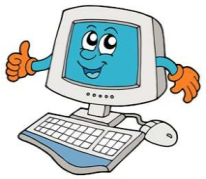
9 Introduction to Database Management System



In current scenario with the advent and reach of digital technologies, almost all data is stored and managed electronically. Day to day transactions in small and medium enterprises are recorded in systems termed as information systems. Users of information system are growing multifold. Users expect flexibility in accessing data irrespective of the places they are at and irrespective of where the data is stored. Organizations now being global, information systems are not now limited to storage and maintenance of enterprise transactions which were understood as replacement of registers/files maintained in organization. Search engines, reservation systems, online banking, online tax filing, stock trading are examples of systems used by millions of users simultaneously. Today usage of information systems is becoming inevitable. All of us directly or indirectly act as users of one or other information systems. Doctors, engineers, shopping store owners, broking agents, educators, and students, none of the entities have remained untouched. Information systems today has also become a strong helping hand in taking decisions. To take best possible decision, everyone needs correct, precise and relevant information on time. It is obvious that the perfect decision depends on the accuracy of information available. Information is the output generated through processing of raw data. So, let us first understand the two important terms i.e., Data and Information.

Data and Information

Many a times the word Data and Information are used interchangeably. So it is necessary to have a clear idea of both the words. Let us first understand Data and Information by their possible definitions.

"All kinds of facts, figures and details related to people, places, things or events is known as data." Data may be in any form. It may be written, oral, computerized or non-computerized. Data in its original form may not be of much use. Data must be processed in a proper way to generate the useful and meaningful information. Looking to the importance of data for decision-making process, many business houses consider it as important asset of the business.

"Information on the other hand can be described as the required result obtained from processing of the data." This means that, the Information is nothing but "The Processed Data". Information is important because it forms the foundation for Decision Making. Taking timely decisions, based on correct and relevant information may be beneficial for us. On the other side, taking decision based on incomplete or incorrect information may badly affect the business.

Let us clear the concept of data and information with an example. Suppose you and your team in a school decide to send birthday wish to each student of the school. This idea requires collection of birth dates like 10th September 1998. The dates collected act as data in this case. If today is 10th September, you will find out names of the boys and girls whose birthday falls on this date. The set of names in your list now acts as information.

Database and Data Model

Having learnt about data and information, let us now have a look at database. We all use one or more type of databases in our day-to-day life. We keep a diary containing address and phone

numbers of our relatives and friends at home. Same way your school maintains details of all the students enrolled in the school. An attendance record of every student has to be maintained by school. Indian government is issuing unique identity numbers to each citizen under Adhar project. Thus a detail of each person has to be maintained. Passport office maintains details of the issued passports. Doctors maintain medical history of each patient. Railway department maintains details of passengers. All these are examples of records.

As per the common definition of the database *"Database is a collection of related data items stored in an organized manner"*. As discussed earlier, data is stored because data is to be used. Thus the word *related* and *organized* is very important in definition of database. You may compare storing data to arranging things in your room. You might be receiving instructions from your parents and teachers to keep your belongings at a proper and pre-decided place. You may fix a place of school bag, stationary, books etc. You may label the compartments in the book shelf. For example, science, general knowledge etc. If your father gifts you a book on human body, you may decide to keep it in science section. Next time when you want to read the book, it would be very easy for you to find out the same book. Logical arrangement of things always makes searching easier as and when required.

Similar to your books, clothes in wardrobe also require logical arrangement. However, you might have noticed that structure of wardrobe and book shelf is normally different. The place designated to store jewellery or kitchen utensils has different structure. Above examples indicate, that depending on the things to be stored, structure of shelves is to be designed.

Similarly, databases are designed according to certain rules. The underlying structure of database is known as data model. Data model describes way of storing and retrieving the data. There are different data models like hierarchical data model, network data model and relational data model. In this text, we will be discussing relational model. Different Database Management Systems are available in the market, which are based on one of the above mentioned data models. Let us now understand what Database Management Systems are.

Database Management Systems

To work with computerized database, one must first define and create a database. However, just creating database is not the only task. In order to keep the database ready with latest up-to-date data, following basic tasks are expected to be performed regularly with as much ease as possible.

- Adding data into the database as and when transaction takes place.
- Editing the content of the database with the corrections required as and when noticed.
- Removing unnecessary data from the database, if any.
- Organizing the database in a proper manner.
- Retrieval of data as per user requirement.
- Securing data stored in database.
- Maintaining backup of database.
- Allowing multiple users to access database simultaneously.
- Allowing data to be stored in multiple languages.
- Allowing data to be stored at multiple places.

All tasks related to database handling is known as database management. Naturally, we need special software to do this kind of work. Such software are known as Database Management Systems

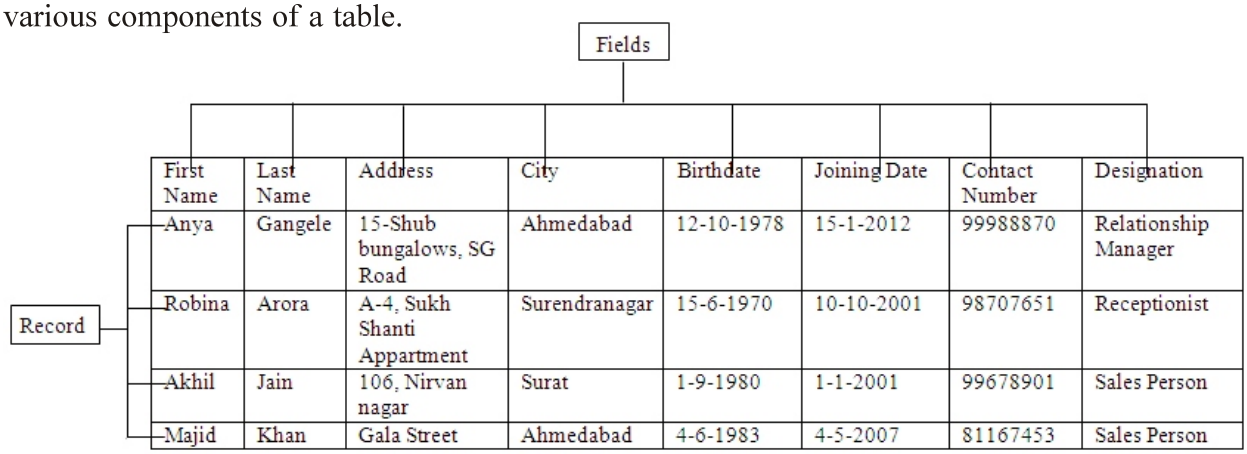
(DBMS). DBMS can be defined as system designed for efficient storage, maintenance and retrieval of data. In short the DBMS software is what makes database practical and more usable. In today's world a database itself is useless if there is no DBMS associated with it for accessing its data. Oracle, Ingress, DB2, SQL server, MYSQL, MS Access, Base etc. are the names of some popular DBMS software available in the market. In this book, we will discuss about a Base Database called as OpenOffice Base. From now onwards we call it as Base. As we are aware of terms data, information, database and database management system, let us proceed further and get acquainted with Base, an open source database available as part of Open Office suite. Base is based on a Relational data model.

Introduction to Base

Base is Relational Database Management System (RDBMS) as it follows relational model. As discussed earlier, the manner in which information is organized can have a profound effect on how easy and efficient it is to access and manage. Perhaps the simplest but most versatile way to organize information is to store it in tables. The relational model is centered on this idea: the organization of data into collections of two-dimensional tables called "relations." Designed by E.F. Codd, relational model is based on the theory of sets and relations of mathematics. A DBMS that is based on relational model is called as RDBMS. Relational model is the most successful data model. One of the reasons behind the success of relational model is its simplicity. It is easy to understand the data relationship and easy to manipulate.

Base is a collection of related data objects known as Tables, Queries, Reports and Application Modules. Let us now understand objects that can be created using Base in brief. Each of these objects are discussed in detail in later chapters.

Table : Being RDBMS, Table is the basic unit for storing data in database designed using Base. Tables are organized in the form of columns and rows. To decide what information would be stored in a table and what would be the column titles, database designer needs to first decide entities in database. Entities are real world objects about which information is to be stored in database. For example, while creating a database design for Employee Attendance System, the first entity which comes to our mind is Employee. Details of each employee need to be stored in database. These details are stored in the form of rows. Each entity generally has collection of attributes associated to it. For example, each employee would have attributes like first name, last name, address, date of birth, contact number etc. Attributes of an entity are represented in the form of columns. Figure 9.1 shows how data is represented in relational model and what are the terms used to refer to various components of a table.



