

D. Answer the given questions

1. Define a query? What is the need of creating a query in a database?
2. Rearrange the steps given below so as to create a query using a wizard.
3. Give Alias
4. Select the fields
5. Set the criterion
6. Set the sorting order
7. Give table name
8. What all information is seen in the overview (last step) of the Query wizard?
9. What is the use of Alias row in the Design grid of the Query Design window?
10. Name any four mathematical functions that can be applied to numerical data in a query.
11. Name the three ways of creating a query in LibreOffice Base?



Chapter 12 Forms and Reports

Introduction

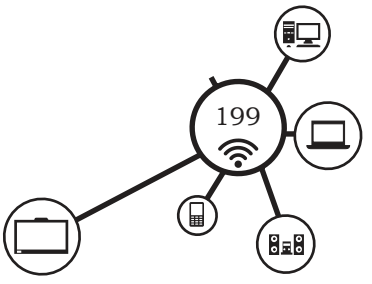
We know that tables in a database are used to store data in an organized manner. For this we have to enter data into the tables. As seen in previous chapters, data-sheet view is used to enter data into the tables. The row and column format of the data-sheet view is not a user friendly interface. So it is required to develop a user friendly data entry screen for data entry operator or a user, who is not very well versed with computers. Moreover, while doing data entry, the user may accidentally or intentionally alter the data in the table. Therefore LibreOffice Base provides the **Form** feature for data entry purposes.

Also, when the user wants to retrieve and view the data from one or more tables in a database using queries, it is again displayed in data-sheet view. The **Report** feature of OpenOffice Base helps to present the retrieved data in a user friendly, understandable and formatted manner.

Both reports and forms are considered as objects of the database and are present in the **Database Pane** of the **LibreOffice Base User Interface**. In this chapter we will learn to create Forms and Reports using LibreOffice Base.

Forms

A form is an object of the database that has a user friendly interface where data can be entered and seen in an attractive and easy-to-read format. For any database, it is the front end for data entry and data modification. It displays the data in a layout design by us and not just in a simple row and column format.



Primarily, a form contains **field controls** arranged in a presentable and user friendly manner. Each field control consists of a label and the field value text box. A label is a piece of text that specifies the data that should be entered in the field value text box. A field value text box is linked to the respective field in the table. We may add all or selected fields from the table on the form. In addition to field controls, it may contain some additional text like titles, headings and names, graphics like logos, list boxes and radio buttons.

There are two ways to create a form:

- Using a wizard
- Using the Design View

Creating a Form Using a Wizard

This is the simplest way to create a form. To explain how to create a form using a wizard, we will get back to the Sports Day database created in the previous chapters. To create a form using wizard, follow the following steps.

Step 1. Open the Sports Day database created in LibreOffice, and click the **Form** icon on the **Database Pane** as shown in Fig. 12.1.

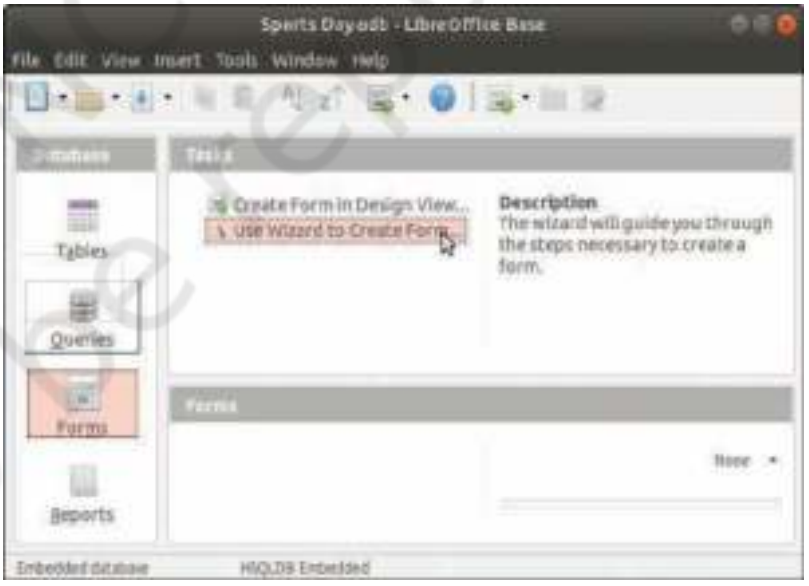


Fig. 12.1: Selecting Forms from Database Pane

Click the option **Use Wizard to Create Form...** on the **Tasks Pane**. The **Form wizard** will open along with a blank database form in design view in the background as shown in Fig. 12.2.

