

**MGM's Jawaharlal Nehru Engineering College
N-6 CIDCO, Aurangabad**

**Department of Information Technology
Lab Manual
Academic Year 2019-2020**

Class: Third Year IT

Course Code: BTITC501

Course Title: Database Management System

Semester-V

**Prepared by:
Mr. Amol Mahadik
Course Coordinator**

MGM's
Jawaharlal Nehru Engineering College
Department of Information Technology

Vision of the department

To develop expertise of budding technocrats by imparting technical knowledge and human value based education.

Mission of the department

- A. Equipping the students with technical skills, soft skills and professional attitude.
- B. Providing the state of art facilities to the students to excel as competent professionals, entrepreneurs and researchers.

Program Specific Outcomes

- PSO1. An ability to design, develop and implement computer programs in the areas related to Algorithms, Multimedia, Website Design, System Software, DBMS and Networking.
- PSO2. Develop software systems that would perform tasks related to Research, Education and Training and/or E governance.
- PSO3. Design, develop, test and maintain application software that would perform tasks related to information management and mobiles by utilizing new technologies to an individual or organizations.

Program Educational Objectives

- PEO1. The graduates will utilize their **expertise** in IT industry and solve industry technological problems.
- PEO2. Graduates should excel in **engineering positions** in industry and other organizations that emphasize design & implementation of IT applications.
- PEO3. Graduates will be **innovators & professionals** in technology development, deployment & system implementation.
- PEO4. Graduates will be pioneers in engineering, engineering management, research and **higher education**.
- PEO5. Graduates will be good citizens & cultured human being with full appreciation of importance of IT **professional ethical & social** responsibilities.

HOD
Information Technology

Program Outcomes:

- PO1. Engineering Knowledge
- PO2. Problem Analysis
- PO3. Design/Development of Problem
- PO4. Conduct Investigation of Complex Problem
- PO5. Modern Tool Usage
- PO6. The Engineer and society
- PO7. Environment and Sustainability
- PO8. Ethics
- PO9. Individual and Team work
- PO10. Communication
- PO11. Project Management
- PO12. Life Long Learning

Lab Outcomes: Students will be able to:

LO1 Use an SQL interface of a multi-user relational DBMS package to create, secure, populate, maintain, and query a database.

LO2 Manage users ,assign roles , access rights to different users in DBMS

LO3 Write PL/SQL programs to query database

LO-PO Mapping

(Mapping scale: 1: Low, 2: Medium, 3: High)

| Lab | Program Outcomes | | | | | | | | | | | | |
|------|------------------|-----|-----|-----|-----|-----|-----|-----|-----|----|----|----|---|
| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO | PO | PO | |
| LO 1 | 3 | 2 | 2 | 2 | 3 | | | | | | | 2 | 2 |
| LO 2 | 3 | 2 | 2 | 2 | 3 | | | | | | | 2 | 2 |
| LO 3 | 3 | 2 | 2 | 2 | 3 | | | | | | | 2 | 2 |

LO-PSO Mapping

(Mapping scale: 1: Low, 2: Medium, 3: High)

| Course Outcomes | PSO1 | PSO2 | PSO3 |
|-----------------|------|------|------|
| LO 1 | 2 | 2 | |
| LO 2 | 2 | 2 | 2 |
| LO 3 | 2 | | 2 |

***PO: Program Outcomes**

***PSO: Program Specific Outcomes**

FOREWARD

It is my great pleasure to present this laboratory manual for third year engineering students for the subject of Software engineering keeping in view the vast coverage required for process involved in database development.

As a student, many of you may be wondering with some of the questions in your mind regarding the subject and exactly what has been tried is to answer through this manual.

As you may be aware that MGM has already been awarded with ISO 9000 certification and it is our endure to technically equip our students taking the advantage of the procedural aspects of ISO 9000 Certification.

Faculty members are also advised that covering these aspects in initial stage itself, will greatly relived them in future as much of the load will be taken care by the enthusiasm energies of the students once they are conceptually clear.