First Name	Last Name	Address	City	Birthdate	Joining Date	Contact Number	Designation
Anya	Gangele	15-Shub bungalows, SG Road	Ahmedabad	12-10-1978	15-1-2012	99988870	Relationship Manager
Robina	Arora	A-4, Sukh Shanti Appartment	Surendranagar	15-6-1970	10-10-2001	98707651	Receptionist
Akhil	Jain	106, Nirvan nagar	Surat	1-9-1980	1-1-2001	99678901	Sales Person
Majid	Khan	Gala Street	Ahmedabad	4-6-1983	4-5-2007	81167453	Sales Person

Figure 9.1 : Employee table

As can be seen in figure 9.1, attributes of an entity are also termed as fields. Thus in employee table, First Name, Last Name, Address, City, Birthdate, Joining Date, Contact Number and Designation are fields. Collection of all these fields forms a record. Thus << 'Anya', 'Gangele', '15-Shub bungalows, SG Road', 'Ahmedabad', '12-10-1978', '99988870', '15-1-2012', 'Relationship Manager'>> is a record.

Forms : We need to enter new records into the table, edit existing records, and view the records available with the table. Form is an object which allows entering the data in the table and editing or deleting existing data in the table. It consists of format, style and widgets like radio button, list boxes that provides easy and systematic way to populate table with data.

Queries : A question asked within the database environment is known as Query. For example, how many Android based mobiles were sold in July in current year? Query displays subset of data contained in various tables of a database.

Reports : The output of Query in Base is displayed in the form of rows and columns. As we discussed earlier, the available data is to be processed for required information and an output comprised of information is to be generated. Obviously, user expects more formal and attractive output. Base has a facility to prepare effective professional report. The presentation of information in an organized and readable format as per the user requirement is known as Reports. Various complex reports could be generated that can help in taking decisions by management in organizations.

Creating a Sample Database for an Application

During the entire journey of learning Base, we will move towards creation of a small application for a hypothetical Modern Electronic Store. This store sells devices like Mobiles, Camera, Handy Cams, Laptops, Netbooks, ipads, Smartphones and accessories related to them. Products are procured from various suppliers. Store has physical shop as well as web shop. Thus customers can either purchase items walking in the shop or place orders online. Sales Persons are paid special incentives for converting leads to customers. Let us name the application we are aiming for as Store Management System. Let us design a sample database for Store Management System. In later chapters, we will design Forms and Reports required for this system. Following steps help us to design the required database.

Step 1 :

Decide about whom data is to be stored? In our example, we will store data pertaining to products, suppliers, employees and customers. All of these are known as *Entities*. As discussed earlier, entities can be defined as any person, thing, place or an object about which we store the data in the database. For each entity a separate table should be designed. Thus we will design four tables named Product, Supplier, Employee and Customer.

Step 2 :

Decide what attributes of each entity is to be stored. For example, what attributes of product, supplier, employee and customer should be stored? These attributes will become fields of a Table. *Attributes (Fields) can be defined as characteristics of an entity*. We will store details of suppliers, employees, customers and products as shown in Table 9.1, Table 9.2, Table 9.3 and Table 9.4.

Supplier	
Attributes	**Description**
Supplier Name	Name of supplier, it could be name of company like Samsung India
Address Line 1	Address from where products are dispatched
Address Line 2	Street and/or area
City	City from where products are dispatched
Pincode	Pincode of area within a city
Contact Person Name	Person to be contacted
Email Id	E-mail address of contact person
Contact Number	Contact number of office

Table 9.1 : Attributes for Supplier entity

Employee	
Attributes	**Description**
Firstname	Name of the Employee
Lastname	Surname of the Employee
Address Line 1	Address of the Employee
Address Line 2	Street and/or area
City	City
Pincode	Pincode of area of above city
Email Id	E-mail address of Employee
Mobile Number	Mobile number of Employee
Birthdate	Birthdate of Employee
Joiningdate	Joining date of Employee
Designation	Current designation of an employee

Table 9.2 : Attributes for Employee entity

Customer	
Attributes	**Description**
Firstname	Name of the Employee
Customer First Name	Name of customer
Customer Last Name	Surname of customer
Address Line 1	Address of customer
Address Line 2	Street or area
City	City
Pincode	Pincode of area of above city
Email Id	E-mail address of customer
Mobile number	Mobile number of customer
CardHolder	Whether customer is member of Modern Electronic Store

Table 9.3 : Attributes for Customer entity

Product	
Attributes	**Description**
Product Category	Category of Product like ipad, mobile, smartphone etc.
Model Name	Name of the Model like Samsung Galaxy III
Supplier Name	Name of the Supplier
Reorder Level	Stock level that triggers purchase order
Selling Price	Current selling price of the product
Quantity on Hand	Stock available
Product Image	Image of Product
Camera Pixel Size	Mega pixels of Camera
Processor	Processor name
OS Support	Name of Operating Systems supported by model
Touch Screen	Touch Screen Feature available or not
Wi-fi Enabled	Whether the phone is wi-fi enabled
Memory Slot	Capacity of Initial slot of memory
FM Support	Whether FM support is available or not
Display Size	Display size in inches
HDD Capacity	Hard disk capacity
Weight	Weight of the item

Table 9.4 : Attributes for Product entity

Logical Names and Physical Names of Attributes

When thinking about the name for attributes it is useful to differentiate between a logical name of the attribute and the physical name used in the database application. For example, "First Name" is the name of the fields that stores values of the first name like "Sonal" or "Minal". "First Name" describes a data entity we are working with. It does not matter if "First Name" is alternatively written "FIRST NAME" or "firstName" or "first_name" or even "FName" as long as we conceptually understand that it refers to the first name.

When we think about conceptual or logical relationships, we need clear and descriptive name. Logical names are used at the time of designing the data model. Unfortunately Logical names do not translate so simply into the names that database software will allow us to use. For example, Base allows us to name a field as "First Name" but other database applications would reject it and ask us to use "first_name" instead. This restriction is due to the way particular database software has been designed. The name actually used in the internal structure of our tables is called a "physical" name, as opposed to the logical name discussed above. This physical name will always have to confirm to the conventions imposed by the software that we are using.

Following the conventions mentioned below to name the attributes will facilitate porting of database design to other database software (Base in our case).

1. Start all attribute names with a letter.
2. For subsequent characters use either letters, numbers or the underscore character.
3. Do not use a space between words; instead separate them with the underscore.
4. Do not use special characters except underscore.
5. Use abbreviations, if needed, to keep the length of attribute names short.

We will use camel back notations in database design for our sample application. This means each word in field name starts with capital letter. For example, FirstName, LastName, AddressLine1 etc. Figure 9.6 and 9.7 shown later in the chapter shows field names that would be used for our sample database for Modern Electronic Store.

Step 3 :

Decide what kind of data we want to enter into each attribute. Depending on it we need to decide data type of each field while designing the table. It is compulsory to define the data type of each field. *Data type is a way to define storage structure of the field.* Computers have different ways of storing data. In general, they trade memory or speed for accuracy. Computations that require more accuracy tend to be slower and use more memory. When building tables with Base, we would be presented with a drop-down menu of options asking us to choose the kind of attributes we want to store. The choice we make here will affect the performance of the database.

Figure 9.2, 9.3, 9.4 and 9.5 shows sample records we would like to insert in the tables discussed. Later we will discuss data types provided by Base. Then we will decide which data type we should designate for the attributes of our sample tables.

Supplier Name	Address Line 1	Address Line 2	City	Pincode	Contact Person Name	Email Id	Contact Number
Samsung Corporation	27, Maker Chambers	Nariman point	Mumbai	400021	Mr.Mahesh Wadkar	purchase@samsung.com	66006600
Hewlett-Packard India Sales Pvt. Ltd	24, Salarpuria Areana	Adugodi, Hosur Road	Bangalore	560030	Mr.Venkatesh Rao	placeorder@hp.com	26543123
Sony India	A-31, Mohan Co-operative Indisurial Estate	Mahura Road	Delhi	110001	Mr.Habib Ansari	purchase@sony.com	33344441
Nokia India	Salarpuria Hallmark	Outer Ring Road	Bangalore	560089	Ms.Rita Kulkarni	po@nokia.com	89674210

Figure 9.2 : Sample records of Supplier

Product Category	Model Name	Supplier Name	Reorder Level	Selling Price	Quantity on Hand	Product Image	CameraPixel Size	Processor	OS Support	Touch Screen	Wi-fi Enabled	Memory Slot (in GB)	FM Support	Display Size (in inches)	HDD Capacity (in GB)	Weight (in gms)
Smartphone	Samsung Galaxy SIII	Samsung Corporation	20	42500	43	./images/sgIII.jpg	8	1.4Ghz Quad Core Processor	Android 4.0	yes	yes	16	yes	4.8		133
Laptop	hp ENVY Sleekbook 6t-1000	Hewlett-Packard India Sales Pvt. Ltd	15	35000	56	./images/hp6t.jpg		Intel Core i5	Windows		yes	16		15.6	500	4.75
Laptop	hp ENVY Sleekbook 6t-1000-L	Hewlett-Packard India Sales Pvt. Ltd	15	30000	56	./images/hp6t.jpg		Intel Core i5	Linux		yes	16		15.6	500	4.75
Mobile	Nokia Lumia 710	Nokia India	30	15499	32	./images/Lumina710.jpg	5	Qualcomm MSM8255 (WCDMA)	Windows Phone 7.5	yes	yes	8	yes	3.7		125.5
Camera	DSC-S3000/S	Sony India	35	4490	38	./images/sdscs3000.tiff	10.1							6.14		11.9