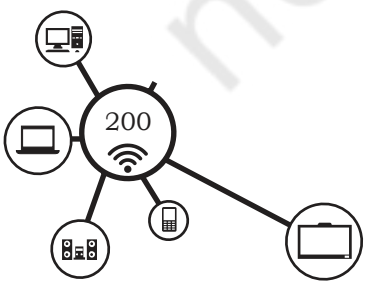


Fig. 12.2: Form Wizard and Database Form in Design View

Step 2. The step 1 of the wizard is to select the tables or queries for which the form has to be created. As we are creating a form for *Events* table, select Events table from "*Tables and queries*" list box as shown in Fig. 12.2.

Step 3. After selecting the *Events* table, all the fields of the *Events* table will be listed in the **Available Fields** list box as shown in Fig. 12.3.

Fig. 12.3: Fields of Events table listed in the Available Fields

Step 4. As we require all the fields to appear in the Form, shift all the fields of Event table from **Available Fields** list box to **Fields in the Form** list box using >> button. Observe that in Fig. 12.4, all the fields are shifted to **Fields in the Form** list box. Click on **Next** button to move forward.

Fig. 12.4: All fields of Events table in Available Fields are shifted to Fields in the Form

Step 5. The second step consists of setting up a subform, i.e. a form within a form. You need to check the checkbox "Add Subform" in Fig. 12.5 to add the subform. Since we do not want to set up any subform, click **Next** button to proceed further.

Fig. 12.5: Step 2 Set up a subform

Step 6. The wizard skips the next two steps that relate to the subform and moves on to step 5. This step arrange controls i.e. to set up the design of the form. Observe in Fig. 12.6 that, by default, all controls will be left aligned. Click **Right Align** radio button to align the controls from the right side of the form. As mentioned before, a field control consists of two parts – label and the field value text box. So in this step we arrange the label and field value text boxes as we want them to be visible on the screen.

Fig. 12.6: Selecting the Layout of the form

Four layouts are given in this step of the wizard to choose from:

- Columnar display with Labels on the left of the field value

- Columnar display with Labels on top of the field value
- Display as datasheet
- Block display with labels on top

Let us select *Columnar Display with labels on the left* arrangement. Also note that as we choose the Layout type, the fields are arranged in the Form Design view also.

Step 7. Click **Next** button.

Step 8. The step 6 of the wizard asks whether the form will be used for displaying data, entering data or both. As we go with the default settings, so we click **Next** button as shown in Fig. 12.7.

Fig. 12.7: Step 6 of form wizard

Step 9. The next step is to apply styles to the form being created. We can select the desired background colour and border type of the field value text boxes. Observe in Fig. 12.8, by default the border of the field text value is displayed in *3D look*. We can select the options *No Border* or *Flat* if required.

Fig. 12.8: Applying styles to the form

Step 10. Click **Next** button.

Step 11. The next step is to set the name of the form. By default the name of the form is same as the name of the table. Click in the textbox and type a different name if desired, say *EventsForm*,

as shown in Fig. 12.9. In the same step, by default, the radio button with the option **Work with the form** is selected. If you wish to modify the form after the wizard finishes, click **Modify the form** option.

Fig. 12.9: Assigning name to form

Step 12. Click **Finish** button. The form with the first record will be displayed on the screen in a separate window. Also, observe that in Fig. 12.10, the name of the form, *EventsForm* will appear in the **Object Area** of LibreOffice User Interface window.

Fig. 12.10: Form created for Events table

As shown in Fig. 12.10, on the left of the Form Design window is the **Forms Control toolbar** and at the bottom is the **Records toolbar**.

Forms Control Toolbar

This toolbar contains various controls that can be added to the form. We will learn about some of the controls later in the chapter.

Records Toolbar

The Records toolbar contains the navigation control buttons in the extreme left. With the help of these

buttons, we can traverse and view the records in the file. As we move from one record to another, the record number in the record text boxes changes. Fig. 12.11 shows the navigation control buttons, there are commands to add a new record, save a record, delete a record and so on.

Fig. 12.11: Records toolbar

Modifying a Form

It is possible to modify the form in any manner once it is created. The modification can be to change the background color, font size and color of the text or even positioning of various controls in the form.