Dr H H Shinde
Principal

LABORATORY MANUAL CONTENTS

This manual is intended for the Third year students of Information Technology in the subject of Database management Systems. This manual typically contains practical/Lab Sessions related Database management Systems covering various aspects related the subject to enhanced understanding.

Although, as per the syllabus, study of creating database is prescribed, we have made the efforts to cover various aspects of Database management Systems covering different aspects of creating database to develop database which contains Writing basic SQL SELECT statements, Restricting and sorting data. Displaying data from multiple tables, Aggregating data using group functions, Manipulating data, Creating and managing tables, Including constraints, Creating views, and Controlling user access.

Students are advised to thoroughly go through this manual rather than only topics mentioned in the syllabus as practical aspects are the key to understanding and conceptual visualization of theoretical aspects covered in the books.

Good Luck for your Enjoyable Laboratory Sessions

SUBJECT INDEX

- 1) Study of E-R Diagram
- 2) Study of Data Definition Language commands
- 3) Study of Data Manipulation Language
- 4) Study of data communication using SQL functions.
- 5) Writing sub queries on join operation
- 6) Creating Views
- 7) Writing and Executing PL/SQL block
- 8) Study of GRANT and REVOKE DBA commands
- 9) Normalization in oracle

DOs and DON'Ts in Laboratory:

1. Do not handle any equipment before reading the instructions/Instruction manuals
2. Read carefully the power ratings of the equipment before it is switched on whether ratings 230 V/50 Hz or 115V/60 Hz. For Indian equipments, the power ratings are normally 230V/50Hz. If you have equipment with 115/60 Hz ratings, do not insert power plug, as our normal supply is 230V/50 Hz, which will damage the equipment.
3. Observe type of sockets of equipment power to avoid mechanical damage
4. Do not forcefully place connectors to avoid the damage
5. Strictly observe the instructions given by the teacher/Lab Instructor

Instruction for Laboratory Teachers::

1. Submission related to whatever lab work has been completed should be done during the next lab session. The immediate arrangements for printouts related to submission on the day of practical assignments.
2. Students should be taught for taking the printouts under the observation of lab teacher.
3. The promptness of submission should be encouraged by way of marking and evaluation patterns that will benefit the sincere students.

WARMUP EXERCISES:

Define Database.

Define tables and how to create it.

How can you create a user in database environment?

How database is restricted.

What manipulations you can do on tables?

What is SQL?

What is relational model?

What are group functions?

How can you create views?