Figure 9.3 : Sample records of Product

First Name	Last Name	Address Line 1	Address Line 2	City	Pincode	Email Id	Mobile Number	Birthdate	Joiningdate	Designation
Anya	Gangele	15-Shub Bungalows	SG Road	Ahmedabad	380067	anya.gangele@gmail.com	99988870	12/12/1981	1/6/2011	Relationship Manager
Robina	Arora	A-4, Sukh Shanti Appartment	Thaltej	Surendranagar	363002	robina.arora@yahoo.com	98707651	6/7/1984	2/7/2006	Receptionist
Akhil	Jain	106, Nirvan Nagar	Army Road	Surat	395001	akhil.jain@yahoo.in	99678901	8/10/1983	1/1/2000	Sales Person
Majid	Khan	B-12, Nishant Appts.	Satellite	Ahmedabad	380058	majid@rediffmail.com	81167453	7/11/1985	1/5/2001	Sales Person

Figure 9.4 : Sample records of Employee

Customer Fname	Customer Lname	Address Line1	Address Line 2	City	Pincode	Email Id	Mobile Number	Card Holder
Puja	Sethia	17,Akash	Jodhpur Char Rasta	Ahmedabad	380056	puja.sethia@yahoo.com	9998886662	Yes
Sakina	Nagingar	T-37, Abhimanyu Nagar	Curry Road	Mehsana	384001	sakina82@gmail.com	8111203450	No
Joseph	Thomas	BG-1, Sunstar Complex	Sola Road	Patan	362268	joseph@yahoo.com	9980054321	Yes
Prerna	Agrawal	Smita Bunglows	PNT Road	Ahmedabad	380089	prerna@gmail.com	9004532189	No
Harshit	Shah	17, Jaldeep-1	Bopal	Ahmedabad	380058	harshit@yahoo.com	9012354310	Yes

Figure 9.5 : Sample records of Customer

Data types available in Base

Data types available in Base can be divided into three categories, namely, Alphanumeric type, Calendar type and Binary type.

Alphanumeric Type

When we wish to enter a data that consists of letter, number as well as special character then we can select one of the data types shown in Table 9.5. For example Addresses may contain

alphabets, special characters as well as numbers, thus the appropriate data type assigned to addresses would be one of the alphanumeric type. We can select Text (Varchar) as data type for Address field. The number of bytes to be allocated to Var Char data type depends on the number of characters entered by the user. For example, if address of Harshit contains 100 characters, 100 bytes would be reserved and if it contains 50 characters only 50 bytes would be reserved.

Name	Data type	Max Length	Description
Memo	Long Var Char	2 GB	Stores up to the max length or number indicated by user. It accepts any UTF 8 Character
Text (fix)	Char	2GB	Stores exactly the length specified by user. Pads with trailing spaces for shorter strings. Accepts any UTF 8 Character.
Text	Var Char	2GB	Stores up to the specified length. No padding (Same as long var char)
Text	Var Char Ignore Case	2GB	Stores up the specified length. Comparisons are not case sensitive. It stores capitals as you type them.

Table 9.5 : Alphanumeric Type Attributes : Used for storing alphanumeric characters

Character data type has different nature. It is used when we are aware of exact number of characters to be entered in the field. For example we can use it in case of license number, passport number as they have fixed number of characters. The character data type though has a limitation on number of characters that can be used.

When we want to store some descriptive data, which may have more than 255 characters, then the Memo data type will be useful. This data type allows us to store any text data up to 64,000 characters. Text (Var Char) Ignore case is efficient when the fields are used for searching and it is unsure that whether user will enter searching text in lower case or upper case. For example, First Name or Last name can be stored in Text (Var Char) Ignore case.

Note that computers have different ways of storing alphanumeric characters. For example there is the ASCII code that needs only one byte to store a character. Unfortunately, this limits the number of possible characters that you can use to only 256. Although enough for many applications, it falls short if you want to have access to expanded character sets like Gujarati or Hindi characters. Standards that allow for larger numbers of characters have been developed, like Unicode that use more bytes per character.

Base will store alphanumeric characters using UTF-8, which is a code that is compatible with both ASCII and Unicode. Base will use one or more bytes for each character according to internal calculations. When Base asks you the length for a particular field, e.g. for Surname of Employee, it is not asking the number of Bytes you want to allocate but the number of characters you want to store. How many bytes are actually used is fixed by the software.

Numeric Type

This data type is used to store numerical information like marks of a student, salary of an employee, quantity on hand or balance in the bank account of a person.

We have already seen that in alphanumeric type, the number of bytes actually used is fixed by the software. This is not the case when you store the value for a number. Different ways of storing numbers will require more or less bytes. Numeric data type attributes are described by the number of bytes they use and whether they are signed or unsigned. These two factors determine the range of possible values they can hold. Base offers several types of numeric data attributes, both signed and unsigned that use different amount of bytes.

At the least memory consuming side of number storage, we have the Boolean numbers. A Boolean number is in fact uses just one bit, and we use it to store YES/NO type of data, like answer to the question 'whether the mobile is Wi-Fi enabled?', At the other end there are attributes called 'floating point numbers' that allow us to store numbers that have decimal places like 1.715249. They are the most memory consuming numbers but the only ones that can perform divisions with good accuracy. Tables 9.6 list different numeric data types along with the number of bits/bytes it uses and its range.

Name	Data type	No. of Bits/Bytes	Signed	Range
Boolean	Yes/No	1 Bit	–	0 – 1
Tinyint	Tiny Integer	1 Byte	No	0 – 255
Smallint	Small Integer	2 Bytes	Yes	-32768 to 32768
Integer	Integer	4 Bytes	Yes	-2.14×10^{9} to 2.14×10^{9}
Bigint	Big integer	8 Bytes	Yes	-2.3×10^{18} to 2.3×10^{18}
Numeric	Number	No limit	Yes	Unlimited
Decimal	Decimal	No limit	Yes	Unlimited
Real	Real	4 Bytes	Yes	$5 \times 10^{(-324)}$ to $1.79 \times 10^{(308)}$

Table 9.6 : Numeric Type Attributes : Used for storing numeric values

Calendar Type

Another important data type is the Calendar. They are used to store calendar information like year, month, day, hour, minute, second and fraction of a second. Date allows storing year, month and day as it is stored in the system through which data is entered. The same is true for the Time type attribute, which stores the time of the day: hour, minute and second. Finally, some procedures might demand recording of both the time and day of an event. Timestamp has been designed for recording all information at once. In our example database, the fields Joining date, Birth date etc. must be assigned Date data type. Table 9.7 list the name and description of the Calendar data type.

Name	Description
Date	Stores month, day and year information
Time	Stores hour, minute and second information
Timestamp	Stores date and time information

Binary Type

The Binary type attributes allow us to store any information that comes as a long string of zeros and ones. Digitized images and Sounds use this format. They are distinguished by the computer because the initial zeros and ones identify the kind of file they represent (a JPEG image or an MP3 file, etc.). However, Base will make no attempt to identify the kind of file you have stored. This is to say that it won't care if the file is an MP3 or a TIFF and it will happily store it. This in effect means that you could use a Base database to store, for example, photos of the products or employees, or sound snippets or voice messages. Table 9.8 lists the details of the binary data types.

Name	Data type	Max Length	Description
Image	Long Var Binary	2GB	Stores any array of bytes (images, sounds, etc).
Binary	Var Binary	2GB	Stores any array of bytes.
Binary (fix)	Binary	2GB	Stores any array of bytes.

Table 9.8 : Binary Type attributes: Used for storing images and sounds

Now let us assign data types as per our requirement of input data. Figure 9.6 and figure 9.7 shows data types assigned to each field of Supplier, Product, Employee and Customer table.

