SECTION 1 DBMS LAB

Structure **Page No.**

1.0   INTRODUCTION

By now, you must have obtained the practical skills of several programming languages. However, when we want to create a secure, managed database application, we need not re-start from scratch and develop a huge system using a programming language; rather we use a database management system: application software. This software allows us to create database, query, report and many more operations with database. This section attempts to provide you the basic skills of data organisation, including database creation, integrity enforcement, query formulation, forms and report creation, etc. You should write SQL queries as well as work using interface provided in software packages. For the present practical we have selected MS-Access. However, you must try to develop some applications using MySQL.

You must go through the MCS-023 courseware in order to get the best of those sessions. During the practical sessions you can make suitable assumptions if necessary.

1.1   OBJECTIVES

By the end of the practical sessions of this section, you will be able to:

- create databases using a user interface and SQL command;
- create integrity and constraints on databases;
- develop forms/reports using sample interface;
- write SQL queries; and
- provide a practical overview of advanced concepts like triggers, assertion, views, etc.

1.2   INTRODUCTION TO MS-ACCESS

This topic gives you an introduction to MS-Access and the basic components of MS-Access will also be discussed in this section. But before we look at the Access software and its capabilities, let us recollect what databases are, just go back to your school days, when you used to maintain different copies of your 'Home Work Assignment' and 'Class Assignment'. In those copies on the first page you used to make the 'Index', which contained the headings as Serial no., Chapter, Date, and Remarks. And under those headings, the details of all the 'Assignments' we used to store. Why did we store these details? What was that? Was it a database? Index! You mean to say that 'Index' was a database? YES. *A database is a collection of data related to a particular topic.* Employee records in a file cabinet, a stamp collection in an album – each of these collections is nothing but a database. Database, typically

consists of a heading that describes the type of information it contains, and each row contains some information. In database terminology, the columns are called fields and the rows are called records. This kind of organization in a database is called a table. A database management system (DBMS) is a system that stores and retrieves information in a database. Data management involves creating, modifying, deleting and adding data in files and using this data to generate reports or answer ad-hoc queries. The software that allows us to perform these functions easily is called a DBMS.

Microsoft Corporation introduced a Relational Database management system for the windows platform in 1992 called MS-Access. Microsoft Access is a development environment used to create computer databases.

Start the MS-Access

For starting MS-Access you must have a licensed copy of it, which is available along with MS-Office Professional.
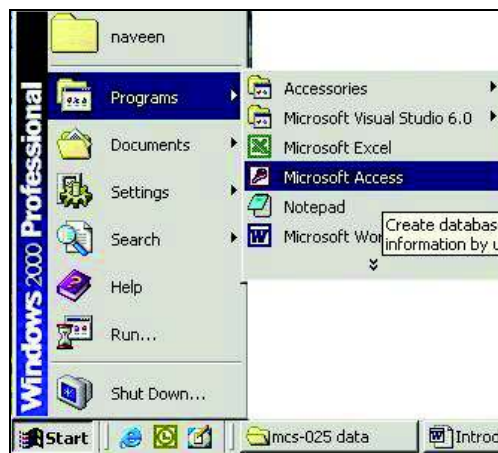

Figure 1: Starting MS-Access

After opening Access as indicated in *Figure 1* above, you will be presented with the Window shown in Figure 2. You can select one of the first two options if you are creating a new database, then go to the second option. If you want to edit an existing database, then go to the third option as shown in *Figure 2*.
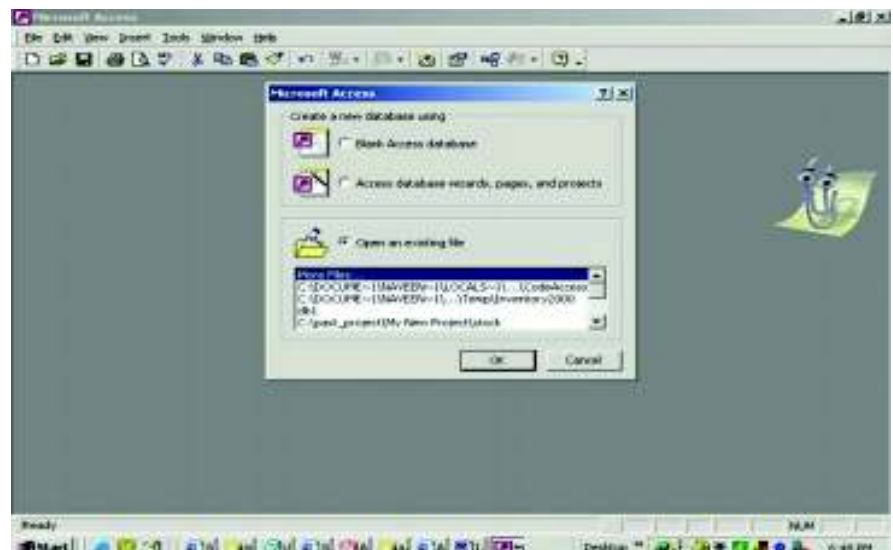