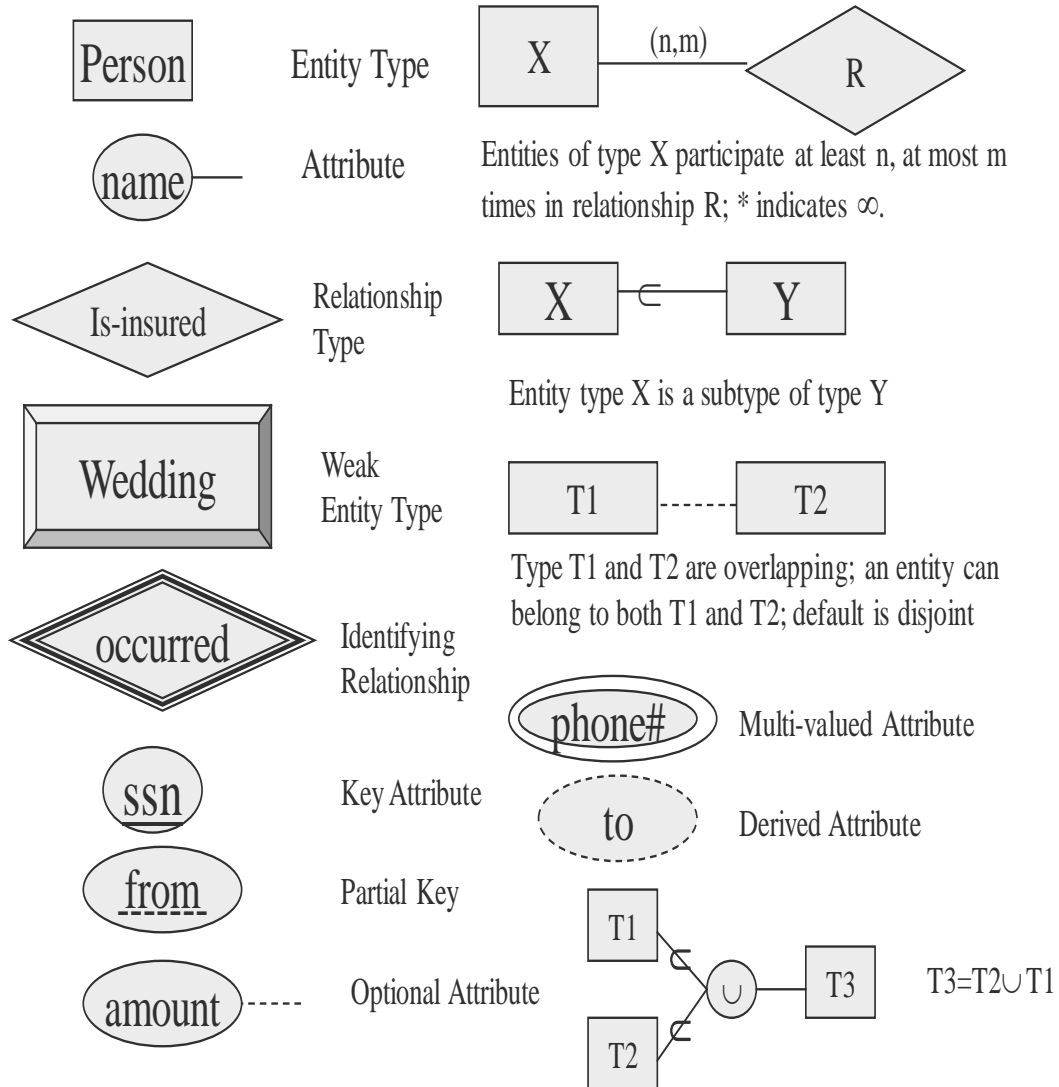
How can you retrieve data?

1. Lab Exercises:

Exercise No 1: (2 Hours) – 1 Practical

Study of E-R Diagram

E-R diagram is a popular, high-level conceptual data model. This model and its variations are frequently used for conceptual design of database applications, and many database design tools employ its concepts. E-R model diagrammatic notations are as follows:



E/R Model Symbols for COSC 3480

Assignment:

Design a “good” entity-relationship diagram for Bank Database. Indicate the cardinalities for each relationship type; assign roles (role names) to each relationship if there are ambiguities! Use sub-types, if helpful to express constraints!

Bank Database

The bank is organized into branches. Each branch is located in a particular city and is identified by a unique name. The bank monitors the assets of each branch.

Bank customers are identified by their `customer_id` values. The bank stores each customer’s name and the street and city where the customer lives. Customers may have accounts and can take out loans. A customer may be associated with a particular banker, who may act as loan officer or personal banker for the customer.

Bank employees are identified with their `employee_id` values. The bank administrator stores the name and telephone number of each employee, the names of the employee’s dependents, and their `employee_id` number of the employee’s manager. The bank also keeps track of the employees start date and thus the length of employment.

The bank offers two types of accounts – the savings and the checking accounts. Accounts can be held by more than one customer, and a customer can have more than one account. Each account is assigned a unique account number. The bank maintains a record of each account’s balance and the most recent date on which account was accessed by each customer holding the account. In addition, each savings account has an interest rate and overdrafts are recorded for each checking account.

A loan originates at a particular branch and can be held by one or more customers. A loan is identified by a unique loan number. For each loan, the bank keeps track of the loan amount and the loan payments. Although a loan payment number does not uniquely identify a particular payment among all those for all the bank loans, a payment number does identify a particular payment for a specific loan. The date and amount are recorded for each payment.