Changing the background color

Follow the following steps to change the background color of the form.

Step 1. Open **OpenOffice Base User Interface** for *Sports Day* database and click on **Forms** icon in the **Database Pane**. The name of the saved form (*EventsForm*) will be displayed in the **Objects Area**.

Step 2. Right click on the form name and select **Edit**... option. A separate **Form Design View** will open.

Step 3. To change the background color of the form, right click on the form and select **Page Style**... option from the pop up menu as shown in Fig. 12.12.

Fig. 12.12: Selecting the Page Style option from the Edit pop down

Step 4. The Page Style dialog box will appear, where you select **Area** tab and choose the desired color from the palette, as shown in Fig. 12.13.

Fig. 12.13: Selecting Colors Palette in Page Style dialog box

Step 5. Click on **OK** button. The selected color will be applied on the form as shown in Fig. 12.14.

Fig. 12.14: Forms with changed background color

Editing the labels

To edit the labels, either by changing the text or by changing the formatting effects, follow the following steps.

Step 1. Place the mouse pointer over the label, say *EventID* to change it.

Step 2. Press the keyboard shortcut key ***Ctrl+Click*** to select the label. The position boxes will appear around the label.

Step 3. Right click on the selected label and select **Control Properties...** option from the pop up menu as shown in Fig. 12.15.

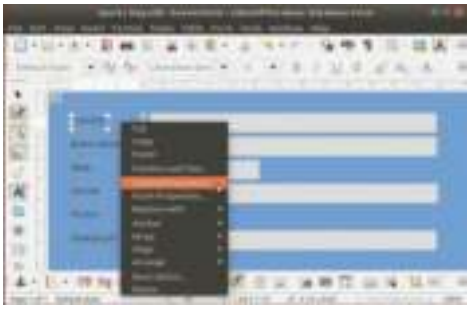


Fig. 12.15: Selecting Control Properties by right clicking on the label

Step 4. The Properties: Label Field dialog box will appear as shown in Fig. 12.16. It contains various properties of the selected label. In the text box after **Label** property, type *Event ID*. The label caption on the form changes accordingly.

Fig. 12.16: Changing the label caption from the Properties Label Field

Similarly we can change other properties of the selected label like width, height, alignment, font style and font size.

Step 5. After making the desired changes close the **Properties** dialog box by clicking the cross (x) button on the top right of the dialog box. The changes made will be applied on the selected text.

Moving a control

It is possible to move the to another location in the form. Click on the control that has to be moved. Both the label and the text box for field value will be selected and position handlers will be placed around the control. If only one of these have to be re-positioned then press **Ctrl** key while clicking on that control. Now, click and drag the control to move to the desired location (Fig. 12.17).

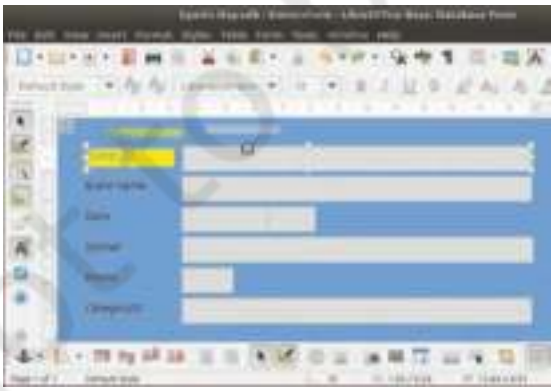


Fig. 12.17: Moving a control

Changing the size of the textbox control

To change the size of the textbox control, press **Ctrl** button while clicking on the textbox. It will be selected with position handlers around it. Place the mouse pointer on any of these handlers. It will change to a double sided arrow. Click and drag them to the desired size.

Adding a Tool tip

A tool-tip is a small piece of text that is displayed when the mouse pointer is placed on a particular control. Say for example, if the mouse pointer is placed over *EventID text-box*, a message '*Enter Event Identification Number*' can be displayed. This will make entering data for the user easy. Such type of text that appears when the mouse pointer is placed on a particular control is called the tool-tip **text** or **help text**. Let us add a tool-tip for *EventID* control by following the steps given below.

Step 1. Press **Ctrl** button and click on *EventID* text box.

Step 2. Right click and select **Control Properties**... option. The **Properties: Text Box** dialog box will be displayed as shown in Fig. 12.18.