Figure 2: Starting an existing file or creating a new database

Open an existing database

If the database was opened recently on the computer, it will be listed on the main window (as shown in *Figure 2*). Highlight the database name and click OK.

To create a new database

Unlike other office software, you must save an Access database before you start working on it. After selecting "Blank Access database", you will first be prompted to specify a location and name for the database.

Figure 3: A sample Database Save Screen

You can select the folder where your database should reside and type the name of the database in the **File name** and click the **Create** button.

Database Components

The Database Window as shown below in *Figure 4* organizes all of main objects in the database like tables, queries, form and reports. Further in this we will discuss all these important components of database, which you will need in your lab exercises.
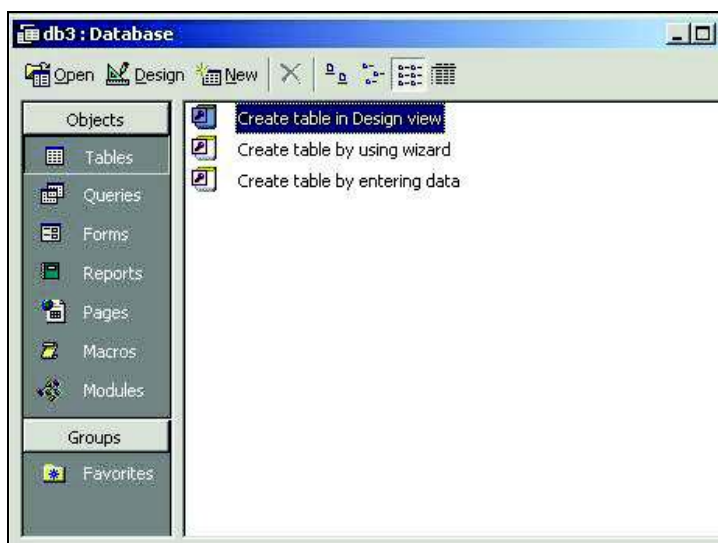
Figure 4: Database Components

Introduction to Tables

A Microsoft Access database is a collection of database files, which are also known as Tables. And each database (a table) is a collection of records, and a record is a collection of fields. You can also understand that the tables are a collection of cells that store information similar to the way an MS-Excel (If you don't know about Excel you can go and check it) worksheet does. MS-Access provides three ways to create a table.

1. Create table in Design view will allow you to create the fields of the table. (Design view is the best way for you).
2. Create table using wizard. (This is best when you are beginning to learn).
3. Create table by entering data, will give you a blank datasheet with unlabelled columns that looks much like an Excel worksheet.

Let us introduce you to Soft Garments, wholesalers for shirts, trousers, and T-shirts. They purchase from various manufacturers and wholesalers. The company has four departments – Sales, Accounts, Stores and Payroll. There are around 2000 employees working under the organization. The company wants to maintain a database, which will store the details and the entire information about all the employees. They want to store the Employee Code, Employee Name, Date of Birth, Date of Joining, Designation, Department and Photographs of the Employees.