Supplier	
Field Name	**Data type**
SupplierName	Text [VARCHAR_IGNORECASE]
AddressLine1	Text [VARCHAR_IGNORECASE]
AddressLine2	Text [VARCHAR_IGNORECASE]
City	Text [VARCHAR_IGNORECASE]
Pincode	Text fix [CHAR]
ContactPersonName	Text [VARCHAR_IGNORECASE]
EmailId	Text [VARCHAR]
ContactNumber	Integer

Employee	
Field Name	**Data type**
FirstName	Text [VARCHAR_IGNORECASE]
LastName	Text [VARCHAR_IGNORECASE]
AddressLine1	Text [VARCHAR_IGNORECASE]
AddressLine2	Text [VARCHAR_IGNORECASE]
City	Text [VARCHAR_IGNORECASE]
Pincode	Text fix [CHAR]
EmailId	Text [VARCHAR]
MobileNumber	Text [VARCHAR]
BirthDate	Date
JoiningDate	Date
Designation	Text [VARCHAR_IGNORECASE]

Figure 9.6 : Data types assigned to fields of Supplier and Employee table

Product	
Field Name	**Data type**
ProductCategory	Text [VARCHAR_IGNORECASE]
ModelName	Text [VARCHAR_IGNORECASE]
SupplierName	Text [VARCHAR_IGNORECASE]
ReorderLevel	SmallInt
SellingPrice	Integer
QuantityonHand	SmallInt
ProductImage	Text [VARCHAR]
CameraPixelSize	Integer
Processor	Text [VARCHAR_IGNORECASE]
OSSupport	Text [VARCHAR_IGNORECASE]
TouchScreen	Boolean
Wi-fiEnabled	Boolean
MemorySlotInGB	SmallInt
FMSupport	Boolean
DisplaySizeInInches	SmallInt
HDDCapacity	SmallInt
WeightInGrams	SmallInt

Customer	
Field Name	**Data type**
CustomerFname	Text [VARCHAR_IGNORECASE]
CustomerLname	Text [VARCHAR_IGNORECASE]
AddressLine1	Text [VARCHAR_IGNORECASE]
AddressLine2	Text [VARCHAR_IGNORECASE]
City	Text [VARCHAR_IGNORECASE]
Pincode	Text [VARCHAR]
EmailId	Text [VARCHAR_IGNORECASE]
MobileNumber	Text [VARCHAR]
Card Holder	Boolean

Figure 9.7 : Data types assigned to fields of Product and Customer table

Note : When we know that field will contain only numbers but, we don't expect to perform any calculations or sorting with the values in the field, it is better to use the Text data type. Hence Pincode and Mobile number have been assigned Text data type.

Here you may think that we can use text data type for the field DisplaySizeInInches defined in Product table in our sample database. Base will allow to insert data like 1, 12, 457 etc. But when we perform sorting (discussed in later chapter) on the records according to Display Size, we will get the result in the reverse numeric order: 457, 12, 1. The reason is when Base sorts numbers in a Text field, it reads the numbers from left to right instead of right to left. Thus while sorting Base reads the right most digit (7 in case of 457) and sorts accordingly.

Step 4 :

Decide a key field of the table designed. The field that has unique importance in the table is known as Key Field.

What is the purpose of model name given to each product manufactured by a company? It helps us distinguish a product from other. You may visit Modern Electronic Store and can directly ask him to show you Samsung Galaxy III. You will not find any confusion on shop keeper's face. On the other hand the Employee table may contain more than one employee with same first names. How could they be identified? We may query the database to return details of Anya, but database may return with multiple records for Anya. We may get confused seeing the result thinking which Anya's details we were looking for? Thus each table should contain at least one field which can identify each record of a table uniquely. For example, a unique employee code may be given to each employee. Hence we may need to add this new field as an attribute in the Employee table.

In terms of database, a field or combination of fields capable of identifying each record uniquely is known as Primary key. You may think even driving license number or passport number can serve as Primary key but then there would be multiple records with null value. It is not always necessary that an employee when recruited has driving license or passport. Have a look at the sample records of Product table shown in figure 9.3 again. You must have observed that we have not entered any value in HDD Capacity of Smart phone. There can be two reasons for this, either we do not know HDD Capacity of Smartphone or that particular attribute may or may not contain any value. Such type of values is known as Null value. A null value means unknown or missing value. *Primary Key values cannot contain Duplicate values or Null values.*

Figure 9.6 and 9.7 shows four tables Supplier, Product, Employee and Customer. As can be seen the Supplier table does not have any field that contains a unique value. Thus we shall need to add an extra field **Scode** in the Supplier table so that each supplier record can be uniquely identified. We also need to decide the structure of Scode. Let us assume it to be a field that has four characters. The first character has to be alphabet 'S' and the other three characters would represent a number. Thus suppliers of the Modern Electronic Store could be identified as S001, S002, S003 and so on. Similarly, an attribute **Pcode** should be added in Product table, **Ecode** in Employee table and **Ccode** in Customer table.

You may observe that Product table consists of field Model Name which can also act as Primary key in the Product table. However, you may notice that model name consists of variable characters with approximately more than 15 characters for each Product. If we designate only 8 characters for the product code, we can identify 9999999 products uniquely (first character would be P). Primary key of a table is also used to establish relationships with multiple tables in the database. We will discuss how to establish relation between tables in next chapter.

Defining a primary key is must for each table. A table though cannot have more than one primary key. However combination of more than one field can serve as primary key. A primary key so designed is known as Composite primary key. Before discussing the design of other tables to be used in our sample application, let us first design the table already discussed using Base.

Opening Base

Let us first start Base. Select Applications → Office → OpenOffice.org 3.2 Base. When you start the Base software, you will see some options available on the Base opening screen as shown in figure 9.8.

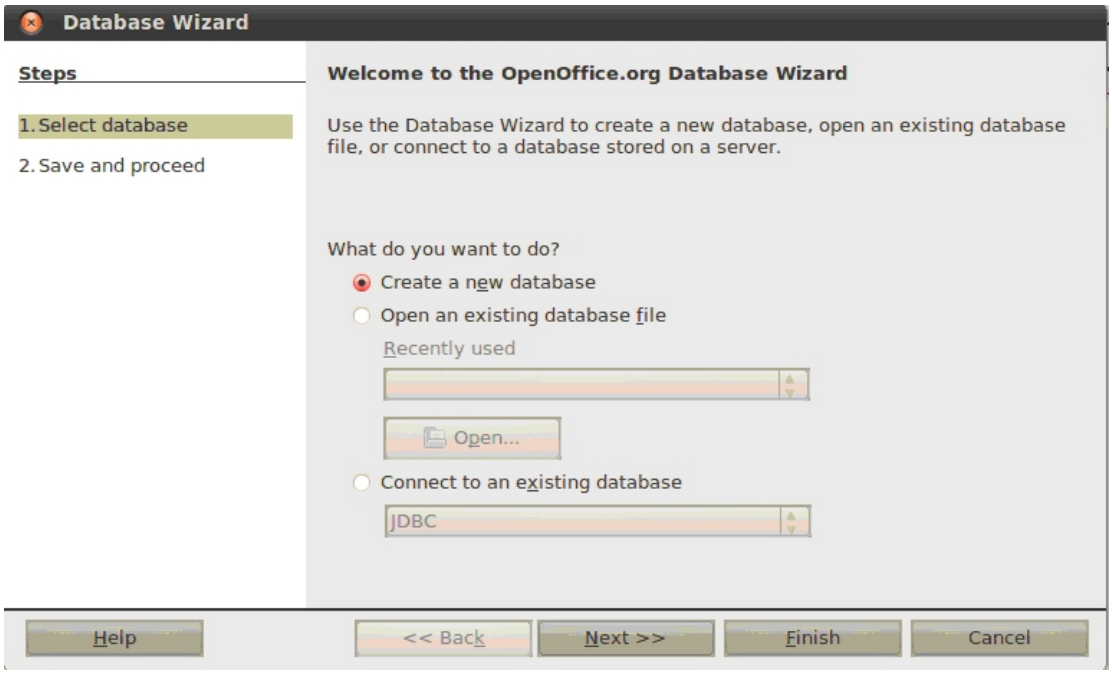


Figure 9.8 : Opening Screen of Base

If we want to work with any of the database created earlier then we can open such databases by choosing an option *Open an existing database file*. Here, in our case, we have not created any database so far. Hence we will choose the option *Create a new database*. By doing so, Base will display a screen as shown in figure 9.9.