Fig. 12.18: Adding a tooltip in Properties Text Box

Step 3. In the dialog box, scroll down till **Help Text** property appears.

Step 4. Type '*Enter Event Identification number*' in the text box.

Step 5. Close the dialog box by clicking on cross (X) button. Now, if the mouse pointer is placed over *EventID* text box, the tool tip will be displayed as shown in the Fig. 12.19.

Fig. 12.19: Displaying tool tip

Forms Controls Toolbar

The forms control toolbar contains various tools to add or edit controls on the form. Let us learn to use few of these tools.

Adding a calendar for the date field

While filling up a form on a computer, mostly a calendar is displayed. This is because it is easy to choose a date rather than typing it. To add the calendar to the date field in the form, follow the steps given below:

Step 1. Place the mouse pointer over the Date text box and press ***Ctrl+Click*** to select it.

Step 2. Right click and select **Control Properties....** option.

Step 3. In the **Properties: Date Field** dialog box, scroll down for **Date Format** property. By default, *Standard* (*short*) format will be displayed.

Step 4. Click to open the list box and select *Standard* (*long*) format.

Step 5. Scroll down further till you find the **DropDown** property. By default its value will be No. Select Yes. The control properties set is shown in Fig. 12.20.

Fig. 12.20: Control Properties set for Date field

Step 6. Close the dialog box. The selected date control text box on the form changes to a list box with an arrow being displayed in the extreme right as shown in Fig. 12.21.

Fig. 12.21: Date Control field with Calender

Adding text to the form

While designing a form, we may need to enter titles, headings or subheadings. It is called as Labels. It is

Fig. 12.22: Selecting Label icon on Form Controls tool box

Fig. 12.23: Properties Label Field dialog box for the created label

Fig. 12.24: Setting font type, style and size in Character dialog box

possible to create Labels in the form while designing. Follow the following steps to insert the title text in the form.

Step 1. Click the **Label** ( ) tool on the Form Controls tool box as shown in Fig. 12.22.

Step 2. On the form, click and drag the mouse to create a label field box. It will also have position handlers.

Step 3. Double click on box to open the **Properties: Label Field** dialog box.

Step 4. Type the title as "*Data Entry Form*" in the **Label** property, as shown in Fig. 12.23.

Step 5. Set the **Font** property by clicking the **Font** button in front of the **Font** property. The **Character** dialog box will be displayed as shown in Fig. 12.24, where you can set the font type, style and size. Choose the desired font style and size and click on **OK** button.

Step 6. Close the **Properties: Label Field** dialog box. The title with the selected formatting effects will be displayed on the form.

After applying and editing various controls, the final form for entering data in the Events table looks as shown in Fig. 12.25.

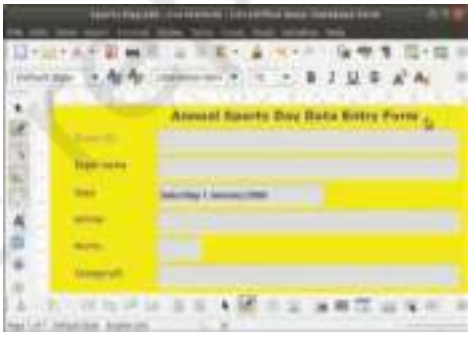


Fig. 12.25: Modified Data Entry Form for Events table

Adding a new record using a form

After you have finished designing the form, you can display or insert records using this form. For this purpose, you have to shift from **Design View** to **Form View** by clicking on **Design Mode** button on the **Forms Controls** toolbar. The **Form View** window appears with the first record displayed in the respective text boxes as shown in the Fig. 12.26.

Fig. 12.26: Displaying form in Design view by clicking on Design Mode button on Form Controls

To add a new record into the table using this form, click on **New Record** button on the **Records** toolbar as shown in Fig. 12.27.

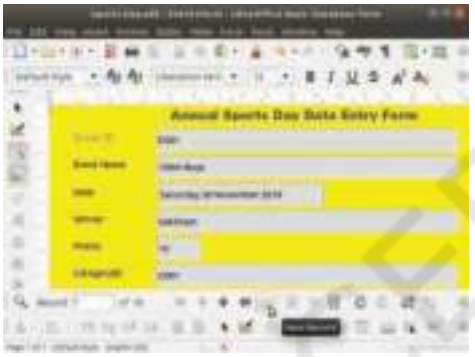


Fig. 12.27: Adding a new record by clicking on New Record

A blank form with the cursor blinking in the first text box i.e. *EventID* will be displayed.