2. Lab Exercises:

Exercise No2: (2 Hours) – 1 Study Practical

Study of Data Definition Language (DDL)

DDL is a language that allows the DBA or user to describe and name the entities, attributes and relationships required for the application, together with any associated integrity and security constraints. The database schema is specified by a set of definition expressed by means of a special language called a Data Definition Language. The DDL is used to define a schema or to modify an existing one; it cannot be used to manipulate data.

The result of the compilation of The DDL statements is a set of tables stored in special files collectively called the system catalog. The system catalog integrates the metadata, is data that describes objects to be accessed or manipulated.

The SQL DDL allows database objects such as schemas, domains, tables, views and indexes to be created and destroyed.

The main SQL DDL statements are:

- **CREATE TABLE** TableName
 {(ColumnName dataType [**NOT NULL**] [**UNIQUE**]
 [**DEFAULT** defaultOption] [**CHECK** (searchCondition)] [,...]}
 [**PRIMARY KEY** (listOfColumns),]
 {[**UNIQUE** (listOfColumns),] [,...]}
 {[**FOREIGN KEY**(listOfForeignKeyColumns)
 REFERENCES ParentTableName [(listOfCandidateKeyColumns)],
 [**MATCH** {**PARTIAL** | **FULL** }
 [**ON UPDATE** referentialAction]
 [**ON DELETE** referentialAction]] [,...]}
 {[**CHECK** (searchCondition)] [,...]}))
- **ALTER TABLE** TableName
 [**ADD** [**COLUMN**] columnName dataType [**NOT NULL**] [**UNIQUE**]
 [**DEFAULT** defaultOption] [**CHECK** (searchCondition)]
 [**DROP** [**COLUMN**] columnName [**RESTRICT** | **CASCADE**]]
 [**ADD** [**CONSTRAINT** [constraintName]] tableConstraintDefinition]
 [**DROP** [**CONSTRAINT** constraintName [**RESTRICT** | **CASCADE**]]
 [**ALTER** [**COLUMN**] **SET DEFAULT** defaultOption]
 [**ALTER** [**COLUMN**] **DROP DEFAULT**]
- **DROP TABLE** TableName [**RESTRICT** | **CASCADE**]

Assignment

Following Tables are used for all assignments given in the lab manual.

Hotel (HotelNo, HotelName, city)

Room (RoomNo, HotelNo, type, Price)

Booking (HotelNo, GuestNo, DateFrom, DateTo, RoomNo)

Guest (GuestNo, GuestName, GuestAddress)

PropertyForRent (PropertyNo, street, city, postcode, Type, rooms, rent, ownerNo, StaffNo, BranchNo)

Staff (StaffNo, FName, LName, position,sex , dob, salary, branchNo)

Branch (BranchNo, Street, City, PostCode)

Client (ClientNo, FName, LName, TelNo, PrefType, MaxRent)

PrivateOwner (OwnerNo, FName, LName, Address, TelNo)

Viewing (ClientNo, PropNo, ViewDate, Comment)

Registration (ClientNo, BranchNo, StaffNo, DateJoined)

- Create the PropertyForRent table using the available features of the CREATE TABLE statement.
- Change the Staff table by removing the default of 'Assistant' for the position column and setting the default for the sex column to Female.
- Change the PropertyForRent table by removing the constraint that staff is not allowed to handle more than 100 properties at a time .Change the client table by adding a new column representing the preferred number of rooms.
- Remove the propertyforrent table
- Create the Hotel table using the integrity enhancement feature of SQL.
- Create the Room, Booking and Guest tables using the integrity enhancement feature of SQL with the following constraints.
 - Type must be one of single, double, or family.
 - Price must be between \$10 and \$100.
 - Roomno must be between 1 and 100.
 - Datefrom and dateto must be greater than today's date.
 - The same room cannot be double booked.
 - The same guest cannot have overlapping bookings.

3. Lab Exercises:

Exercise No 3: (2 Hours) – 1 Study Practical

Study of Data Manipulation Language (DML) COMMANDS:

DML is a language that provides a set of operations to support the basic data manipulation operations on the data held in the database. Data manipulation operations usually include the following.