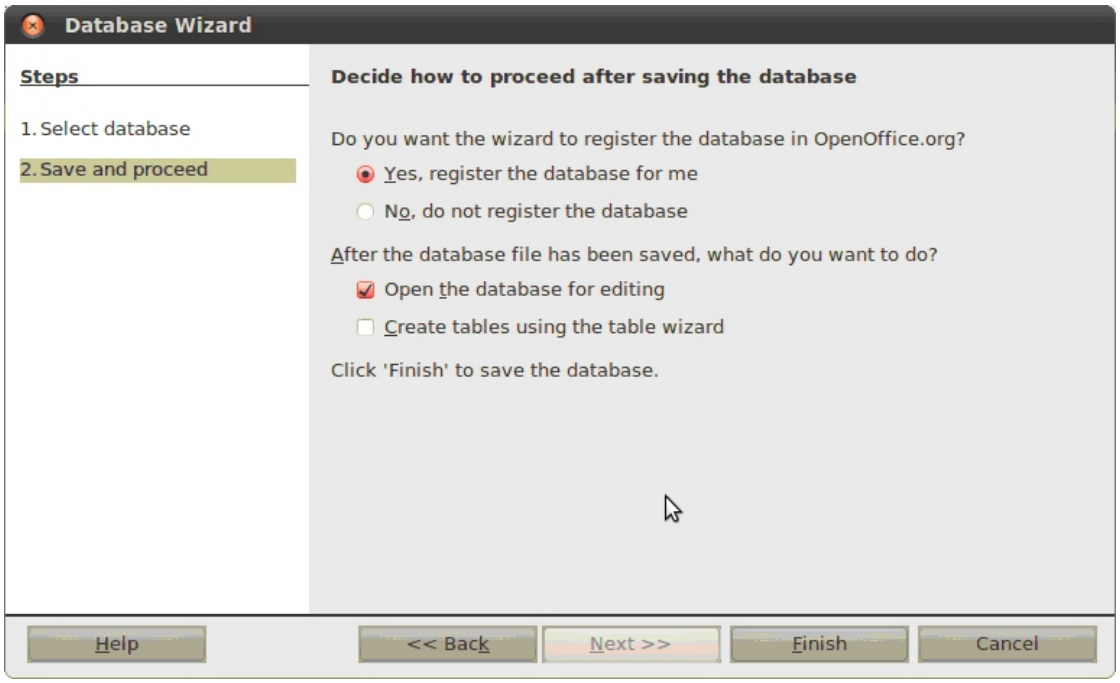


Figure 9.9 : Opening the Database

In figure 9.9, you may observe that we get an option to register our database in OpenOffice.org web site. We may select this option in case we want to make the database public. Normally we would not make the database public, hence select the second option "*No, do not register the database*". Further Base gives us two options to start working with database. We can either create tables using Design view or can use Wizard. By default, *Open the database for editing* option would be checked. If we want to use wizard, Select the check box with option *Create tables using the table wizard*. Let us create the tables using wizard. Select the option *Create tables using the table wizard* and click on Finish button. The Save As dialog box shown in figure 9.10 will now open.

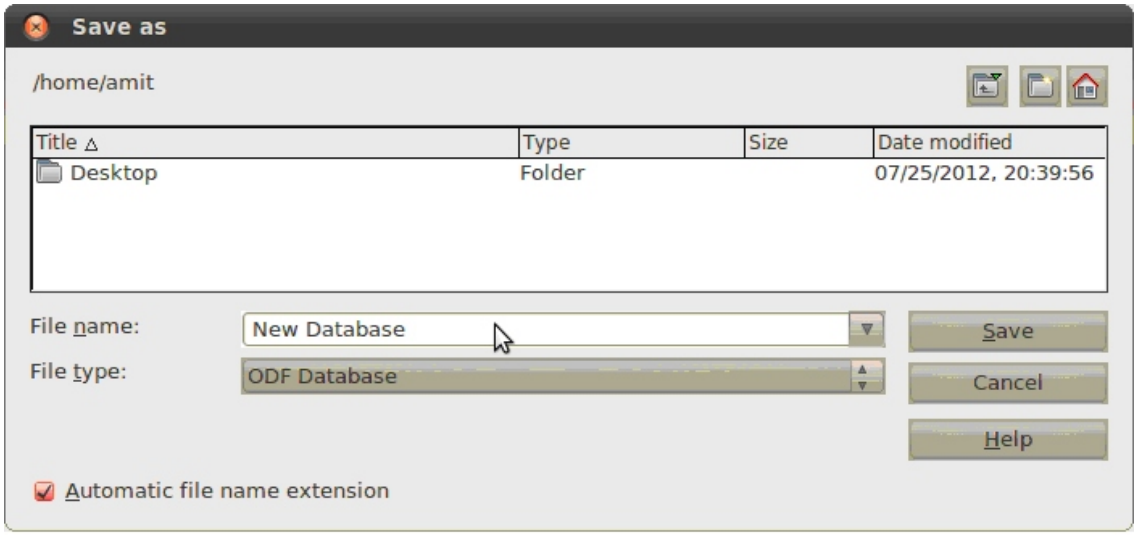


Figure 9.10 : Naming Database File

In textbox with label *File name:* where text *New Database* is visible type ModernElectronicStore and click Save button. Choose appropriate location to save the file by clicking the drop down arrow. Base will automatically assign .odb extension to the database file. Click Save Button and Table Wizard as shown in figure 9.11 will be displayed.

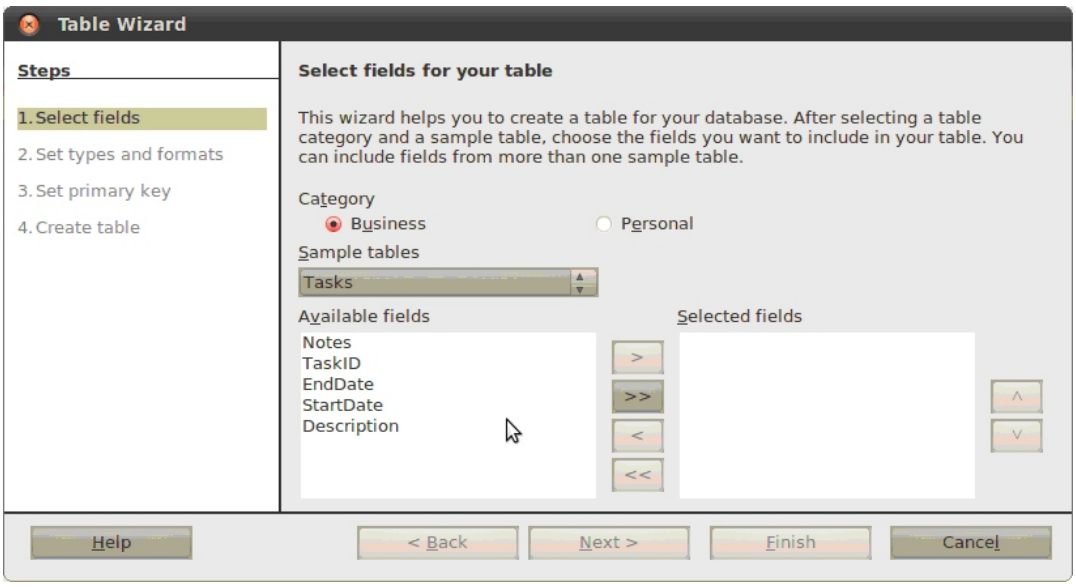


Figure 9.11 : Table creation using Wizard

We will first use Table Wizard of Base software which consists of readymade tables divided into two categories, viz., Business and Personal.

Business Category consists of sample tables like Tasks, Assets, Contacts, Expenses, Invoices etc. Personal Category consists of tables like Photographs, DVDCollection, Recipes, Plants etc. Select Business category and click on drop down list available under *Sample tables* list. Observe the list and you will find that all the four tables we intend to create are available in the list. Select Supplier. List of fields as shown in figure 9.12 will be populated.

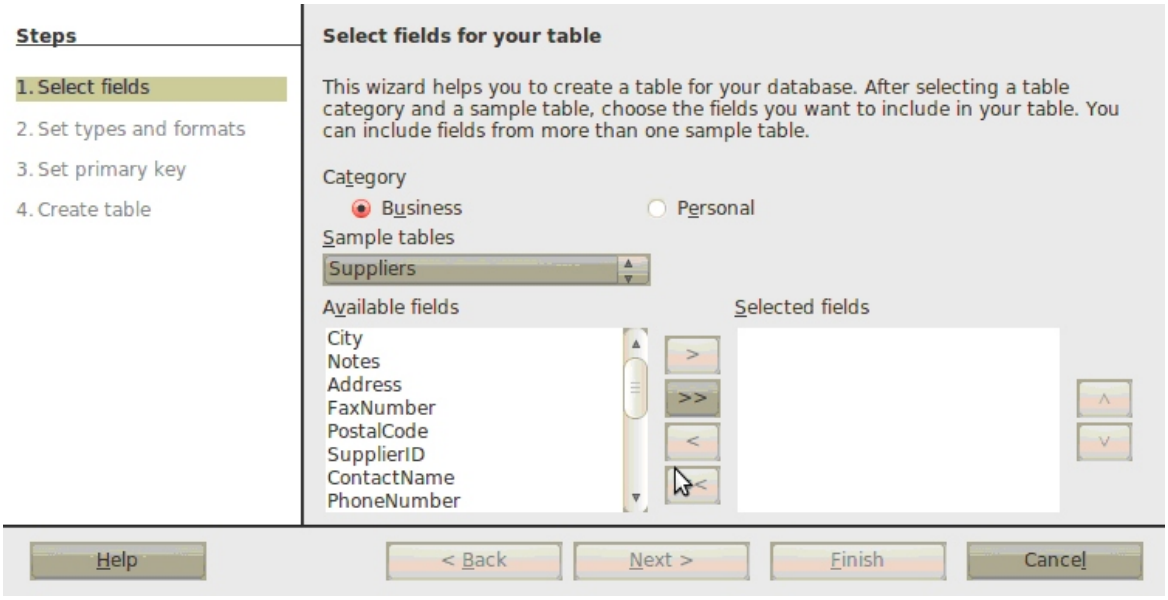


Figure 9.12 : Creating Suppliers table

Click on the >> button to select all the fields that are visible under *Available fields* list. Alternatively you can hold control key and select only the desired fields one by one. Once you are done, click on > button. Time being we will click on >> button so that all the fields are part of our table.