Fig. 12.28: New blank record showing the next record number

Observe that the record pointer shows the record 11, as there were already 10 records in the table. Once the record is entered, click on **Save Record** button on the **Records** toolbar to save the record in the table.

To delete any record, navigate to the record by either typing the record number in the **Record** text box or by

using the navigation buttons. Thereafter click **Delete** button on the **Records** toolbar. The next record will be displayed in the form.

To toggle between Design view and Form view press Design Mode button on the Forms Controls toolbar.

When we create a form in Design View, all the controls are placed on the form using various tools given in the Forms Control toolbar.

Reports

A report is another useful feature of a database management system. We have seen that the records that have been extracted using a query are displayed in a simple row and column format. Instead, using a report we can present the retrieved data in an attractive and customized manner. We can create a report based on a table or a query or both. Preferably if a report has to be generated from multiple tables, a query should be created first and then that query can be used to generate the report.

Let us create a report using the table *Events* from the *Sports Day* database. Follow the following steps to create a report.

Step 1. In the **LibreOffice Base User Interface**, click on the **Reports** icon in the **Database Pane**.

Step 2. From the **Tasks Pane**, click **Use Wizard to Create Report**... option.

Step 3. The **Report wizard** along with two other windows will be displayed. One of the window is **Report Builder** window and the other is **Add Field** dialog box. We will confine our study to the wizard (Fig. 12.29).

Fig. 12.29: Report Wizard, Report Builder and Add Fields dialog box

Step 4. The first step of wizard is to select the table and the corresponding fields that we want to

display in our report. From the Tables or Queries list box, select the table *Events*.

Step 5. All the fields of the Events table will be listed in the **Available Fields** list box. Click >> button to shift all the fields to **Fields in report** list box. Fig. 12.30 shows the **Fields in report** list box after shifting.

Fig. 12.30: Selecting table Events and fields after shifting to Fields in report list box

Step 6. Click on the **Next** button. The next step is to label the fields. By default, the column headers will be displayed as labels or column headers for the field values. As fields names are generally shortened, to change to more self-explanatory names, type the new names in the respective text boxes.

Fig. 12.31: Step 2 of wizard Labelling the fields

Step 7. Click on the **Next** button. The next step is to group the data based on any of the fields in the report. Since we do not want to group as of now, click on the **Next** button.

Step 8. The fourth step is to set the Sort options. If the data to be displayed in the report has to be sorted in either ascending or descending order of a particular field, specify the field and sorting order in this step. Say, for example, select the field *EventID*. The radio button for *Ascending* is already selected. Select Descending radio button to display the records in descending order as shown in Fig. 12.32.

Fig. 12.32: Setting the Sorting options

Step 9. Click on the **Next** button to move on to the next step in which the layout of the report will be selected.

Step 10. A layout is the manner in which the labels, field values and titles will be displayed in the report. Out of various Layout options given, choose the desired layout, say *Tabular* and also the layout of headers and footers (*Default*). You may also choose the orientation option **Landscape** or **Portrait** in this step.

Fig. 12.33: Choosing the Layout

Keep the default orientation option *Landscape* selected by default as shown in Fig. 12.33.

Step 11. Click on the **Next** button to move to last step. Here we name the report and to specify the manner in which we want to proceed after the wizard finishes. Type the name of the report as *EventsReport*, as shown in Fig. 12.34. By default the type of report is **Dynamic**. That means as the field values in the base table or query change, the report will also change automatically. If you don't want automatic updation of the report, choose the **Static** option. In this step, specify whether you would like to modify the report or create the report once the wizard finishes. Select the default option to create the report with current setting.

Fig. 12.34: Step 6 Create report of the wizard

Step 12. Click on **Finish** button to display the report as shown in Fig. 12.35.