- Insertion of new data into the database
- Modification of data stored in the database
- Retrieval of data contained in the database.
- Deletion of data from the database.

The SQL DML statements are:

- SELECT –To query data in the database

```
SELECT      [DISTINCT | ALL] {*} | [columnExpression [AS newName]] [,...]  
FROM       TableName [alias] [,...]  
[WHERE      condition]  
[GROUP BY  columnList] [HAVING condition]  
[ORDER BY  columnList]
```

- INSERT – To insert data into the table

```
INSERT INTO TableName [(ColumnList)]  
VALUES (data ValueList)
```

- UPDATE – To update data in a table

```
UPDATE TableName  
SET columnName1=dataValue1 [, columnName2=dataValue2....]  
[WHERE searchCondition]
```

- DELETE – To delete data from a table

```
DELETE FROM TableName  
[WHERE searchCondition]
```

Assignment

- List full details of all Staff.
- Produce a list of salaries for all staff showing only the staff number, the first and last names, and the salary details.
- List the property numbers of all properties that have been viewed.
- Produce a list of monthly salaries for all staff, showing a staff number, the first and last names and the salary details.
- List all staff with a salary greater than \$10000.
- List the addresses of all branch offices in London or Glasgow.
- List all staff with a salary between \$20000 and \$30000.
- List all managers and supervisors.
- List the details of all viewings on Property PG4 where a comment has not been supplied.
- Produce a list of salaries for all staff range in descending order of salary.
- List the details of PrivateOwners where address field contains a string 'Glasgow'.
- Produce an abbreviated list of properties range in order of property type.
- How many properties cost more than \$350 per month to rent?
- How many different properties were viewed in May 2001?
- Find the total no of managers and the sum of their salaries.
- Find the Min and Max staff salaries.
- Find the no of staff working in each branch and the sum of their salaries.
- For each branch office with more than one member of staff find the no of staff working in each branch and the sum of their salaries.
- List the staffs who work in the branch at '163 Main Street'.
- List all staff whose salary is greater than the average salary and show by how much their salary is greater than the average.
- List the properties that are handled by staff who work in the branch at '163 Main Street'.
- Find all staff s whose salary is larger than the salary of at least one member of staff at branch B003.
- Find all staff whose salary is larger than the salary of every member of staff at branch B003.
- Insert a new row into the staff table supplying data for all columns.
- Populate the StaffPropCount table using details from the staff and PropertyForRent table.
- Give all staff a 3% pay increase
- Give all managers a 5% pay increase
- Promote David Ford (StaffNo-SG14) to manager and change his salary to \$18000.
- Delete all viewings that relate to property PG4.
- Delete all rows from the viewing table.

4. Lab Exercises:

Exercise No 4: (2 Hours) – 1 Practical

Study of data communication using SQL functions:

Functions serve the purpose of maintaining data items and returning a result .Functions are also capable of accepting user supplied values and constants and manipulating them. Such user defined values or constants are called arguments. Any number of arguments can be passed to function. Functions that act on only one value at a time are called scalar functions. Functions can be classified corresponding to different data types as:

Aggregate Functions

- AVG (DISTINCT | ALL | n)
- MIN (DISTINCT | ALL | expr)
- COUNT (DISTINCT | ALL | expr)
- COUNT (*)
- MAX (DISTINCT | ALL | expr)
- SUM (DISTINCT | ALL | n)

Numeric Functions

- ABS(n)
- POWER(m,n)
- ROUND(n,m)
- SQRT(n)

String Functions

- LOWER(char)
- INITCAP(char)
- UPPER(char)
- SUBSTR(char,m[,n])
- LENGTH(word)
- LTRIM(char[,set])
- RTRIM(char[,set])
- LPAD(char1,n [,char2])
- RPAD(char1,n [,char2])

Conversion Functions

- TO_NUMBER(char)
- TO_CHAR(n[,fmt])

Date Conversion Functions

- TO_DATE (char [, fmt])

DATE Functions

- ADD_MONTHS(d,n)
- LAST_DAY(d)
- MONTHS_BETWEEN(d1,d2)
- NEXT_DAY(date, char)