Click on Finish button, the Suppliers table will be opened in Data View. Opening table in Data View allows user to enter records in the table. Figure 9.13 shows data view of Supplier table.

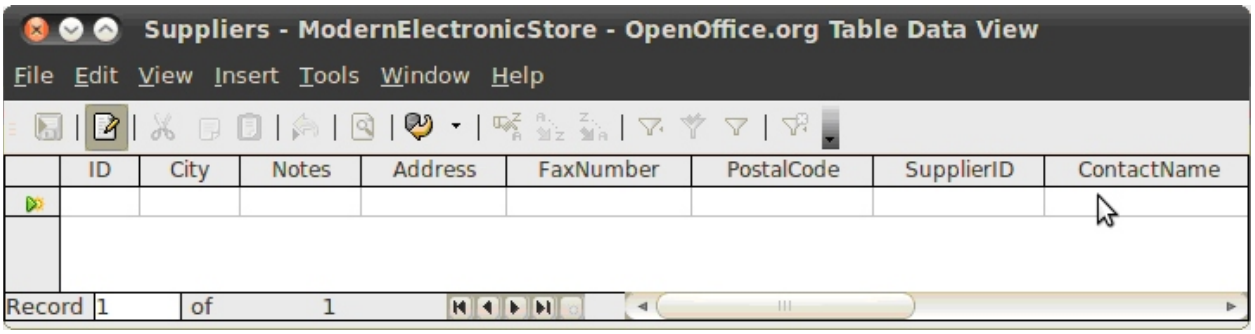


Figure 9.13 : Supplier Table in Data View

Base opens each table in a separate window. Close the Suppliers Data View window. Open ModernElectronicStore.odb window and you will find main Database window as shown in figure 9.14.

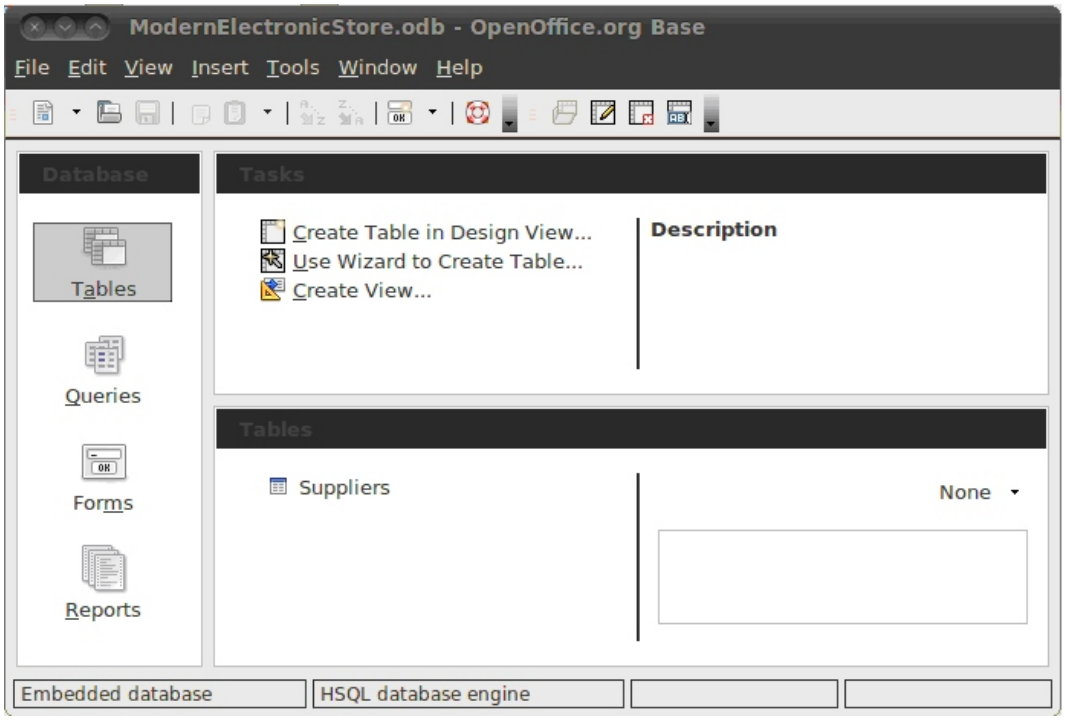


Figure 9.14 : Database Window

In the left pane, you can see Database Objects, viz., Tables, Queries, Forms and Reports. The Tables object icon would be a default selection and you can see that Suppliers table is visible under the label *Tables*. Let us view details of fields of Suppliers table. Right click on Suppliers table icon. Choose Edit option from the available options (see figure 9.15).

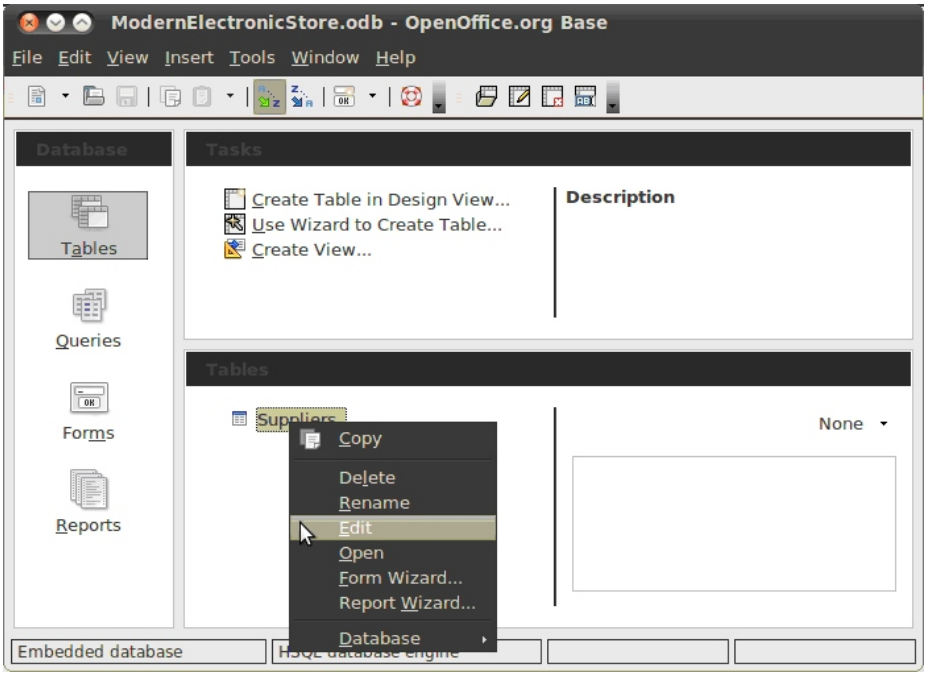


Figure 9.15 : Editing Suppliers Table

Suppliers table is now opened in Design View as shown in figure 9.16. We can now edit, update or delete the fields of the table.