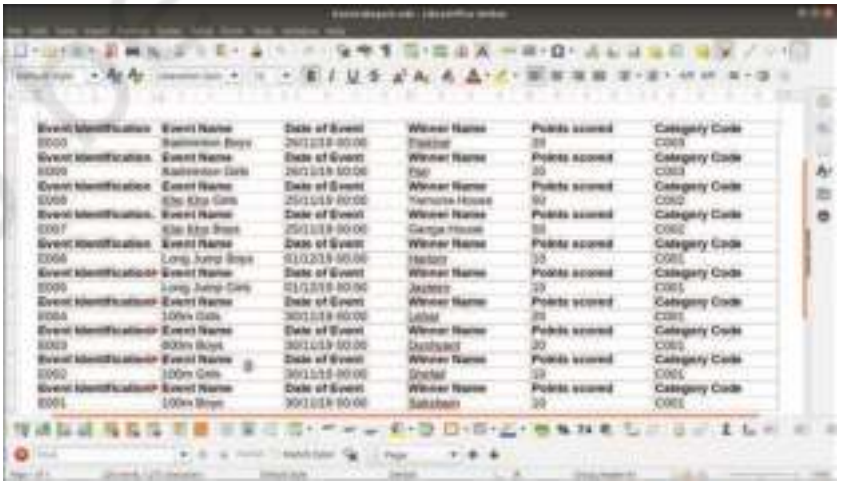


Fig. 12.35: Report generated from Events table

Inserting other controls in the report

As you can observe the report generated in Fig. 12.35, the report is very simple and boring. We can make it more presentable by inserting some more controls like titles, author name, date of generation of report etc.

Right click on the Report name (*EventsReport*) on the **LibreOffice User Interface** and then select the **Edit**... option. The **Report Builder** window will open as shown in Fig. 12.36. In this window, various controls can be inserted using the **Report Controls toolbar**.

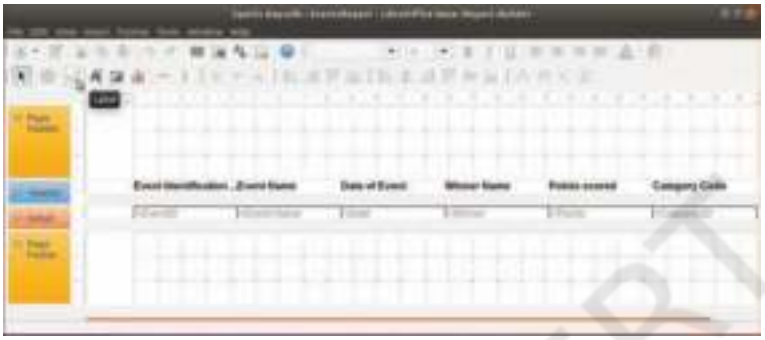


Fig. 12.36: Report Builder window

Inserting Titles and Headings

To insert the title of the report, follow the following steps.

Step 1. Click on the **Label** tool available on the **Report Controls** toolbar.

Step 2. Bring the mouse pointer on the report. Click and drag to insert the label textbox.

Step 3. Double click on it to open the **Properties** dialog box. Type the title text that you want to display in the **Label** property textbox, say ***"Annual Sports Day Report"*** and also set the font style and size using the Font property as shown in Fig. 12.37.

Fig. 12.37: Assigning label name and setting font in Properties dialog box

Step 4. Close the Properties dialog box. Double click on the EventsReport in the Reports of Database Pane. The report will be displayed with with the formatted effects as shown in Fig. 12.38.

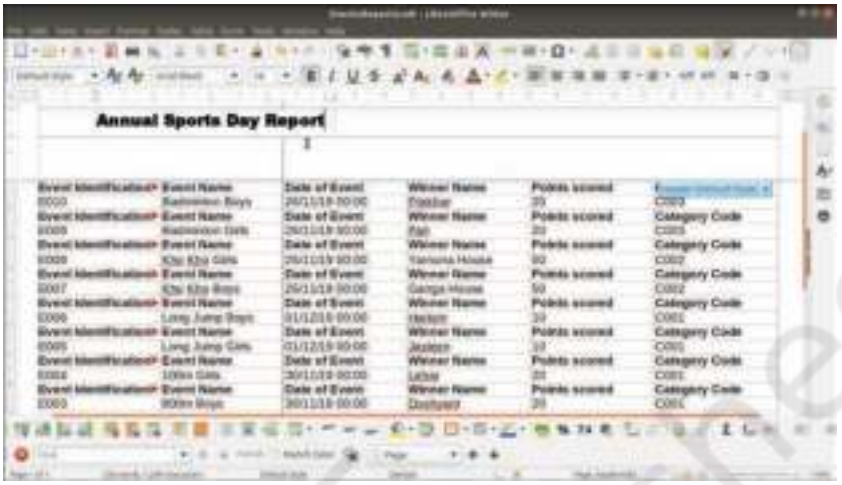


Fig. 12.38: Report with formatted effects

Inserting Date and Time

To insert the date on which the report is generated, follow the following steps.