Now, if the *Soft Garments* wants to store the employee details, they will have to make a table, which will be a part of some database. The information about one employee will make one record of that table, and the information will be stored under fields as shown in *Figure 5,* fields are *EmployeeID* and *FirstName* and others.
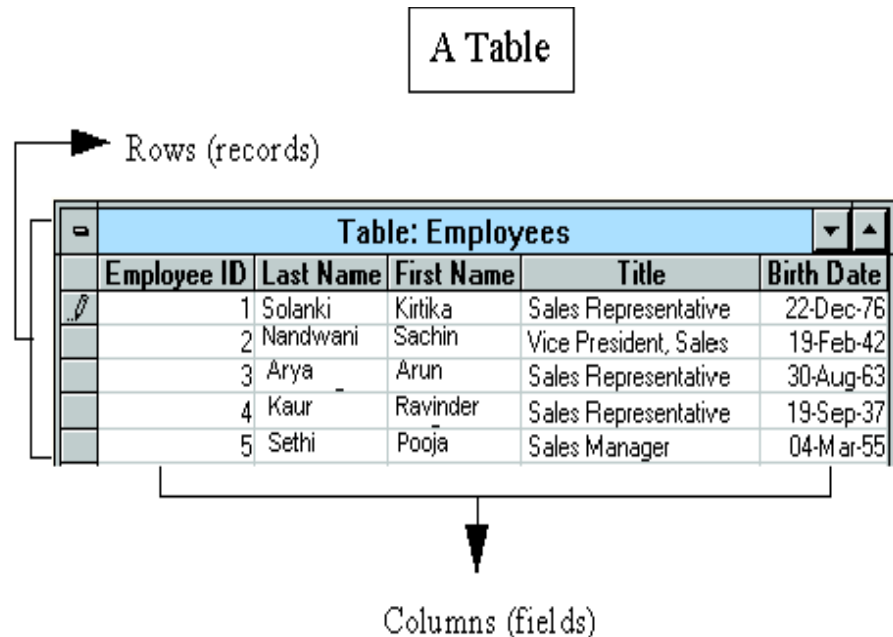

Figure 5: Records and fields of an Employees Table

Each record in a table contains the same set of fields and each field contains the same type of information for each record.

Introduction to Queries

Queries select records from one or more tables in a database so they can be viewed, analyzed, and sorted on a common datasheet. The resulting collection of records, called a dynaset (short for dynamic subset), is saved as a database object and can

therefore be easily used in the future. The query will be updated whenever the original tables are updated. Types of queries are select queries that extract data from tables based on specified values, find duplicate queries that display records with duplicate values for one or more of the specified fields, and find unmatched queries display records from one table that do not have corresponding values in a second table.
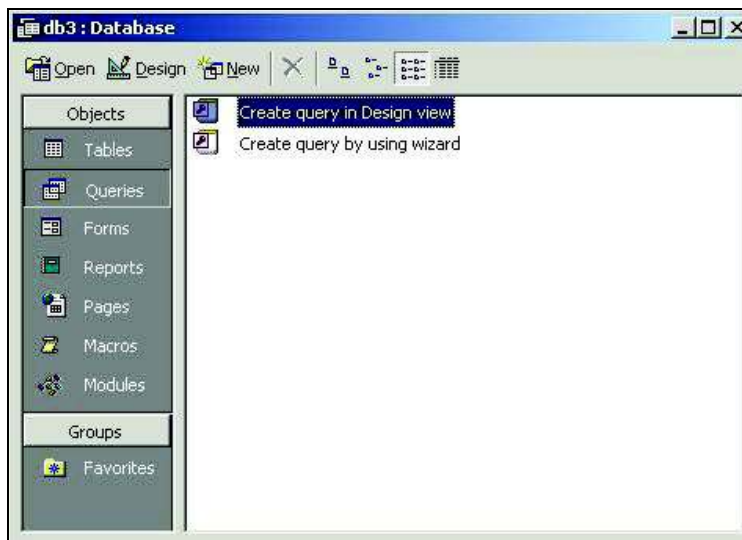

Figure 6: Creating Queries

Assume that you are a senior executive in the *Soft Garments* and heading the payroll department. One day the manager of the company calls you, and wants to know how many employees are in 'A' grade. Will you be in a position to answer that Query, right at that moment? May be Yes, May be No. Keeping track of 2000 employees is quite difficult. Not to worry. The manager had a query, he asked you. If you don't know the answer, since you kept your data in database, you can ask the '*Query*' *to* your database.

In MS-Access, A Query is a question you ask about the data in your database. The data that answers the question can be from a single table or several – the query brings the information together.

For solving the above query asked by the manager you can write the following query in access *SQL view*. As shown in *Figure 7* after performing this query on the Employee table you will get the result showing details about employees are who in Grade A. In this example, you have very few employees listed but it is really helpful when the number of employees is huge like 2000 or 20000.