Set Operations

- UNION
- UNION ALL

- INTERSECTION
- MINUS

Assignment

- List the no of jobs available in the EMP table
- List the no of employees working with the company.
- List the dept no and number of employees in each dept.
- List the total salary, max and min salary and the average salary of employees' job wise for deptno 20 only.
- List the average salary for all department employing more than five people.
- List the max salary of employee working as salesman.
- Display the name in the lowercase, job in the initial capitalization for all managers.
- Display the value 45.923 rounded to the hundredth number and ten decimal places.
- For employees in dept 30, display the ename and number of weeks/months employed.
- Display the head of the company, who has no manager, display that there is no manager for that name.
- Display the job and total salary for each job category with a total salary for each job category 3000 and sort list by total salary.
- Display the number of managers without listing them.
- Write a query to display the minimum and maximum salary for each job type ordered alphabetically.

5. Lab Exercises:

Exercise No 5: (2 Hours) – 1 Practical

Writing sub queries on join operation

The SQL join operation combines information from two tables by forming pairs of related rows from the two tables. The row pairs that make up the joined table are those where the matching columns in each of the two tables have the same value.

If we need to obtain information from more than one table the choice is between using a sub query and using a join.

Assignment

- List the names of all clients who have viewed a property along with any comment supplied.
- For each branch office list the numbers and names of staff who manage properties and the properties that they manage.
- For each branch list the numbers and names of staff who manage properties including the city in which the branch is located and the properties that the staff manage.
- Find the no of properties handled by each staff member.
- List the branch offices and properties that are in the same city along with any unmatched branches.
- List the branch offices and properties that are in the same city along with any unmatched properties.
- List the branch offices and properties that are in the same city along with any unmatched branches or properties.
- Find all staff who work in London branch office.

6. Lab Exercises:

Exercise No 6: (2 Hours) – 1 Study Practical

Creating views:

After a table is created and populated with data, it may become necessary to prevent all users from accessing all columns of a table for data security reasons. This would mean creating several tables having the appropriate number of columns and assigning specific user to each table as required. This will answer data security requirements very well but will give rise to a great deal of redundant data being resident in tables in the database.

To reduce redundant data to the minimum possible, oracle allows the creation of an object called a view. A view is mapped to a select sentence. The table on which the view is based is described in the FROM clause of the SELECT statement.

Assignment:

1. Use SQL'S CREATE TABLE statement to create the appropriate tables having the table definitions mentioned below.

Table Name: Material_Mstr

Column Definition:

| Column Name | Data Type | Width | Allow Null | Default |
|-------------|-----------|-------|------------|---------|
| RmId | Number | 5 | No | |
| RmName | VarChar2 | 20 | No | |
| ReOrdLv1 | Number | 5 | No | |

Table Name: Material_Dtls

Column Definition:

| Column Name | Data Type | Width | Allow Null | Default |
|-------------|-----------|-------|------------|---------|
| RmId | Number | 5 | No | |
| SuplrId | Number | 5 | No | |
| PurDt | Date | -- | No | |

2. Using the SQL's ALTER TABLE statement, modify the table structure such as to add the relationship constraints as described below.

Table Name : Material_Mstr

Primary Key : Material_Mstr. RmId

Table Name : Material_Dtls

Foreign Key : Material_Mstr. RmId, Supplier_Mstr. SuplrId

3. Using ANSI SQL's INSERT statement, insert the following set of information into the tables created earlier (Material_Mstr and Material_Dtls).