Step 1. Click in the **Page Header** area to make it active.

Step 2. Click **Insert > Date and Time...** option as shown in Fig. 12.39.

Fig. 12.39: Inserting Date and Time in Header

Step 3. The **Date and Time** dialog box will be displayed as shown in Fig. 12.40. Select the desired format for date and time format and click on **OK** button.

Fig. 12.40: Date and Time dialog boxu

The date will be inserted on the top left corner of the **Page Header** area. You may click and

drag it to reposition it in any place in the **Page Header** area. The final report looks as shown in Fig. 12.41.

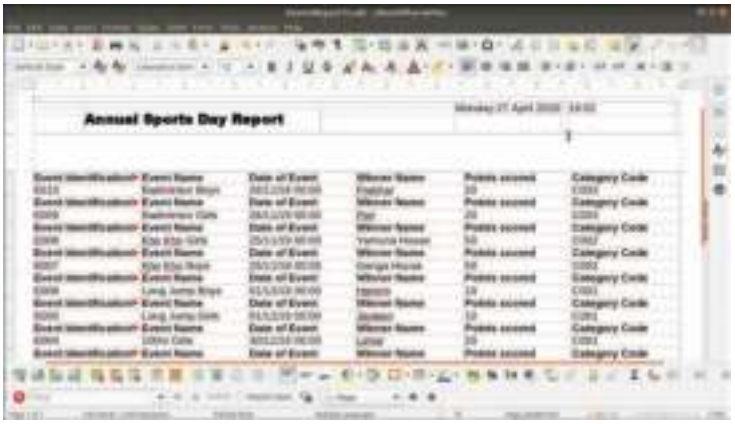


Fig. 12.41: Report with Title and Date

To create a report that groups the retrieved data according to a particular field, specify the field name according to which the retrieved data has to be grouped in step 3 of the wizard. For example, if we want to display the details of events category wise, shift *CategoryID* field to **Groupings** list box as shown in Fig. 12.42.

Fig. 12.42: Moving CategoryID for Grouping records

The category wise report will be generated as shown shown in Fig. 12.43.

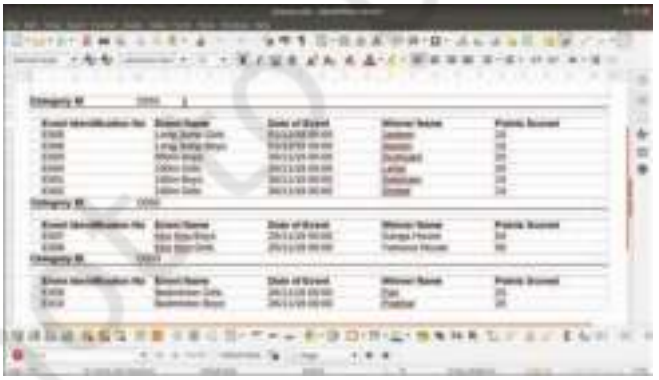


Fig. 12.43: Report generated Category wise

SUMMARY

- A form is an object of the database that has a user friendly interface where data can be entered and seen in an attractive and easy-to-read format.
- A form contains **field controls** arranged in a presentable and user friendly manner.
- Each field control consists of a label and the field value text box.
- There are two ways to create a form – Using a wizard, Using the Design View
- The forms control toolbar contains various tools to add or edit controls on the form.
- A report is used to present the retrieved data in an attractive and customized manner.
- We can create a report based on a table or a query or both.

Practical Exercises

1. Prakasan lives in Happy Home Society. He has created the following tables in his database so as to store and manage the maintenance dues received from flat owners of the society.

Table: Residents Details

Receipt_Id
Flat No
Date
Amount

Table : Maintenance Dues

Flat no
Owner Name
Contact No
Flat Category

In the previous chapter, Prakasan had created a query to display the following information

Flat No	Owner	Receipt Number	Date	Amount

(Please note the change in column headings)

Create a form for data entry of Resident Details and Maintenance Dues table. Also create a report based on the query created above.