SELECT [Employees].[Grade], [Employees].[EmployeeID],
[Employees].[LastName], [Employees].[FirstName], [Employees].[Title]
FROM Employees
WHERE ((([Employees].[Grade])="A"));
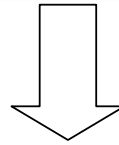
Figure 7: Result of query performed on an Employee table

Forms and Reports

Forms are used as an alternative way to enter data into a database table. There are two ways in which you can view the data, stored in a table. Those ways are:
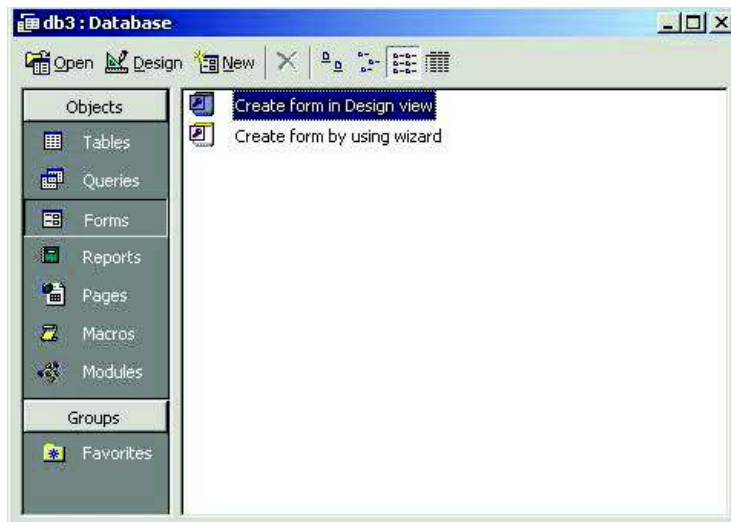
Figure 8: Creating Forms

To Create a Form Using Wizard

To create a form using the assistance of the wizard, follow these steps:

Click the Create form by using wizard option on the database window. From the Tables/Queries drop-down menu, select the table or query whose datasheet the form will modify. Then, select the fields that will be included on the form by highlighting each one, the Available Fields window and clicking the single right arrow button > to move the field to the Selected Fields window as shown in *Figure 9*. To move all of the fields to Select Fields, click the double right arrow button >>. After the proper fields have been selected, click the Next button to move on to the next screen.

Figure 9: Create Employee Form by Using Wizard

Afterwards select the layout and visual style for the form from the next set of options and click Next. On the final scree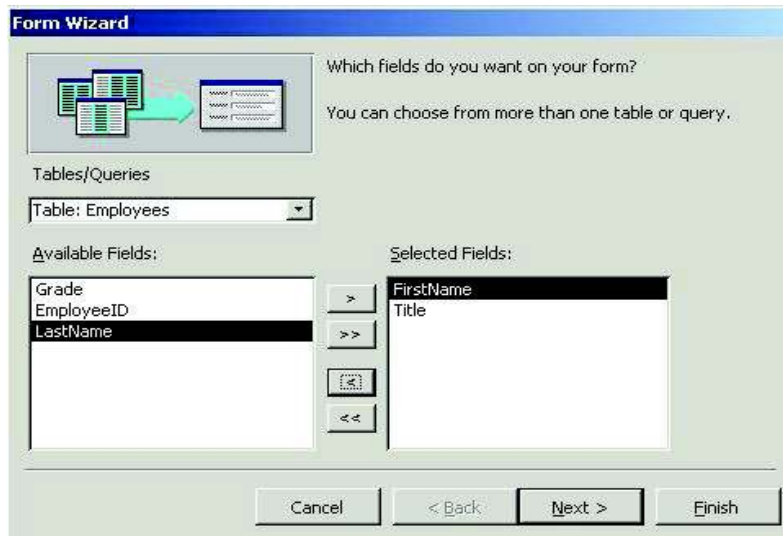n, name the form in the space provided. Select "Open the form to view or enter information" to open the form in Form View or "Modify the form's design" to open it in Design View. Click Finish to create the form.

To Create a Form using Design View

To create a form from scratch without the wizard, follow these steps:

Select "Design View" and choose the table or query the form will be associated with the form from the drop-down menu. Select View |Toolbox from the menu bar to view the floating toolbar with additional options. The toolbar contains different controls as shown in *Figure 10*.
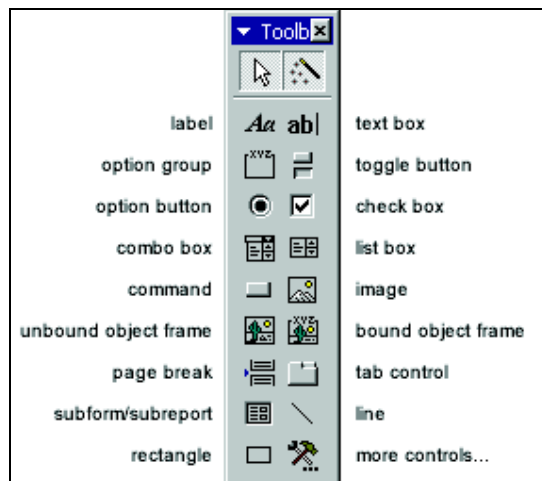
Figure 10: Different controls in Toolbar