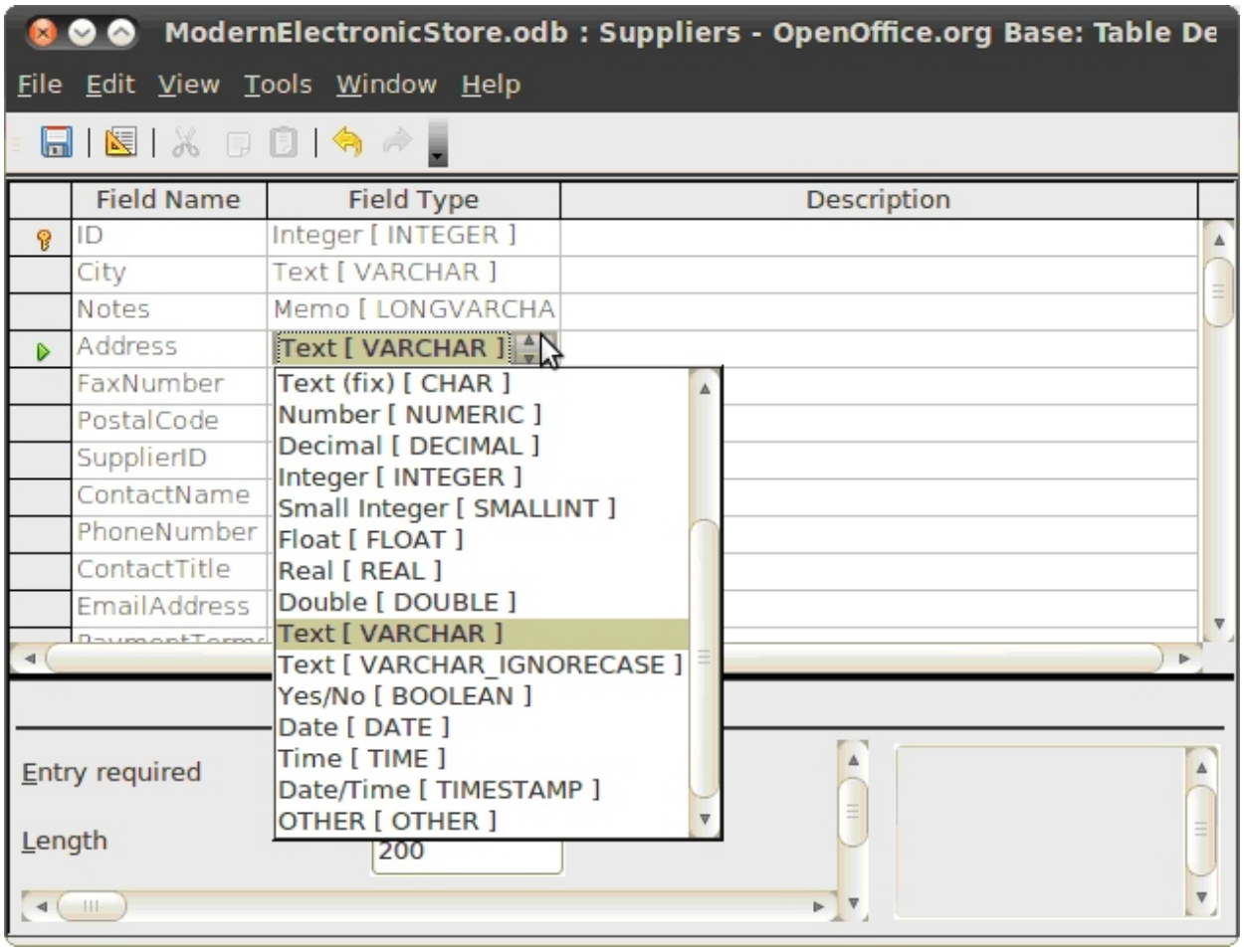


Figure 9.16 : Choosing Data type for a field

In this view defining each field is associated with four terms as stated below:

1. Field Name
2. **Field Type :** It allows us to assign a data type to each field.
3. **Description :** It is good practice to enter description for each of the field, though it is not compulsory to write description for each field of the table. The description helps the user to understand what the purpose of each field is.
4. **Field Properties :** It is used to control and validate the data that is to be entered. For each data type selected, a list of field properties related to it is displayed. It also has some default selection. We can change all the field properties as per our requirement. Various field properties will be discussed in detail in later chapter.

Earlier in this chapter we have discussed that we will be using Text[VARCHAR_IGNORECASE] data type for Address field. Click on the drop down arrow in front of Address field in *Field Type*

column, data types as shown in figure 9.16 will be displayed. Choose Text [VARCHAR_IGNORECASE] data type. You can also rename the Address field to AddressLine1 and update its data type. The Products, Employees and Customers table can be created and modified as per our requirement in the similar manner.

Deleting Objects

To delete any object created in Base, right-click the object icon and click on Delete option. Delete all the tables including the Suppliers table created through the wizard. We will recreate the required tables as per the specifications given in figure 9.6 and 9.7 using Design View.

Note : While creating the tables in Design view we will add Scode, Pcode, Ecode and Ccode field in the Supplier, Product, Employee and Customer table respectively.

Creating Table using Design View

You might have observed that when we use wizard to create a table we need to make certain changes to the pre-designed tables so that it can be readily used. In real scenario the tables provided by Base would not suffice. **Hence creating tables using Design View is more beneficial.**

Let us create table using Design View. In main Database window, under Tasks pane, two options for creation of table, *Create Table in Design View....* and *Use Wizard to Create Table...* as shown in figure 9.17 will be available. Click on the option *Create Table in Design View* and Table Design Window will be opened.

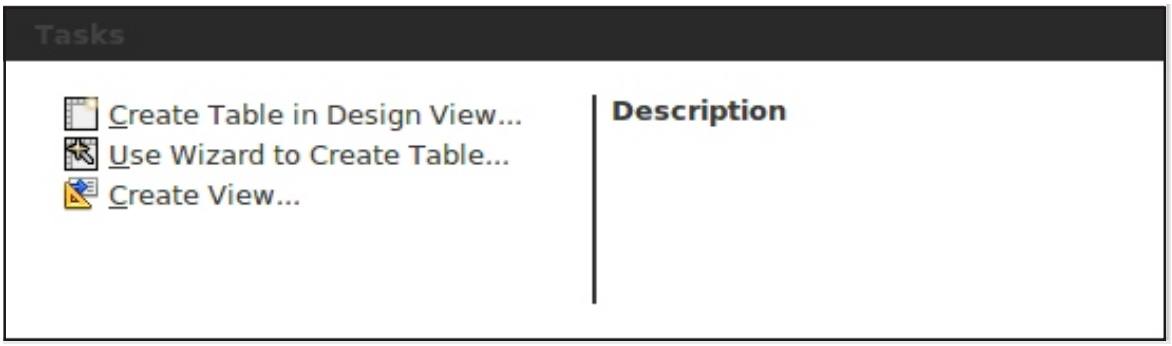


Figure 9.17 : Tasks Pane in Database Window

Table Design View as shown in figure 9.18 is grid structure with three columns, Field Name, Field Type and Description and Field Properties pane at the bottom half of the window. Create a Supplier table discussed figure 9.6. Type name of the field and select data type corresponding to each field also make sure to add Scode as the first field.

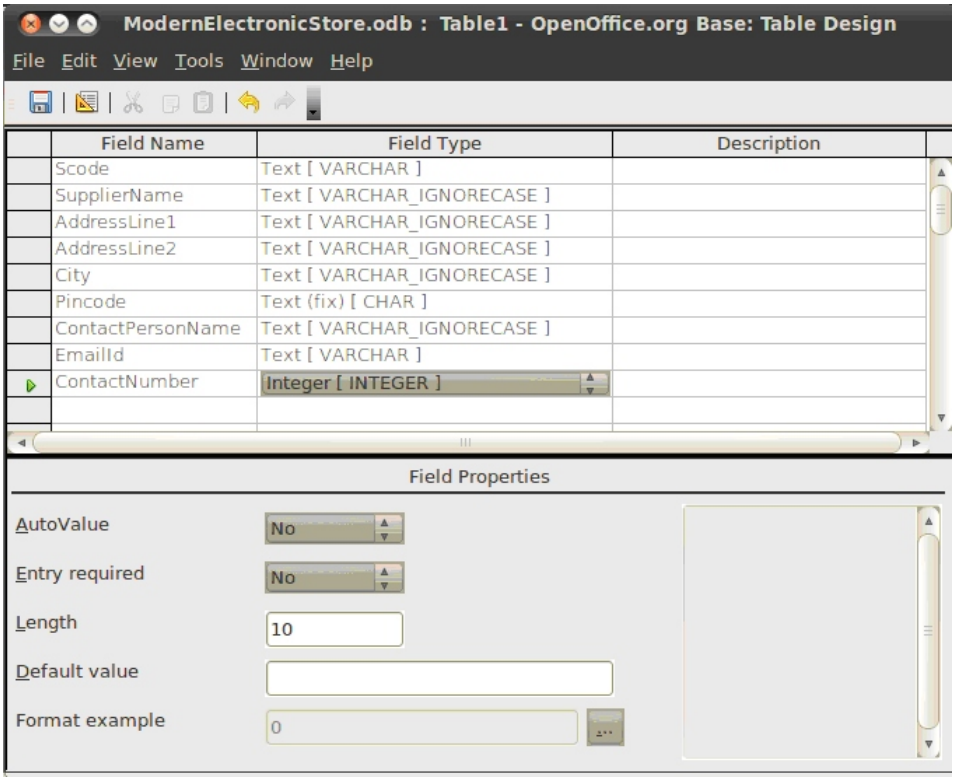


Figure 9.18 : Table Design View Window

Set Primary key

To designate a field as Primary key, right click on the Scode field to open a sub-menu as shown in figure 9.19. Select the *Primary Key* option and a symbol which looks like a key ( ) will be displayed in first column against the chosen field.