2. Uzair manages a boys hostel in an engineering college. To manage the records of the people saying in his hostel, he created a database with the following related tables.

Table : Student_Details

Student_Id
Name
DOB
Course
Father's Name
Contact No_student
Contact No_ guardian
Room No

Table: Room Details

Room No
Floor (data value can be first, second or third)
Category (data value can be AC / Non- AC)
Student Id
DOO (Date of occupancy)
Monthly Rent

Create a query to display the following tabular format

Room No	Student Name	Category	Monthly Rent

Create a query that displays total monthly rent collected for each category.
Create a form for data entry of both the tables.
Generate a report the displays information retrieved from both the queries in a presentable manner.

Check Your Progress

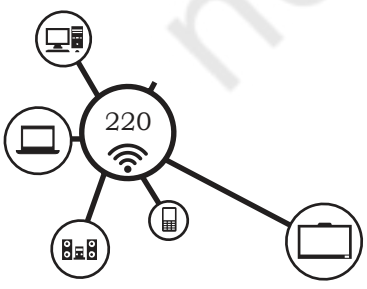
A. Multiple choice questions

1. Which of the following toolbars contains the Label tool?
   (a) Standard Toolbar
   (b) Forms Controls Toolbar
   (c) Records toolbar
   (d) Formatting toolbar
2. The Record toolbar has the buttons to move to the
   (a) first record
   (b) second record
   (c) last record
   (d) all records
3. Which of the following is NOT true about forms?
   (a) It is the front end for data entry
   (b) It can contain only text fields
   (c) Graphics can be inserted on the form
   (d) It can contain only fixed number of records

4. Which of the following keys is pressed to select only textbox on the form?
   (a) Alt
   (b) Shift
   (c) Ctrl
   (d) Tab
5. Which of the following properties in the Properties: Label Field text box is used to insert a tool-tip on the form?
   (a) Tool Text
   (b) Help Text
   (c) Tool Tip
   (d) Help Tip
6. Which of the following objects of LibreOffice Base is used to display data retrieved from one or more tables in a presentable manner?
   (a) Query
   (b) Form
   (c) Report
   (d) Panel
7. Which of the following values of Date Format property is selected to view a calendar on the form?
   (a) Standard (short)
   (b) Standard (long)
   (c) Default
   (d) Standard (Medium)
8. Which of the following commands on the Forms Control toolbar is used to toggle between Design View and Form view?
   (a) Design Mode
   (b) Toggle Mode
   (c) View Mode
   (d) Print mode
9. Using which of the following objects in a database, can a report be generated?
   (a) Tables
   (b) Queries
   (c) Both a and b
   (d) Neither a nor b
10. Which of the following components open along with the Report Wizard?
    (a) Report Builder
    (b) Add Fields dialog box
    (c) Both (a) and (b)
    (d) Neither (a) nor (b)

B. State whether the following statements are True or False

1. Report is an object of a database but form is not.
2. We can choose the layout of the form.



3. We have to add all fields of the table on the form.
4. There are two ways n which a form can be created.
5. A report is generated in a separate window.
6. Once a control is added on to the form, it cannot be re-positioned.
7. The Record toolbar has the button to add a new record.
8. We can create a report only using a table.
9. By default, the records in a report are sorted in descending order.
10. We can group data based on a particular field in a report.
11. A report can have data only in row and column format.
12. We can insert both date and time of generation of report.
13. A report once created cannot be edited.

C. Fill in the blanks

1. A form can be used for _____ and _____.
2. Each field control consists of a _____and _____
3. A _____ is a piece of text that specifies the data that should be entered in the field value text box.
4. By default the border of the field text value is displayed in _____.
5. A _____ is a small piece of text that is displayed when the mouse pointer is placed on a particular control on the form.
6. The default orientation option for a report is _____.
7. A _____ is the manner in which the labels, field values, titles etc. will be displayed in the report.
8. The option to insert date and time in the report is present in _____ menu.
9. A Report Wizard contains _____ steps.
10. A _____ type of report changes automatically as the field values in the base table or query change.

D. Answer the following questions

1. Give one difference between a form and a report.
2. What is a field control with respect to forms?
3. Which tool on the Forms Record toolbar is used to insert text on the form?
4. Name the two ways to create a form in LibreOffice Base.
5. What is the difference between a static and a dynamic report?
6. Write the function of Forms Controls toolbar and Records toolbar.