As shown in *Figure 11* you can add controls to the form by clicking and dragging the field names from the Field List floating window. Access creates a text box for the value and label for the field name when this action is accomplished. To add controls for all of the fields in the Field List, double-click the Field List window's title bar and drag all of the highlighted fields to the form.
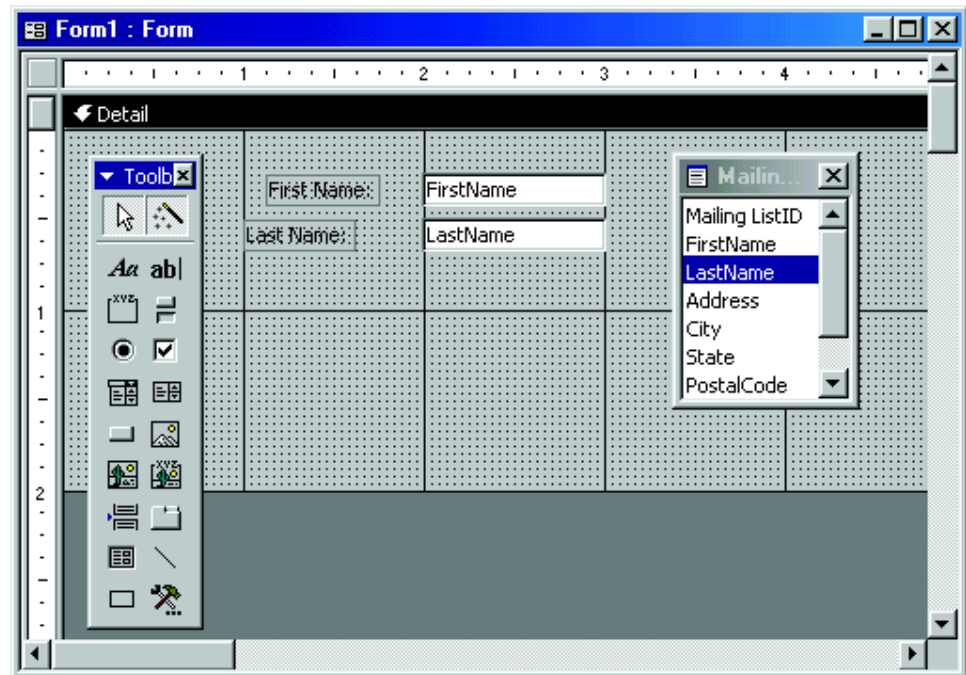
Figure 11: Adding controls to the form

A form is a customized way of viewing, entering and editing records in a database. You can specify how data is to be displayed when you design the form. Form can be created to resemble more closely the way data would be entered on paper form so that the user feels familiar with the operation.

Reports

Forms and Queries present the data on screen. Reports are used to present data on printed-paper. It provides a way to retrieve and present data as meaningful information, which might include totals and grand totals, which have to be shown across an entire set of records. Similar to Form in Reports creation also Access provides two ways for report creation. As shown in *Figure 12* you can select any way of report creation. For example in *Figure 13* you can see a report showing summary report of employee sales and category sale.

Figure 12: Creating reports

Figure 13: A sample report

1.3 DATABASE CREATION

In this section let us do some exercises relating to DBMS. The sessions are structured for your benefit.

Session 1: In this session you need to create database for an Employee management system of an ABC organisation. The details about different tables are given below. According to that you can proceed further and create tables using MS-Access.
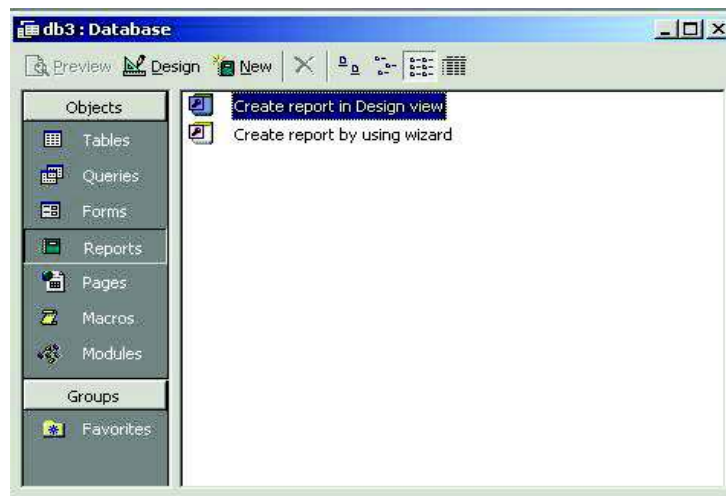
Create the following tables with the specified constraints:

Employee

First name	-	Not NULL
Middle initials	-	
Last name	-	Not NULL
Employee-id	-	Primary Key
Date of Birth	-	
Address	-	
Gender	-	M or F
Salary	-	Range of 5000 to 25000
Date of Joining	-	
Department number	-	Refers to Department Number of Department table.

Department

Department name	-	Not NULL unique
Department number	-	Primary Key
Manager_id	-	Refers to employee-id of employee table.
Manager date of joining	-	Not NULL.

Department location

Department number	-	Refers to Department number of department table.
Department location	-	Not NULL.

Department number & Department location are combined Primary Key.

Project

Project name	-	Not NULL.

13

Project number	-	Primary Key.
Project location	-	Not NULL.
Department number	-	Refers to department number of Department table.

Works-on

Employee-id	-	Not NULL refers to employee-id of employee table.
Project number	-	Not NULL refers to Project number of Project table.
Hours	-	Not NULL.

Employee-id & Project number are combined primary key.

Dependent

Employee-id	-	Refer to employee table employee id field
Dependent name	-	
Gender	-	M or F
Date of Birth	-	Not NULL
Relationship	-	Not NULL

Now enter a few sets of meaningful data and answer the following queries.

1. List the department wise details of all the employees.
2. Find out all those departments that are located in more than one location.
3. Find the list of projects.
4. Find out the list of employees working on a project.
5. List the dependents of the employee whose employee id is '001'

Session 2:

This session is similar to the previous one, but in this session assume that you are developing a prototype database of the IGNOU library management system, for that you need to create the following tables:

(a) Book Records
(b) Book details
(c) Member details and
(d) Book issue details

Structure of the tables are given below:

Table Name	Attribute Name
Book Records	Accession Number
	ISBN Number
Books	ISBN Number
	Author
	Publisher
	Price
Members	Member Id
	Member Name
	Maximum Number of books that can be issued
	Maximum Number of days for which book can be issued
Book Issue	Member Id
	Accession Number

Issue Date
Return Date

You must create constraints, including referential integrity constraints, as appropriate. Please note accession number is unique for each book. A book, which has no return date, can be considered as issued book. Enter suitable data into the tables. Now answer the following:

1. Insert data in all the three tables (use insert).
2. Insert appropriate description associated with each table and the column (use comment).
3. Display the structure of the tables.
4. Display the comments that you have inserted.
5. Using SELECT statement, write the queries for performing the following function:

   (a) Get the list of all books (No need to find number of copies)
   (b) Get the list of all members
   (c) Get the Accession number of the books which are available in the library
   (d) On return of a book by a member calculate the fine on that book.
   (e) List of books issued on 01-Jan-2005
   (f) Get the list of all books having price greater than Rs. 500/-
   (g) Get the list of members who did not have any book issued at any time.
   (h) Get the list of members who have not returned the book.
   (i) Display member ID and the list of books that have been issued to him/her from time to time.
   (j) Find the number of copies of each book (A book accession number would be different but ISBN number would be the same).
   (k) Find the number of copies available of a book of given ISBN number.
   (l) Get the member ID and name of the members to whom no more books can be issued, because they have already got as many books issued as the number for which they are entitled.

1.4 USE OF DBMS TOOLS/ CLIENT-SERVER MODE

Session 3:

This session is based on Session 2 where you have created a library management system. In this session you have different **query specification.** You must create appropriate forms, reports, graphs, views and data filtering, use of multilevel report, etc. to answer these queries.

1. Get the list of ISBN-Number, Book name, available copies of the books of which available copies are greater than zero.
2. Get the list of ISBN-Number, Book name, Total copies, available copies of the book of which available copies are greater than zero. List should be displayed in alphabetical order of book name.
3. Get the list of ISBN number, Book name, Author, total copies, cost (cost is price × total copies). List should be displayed in descending order of cost.
4. Get the list of books issued to each member.
5. Write query to know the maximum and average price of the books.
6. Get the list of all existing members and the number of days for which a member is allowed to keep the book. Also find out the members who have got the maximum number of books issued.
7. Get the list of *member codes* of those members who have more than two books issued.