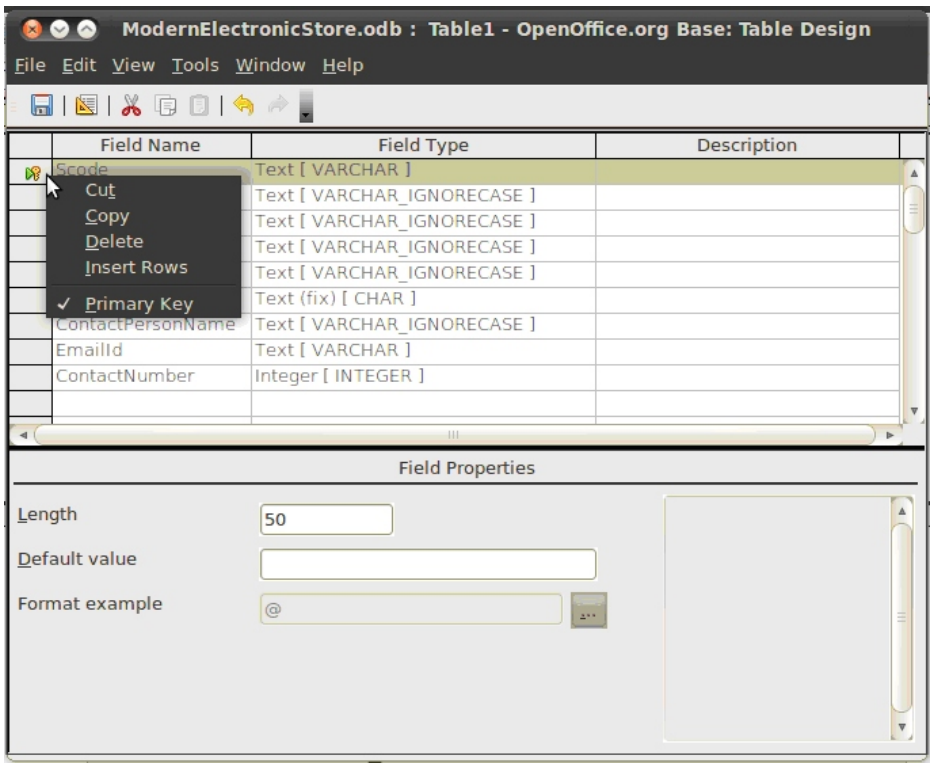


Figure 9.19 : Designating Primary key of a Table

Save Table

Having created the table structure, it's now time to save the table. Click on Save button (--) visible in figure 9.19. The Save dialog box as seen in figure 9.20 will be displayed. Type the name of the table and click on OK button.

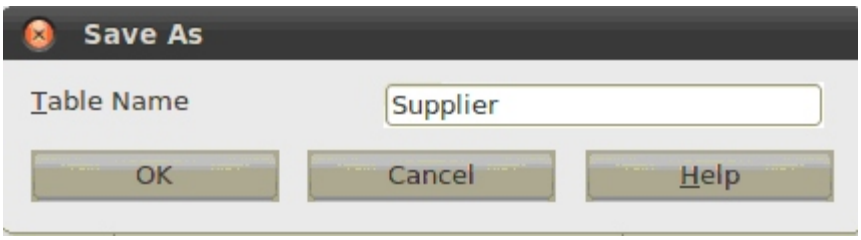


Figure 9.20 : Saving Table

Similarly create the tables Product, Employee and Customer. The tables created by us till now are known as **Master Tables**. These tables store data that has information of transactions like, customer purchasing a mobile phone from Modern Electronic Store or details of purchase made by Mobile Electronic Store from its supplier Sony India. It is suitable to design separate tables to store transactions and relate them with master tables. We will discuss transaction tables in later chapter.

Summary

In this chapter we have discussed about importance of data, information, database and Database Management System (DBMS). A raw fact generally refers to data, data when processed becomes information, and properly arranged information is stored in database. The DBMS includes different objects like table, query, report and form. We have also discussed the steps required to design the database. Further we saw the usage of OpenOffice Base software to create tables making use of Wizard and Design view. In the next chapter we will learn how to relate tables and perform operations on table records using Base.

EXERCISE

1. Explain the terms Data and Information with example.
2. Explain the need of information.
3. Define the terms database, data model and Database Management Systems.
4. Define Table, Record and Field with example and diagram.
5. What are logical field names and physical field names?
6. Write down instructions to be followed while deciding a field name.
7. Write down the steps to design the database.
8. What is the difference between Text (fix)[CHAR] and Text[VARCHAR]? Explain giving suitable example.

9. Define Key field.

10. What is primary key? Explain giving a suitable example?

11. What are the properties of primary key?

12. What is Data type? List and explain the data types available in Base.

13. When should one use memo data type?

14. List the objects available in Base.

15. What is Null value? Explain giving a suitable example.

16. **Choose the most appropriate option from those given below :**

(1) DBMS Stands for

(a) Database Management System (b) Database Migration System

(c) Data Management System (d) Direct Base Management System

(2) Which of following is a term used for "Raw facts or figures"?

(a) Information (b) Variable

(c) Data (d) Field

(3) Which of the following best describes customer in Database?

(a) Relationship (b) Attribute

(c) Entity (d) Data

(4) Processed data is known as

(a) Fact (b) Prepared data

(c) Information (d) Decision

(5) Which one of the following is not a database?

(a) MySQL (b) Base

(c) SQL Server (d) SpreadSheet

(6) Which of the following is not an Open source DBMS?

(a) MySQL (b) Ingress

(c) Base (d) Oracle

(7) The alphabet R in RDBMS stands for which of the following?

(a) Rotational (b) Relational

(c) Random (d) Ring

(8) What is the name of the database available as part of Open Office Suite?

(a) OfficeDB (b) Base

(c) OpenDB (d) Access

(9) Which of the following is the default extension of Base Database file?

(a) .bdf (b) .odf

(c) .odb (d) .dbf

(10) Which of the following is not an object maintained by Base?

(a) Tables (b) Queries

(c) Charts (d) Forms

(11) Which data type cannot be used to store numbers?

(a) Decimal (b) Integer

(c) Text (d) Date

(12) Which data type is used to store image in Base database?

(a) Binary (b) Photo

(c) Long (d) Huge

(13) Which of the following is false statement?

(a) Primary key cannot contain null values.

(b) Primary key cannot contain duplicate values.

(c) Primary key can be combination of more than one field.

(d) Primary key is always numeric field.

LABORATORY EXERCISES

1. Figure 9.21 shows the fields and their description that can be used in Student, Teacher and Subject tables. Decide data types for each field and create the tables using Base. Enter at least five records in it.

Student

Field Name	Description
Grno	General Register Number
Firstname	Name of the Student
Surname	Surname of the Student
Address	Address of the Student
City	City
Pincode	Pincode
Birthdate	Date of Birth
Gender	Male or Female
Standard	Studying in which standard
Join_date	Date of Joining School
Leaving_date	Date of Leaving School

Teacher

Field Name	Description
Firstname	Name of the Teacher
Surname	Surname of the Teacher
Address	Address of the Teacher
City	City
Pincode	Pincode
Phone_no	Phone number of teacher
Email_id	E-mail id of teacher
Mobile_no	Mobile number of teacher

Subject

Field Name	Description
Sub_Name	Name of the Subject
Details	Description of the subject

Figure 9.21 : Tables for School Management System

2. Create the transaction tables shown in figure 9.22 in Base. Enter at least five records in it.

Standard_Subj : To store details about subjects taught in each standard

Field Name	Data Type	Description
Standard	Number	Which Standard
Scode	Text	Which Subject

Attendance: To store daily attendance of the Students

Field Name	Data Type	Description
Grno	Number	General Register of a Student
Date	Date/Time	On which Date
Pr_ab	Yes/No	Present or Absent

Fees: To store details of Fee amount paid by students

Field Name	Data Type	Description
Grno	Number	General Register Number
Date	Date/Time	On which date
Tuition_fee	Currency	Tuition fee paid by the student
Term_fee	Currency	Term fee paid by the student

Subj_Teacher: To store details about which teacher is teaching the subject in each standard

Field Name	Data Type	Description
Year	Number	Current Year
Standard	Number	In which Standard
Scode	Text	Subject Code
Tcode	Text	Teacher Code

Result: To store result details of each student

Field Name	Data Type	Description
Grno	Number	General Register No.
Scode	Text	Of which Subject
Month	Text	In which Month
Year	Number	Which Year
Term	Number	Name of the Term
Marks	Number	Marks scored

Cultural: To keep record of Interest of students in Cultural Activities

Field Name	Data Type	Description
Grno	Number	General Register Number
Interest	Text	Interested in which activity
Achievement	Memo	Any achievements?

Sports: To keep record of Interest of students in Sports

Field Name	Data Type	Description
Grno	Number	General Register No.
Interest	Text	Interested in which game
Achievement	Memo	Any achievements?

Figure 9.22 : Tables for School Management System