8. Find the details of the books presently issued to a member.
9. Create the history of issue of a book having a typical accession number.
10. To set the width of the book name as 35.

Session 4:

Create the following table and perform the necessary tasks defined below one by one. You must use the query tools/ SQL/ Reports/ Forms/ Graphs/Views/ using client/server wherever needed.

1. Create the following table named **customer**

Column name	type	size
Customer id	Character	10
Name	Character	25
Area	Character	3
Phone	Numeric	7

Insert the appropriate data into table.

a. Update Phone numbers of all customers to have a prefix as your city STD Code
b. Print the entire customer table
c. List the names of those customers who have 'e' as second letter in their names.
d. Find out the Customer belonging to area 'abc'
e. Delete record where area is NULL.
f. Display all records in increasing order of name.
g. Create a table temp from customer having customer-id, name, and area fields only
h. Display area and number of records within each area (use GROUP by clause)
i. Display all those records from customer table where name starts with 'a' or area is "abc."
j. Display all records of those where name starts with 'a' and phone exchange is 55.

2. Answer the following queries using Library system as created earlier. You must create a view to know member name and name of the book issued to them, use any inbuilt function and operators like IN, ANY, ALL, EXISTS

a. List the records of members who have not been issued any book using EXISTS operator.
b. List the members who have got issued at least one book (use IN / ANY operator).
c. List the books which have maximum Price using ALL operator.
d. Display Book Name, Member Name, Issue date of Book. Create a view of this query of the currently issued books.

3. Create a table of Employee (emp-number, name, dept, salary) and Department (dept number, dept name). Insert some records in the tables through appropriate forms having integrity checks. Add some records in employee table where department value is not present in department table. Now answer the following query:

a. Display all records from employee table where department is not found in department table.
b. Display records from employee table in a report format with proper headings. This report must also contain those records where department number does not match with any value of department table.

c.   Display those employee records who have salary less than the salary of person whose empcode= 'A100'.
d.   Create another table : Sales_data (Region-code, City, Salesperson-code, Sales-qty).
e.   Display records where salesperson has achieved sales more than average sales of all sales persons of all the regions.

Session 5:

For the following queries use Library System as created by you in earlier sessions. You must use the query tools/ SQL/ Reports/ Forms/ Graphs/Views/ using client/server wherever needed.

1.   Get the list of books presently issued to the members, along with the names of the book as well as names of the members.
2.   Get the list of the members who
     (a)   are entitled for more books than that the entitlement of member name "abc".
     (b)   are issued the books for more days than the number of days for "abc".

3.   Find out the history of issuing of a list of books that has been identified during inspection as damaged books. (Create the necessary tables if needed).
4.   Create the tables Item master and Transaction having following format:

Item Master:	Transaction:
Item-code	item-code
Item-name	Quantity
Price	Date of transaction

     Set the foreign key constraints in the tables and insert at least 5 records having meaningful data. Now answer the following queries.

a.   Display Item-code, Name, Quantity, Date of transaction, where
     Sales amount = Quantity *Price using a report.
b.   Display all transactions of item 'X' using a report.
c.   Display all the items whose price is more than the price of item 'X'.
d.   Saving the previous query into a temporary file.
e.   Store the database in a new file.
f.   Create all the forms for data entry and create at least 5 meaningful reports.

1.5   FORMS AND PROCEDURES

This section covers design and implementation of different kinds of forms to create user interactivity. Also, you can design different procedures/triggers to perform different operations on databases.

Session 6:

1.   Create the following tables:

     Order party: (Order number, Order date, customer code)
     Order: Order number, Item code, Quantity

     The key to the second table is order-number + item-code
     Create a **form** for data entry to both the tables.

2.   Create a form for storing Bio-data of students. Create the supporting tables to store the data.

3. Design a suitable form for storing basic information and salary details of employees of an organisation. Design and implement the necessary tables.

Session 7:

1. Write a procedure/trigger on department code so such that the validity of the code is checked and the name of department is automatically displayed on selection of department code. Assume, design and create the necessary tables and constraints.

2. Write a procedure/trigger on a numeric field named value1 to check if the entered value is 1 (Married) or 2 (Unmarried). In case, the entered value is 1 (Married) then the control should pass to a field named 'spouse name' or else it goes to a field named: Father's Name.

3. Employee code must begin with 'P' (Permanent) or 'T' (Temporary) and its second character must be a digit. Write procedure/trigger to check if the entered value is correct.

4. Write a procedure/trigger to generate Order Number automatically in any of the order tables created in Session 6.

Session 8:

1. Design a form that shows the status of books pending on a member on entering the member-id.
2. Design a **form** that modifies the records of an Item Table having the fields: Item Code, Item Name, Quantity, Price, Re-order Level.

   (a) Enter the Item Code and get all the details from the tables
   (b) Check if negative values are entered in the field.

3. Design the **form** to display the leave information of each employee following. The validations must be made for the fields:

   - Leave information of every employee must be display grouped by month
   - Display total of all leave taken.

Let us now perform all the operations you have practiced till now. You must use the query tools/ SQL/ Reports/ Forms/ Graphs/Views/ procedures/ using client/server wherever needed.

Session 9:

1. Add one more table employee with fields employee-number, employee-name, Basic pay, Department in the Library management system.
2. Add a new column Date of Joining in the table.
3. Modify the length of field employee name.
4. Delete the column basic from basic pay.
5. Find the details of members who were issued a book prior to Feb 1st 2005.
6. In previous query 5, list the details of the particular members.
7. In previous query 5, list the details of only two such members.
8. List the details of the persons who have been issued at least one book.
9. List the names of three persons who have not been issued any book.
10. List of members, who are entitled for 5 books or are issued the books for 15 days.
11. List the names of members in fixed length of 30 characters followed by their codes in parenthesis and with first character of the name in capital.
12. Find the list of the members who have been issued the books having the same ISBN number.

13. Display book issue/return data of various books in the following form

    Book Accession number.   Book Title   Issued on   Returned on

Session 10:

1. Create the following tables for a video library after normalizing these tables:

Customer

Customer_id	Primary Key	Not NULL
Name		Not NULL
Area		
Phone_number		

Movie

Movie_no	Primary Key	Not NULL
Title	<film title>	Not NULL
Type	Action or Thriller or Romance or Comedy or Suspense or Horror etc.	
Actors	Not NULL	
Rent-Price	Not NULL	
Rent applicable	data part of primary key	

Issues

Issue_no	Primary Key	Not NULL
Movie_no	Refers to Movie_no of movie table	
Customer_id	Refers to Customer_id of Customer table	
Issue_date	not greater than current date.	
Return_date	not greater than current date.	

Write down SQL statements to perform the following Queries:

1. List the names of all the customers.
2. Print the entire customer table.
3. List the name and area of all the customers.
4. List the various movie types available.
5. List the names of all customers having 'i' in any case as the second letter in their names.
6. List the names of all customers that begin with 's' or 'j'.
7. Print the list of employees whose phone numbers have area code as 011.
8. Print the information of customers who have been issued movies in the month of February.
9. List the movies that have been issued to the customers with customer-id between '9000' and '9999'.
10. List the names of movies whose Rent - price is greater than Rs. 100/-.
11. Increase the Rent-price of each movie by 10%. Modify rent applicable data suitably.
12. List the movies in sorted order of their title, and types of all the movies except Drama.
13. Find the recovery made from each movie.
14. Calculate the total revenue of all movies.
15. Determine the maximum and minimum movie prices and Rename the title as Maximum Price.
16. List the Movies which are issued to customers for more than a week.
17. Print the type, average price, total number of prints, for each type of movie.
18. Find out the movies issued to customer 'X'.

19. Find out the names of the movies that have been issued to the maximum number of customers.
20. Display the month in which customers take the maximum number of movies.
21. Display the history sheet of each movie.
22. List the customers who have not been issued any movie in the last 6 months.

1.6 SUMMARY

This section has provided you with problems with respect of creation of database and integrity using constraints and using an interface and also using SQL commands. Some of the exercises provided include creation of forms and reports, creation of SQL queries and an overview of various databases related concepts. We hope by now you must be familiar with at least one database application and would be able to migrate to other DBMSs.

1.7 FURTHER READINGS

Reference Books

- MS-Access user guide.
- Microsoft Access 2000 Bible by Cary Prague and Michael Irwin, IDG Books.
- Access 2003 Bible by Cary N. Prague, Michael R. Irwin, Jennifer Reardon; John Wiley & Sons publication.

Web references and tutorials

http://mis.bus.sfu.ca/tutorials/MSAccess/tutorials.html
http://www.aspfree.com/c/b/Microsoft-Access/
http://netforbeginners.about.com/od/msaccess/
http://www.vbtutor.net/vbtutor.html
http://www.w3schools.com/sql/default.asp
http://sirius.cs.ucdavis.edu/teaching/sqltutorial/