| Table Name:Material_Mstr | | |
|--------------------------|--------------|----------|
| RmId | RmName | ReodrLv1 |
| 1 | Monitor | 4 |
| 2 | 32 MB RAM | 10 |
| 3 | PC Cabinet | 10 |
| 4 | Mother Board | 3 |
| 5 | Floppy Drive | 15 |
| 6 | CD Writer | 3 |

| | | |
|----|--------------|----|
| 7 | CD Drive | 5 |
| 8 | Key Board | 10 |
| 9 | Power Supply | 8 |
| 10 | Mouse | 10 |

| Table Name:Material_Dtls | | | | |
|--------------------------|---------|------------|--------|--------|
| RmId | SuplrId | PurDt | QtyPur | UtCost |
| 1 | 2 | 01/02/2003 | 10 | 5550 |
| 2 | 2 | 01/02/2003 | 20 | 1250 |
| 3 | 2 | 01/02/2003 | 20 | 500 |
| 4 | 5 | 24/02/2003 | 5 | 10000 |
| 5 | 3 | 25/02/2003 | 25 | 300 |
| 6 | 3 | 25/02/2003 | 5 | 4250 |
| 7 | 3 | 25/02/2003 | 10 | 3000 |
| 8 | 1 | 28/02/2003 | 20 | 400 |
| 10 | 1 | 28/02/2003 | 20 | 500 |
| 9 | 4 | 01/03/2003 | 20 | 500 |
| 3 | 4 | 01/03/2003 | 15 | 450 |

4. Use the ANSI SQL's SELECT statement to perform the following operations.
 - a) View all records from the table Material_Mstr.
 - b) View all records from the table Material_Dtls.
 - c) View the quantities of each material purchased listed alphabetically.
 - d) View the names of material whose total quantities is greater than 10.
 - e) View the names of material which has the highest total quantity.
 - f) View the total quantities of various material purchased.
 - g) View just the names of material purchased without repeating any item.
5. Using ANSI SQL's UPDATE statements, to perform the following operations:
 - a) Replace the material named Power Supply with UPS,
 - b) Add 10% to the quantities purchased for material named Mouse,
 - c) Increase the quantities of material purchased by 5 for all purchases done after Feb 2003.
6. Using ANSI SQL's DELETE statement, to perform the following operation.
Delete records from Material_Dtls for material having RmId as 5.
7. Create an English sentence, by joining predetermined string values with column data retrieved from the Material_Mstr table.

The string literals are:

| | |
|--------------|----------------------------|
| The Material | Has the identity number as |
|--------------|----------------------------|

The columns are:

| | |
|--------|------|
| RmName | RmId |
|--------|------|

8. Create VIEW OBJECT containing the following columns:

Column Definition:

| Column Name | Data Type | Width | Allow Null | Default |
|-------------|-----------|-------|------------|---------|
| RmName | Varchar2 | 20 | No | |
| PurDt | Date | - | No | |
| QtyPur | Number | 5 | No | 0 |
| UtCost | Number | 7,2 | No | 0.00 |

The data required in this VIEW should be taken for Material_Mstr and Material_Dtls tables

7. Lab Exercises:

Exercise No 7: (2 Hours) – 1 Study Practical

Writing and Executing PL/SQL block:

- Cursor
- Triggers

Cursor

The RDBMS engine uses a work area that is a specific place in computer's memory for all its internal processing in order to execute SQL statement. This work area is private to SQL operations and is called a cursor.

The data that is present in the cursor is called the active data state. Conceptually the size of the cursor in memory must be the size required to hold all the rows of the active data state. However RDBMS engines built-in memory management capabilities and the amount of free memory available in the computers determines actual cursor size.

Assignment:

- An HRD has decided to raise the salary of employees working in department 30 by 0.15%. Write a PL/SQL block to update the same using implicit cursors.
- An HRD manager has decided to raise the salary of all employees working whose salary is less than Rs.5000/- by Rs.1000.
- Write a PL/SQL code block that does this using an Explicit Cursor and treats the updating process as a single transaction.
- Write a PL/SQL code block that performs the task mentioned in Exercise 2 while using the FOR... IN statement.
- The Emp table mentioned below will be used as a base table for creating the solutions for the above exercises.

Table Name: EMP

| EmpNo | EmpName | DeptNo | Salary |
|-------|---------|--------|--------|
| E0001 | Mamta | 30 | 6000 |
| E0003 | Hansel | 10 | 6500 |
| E0005 | Ashwini | 20 | 6000 |

Triggers

Information about triggers can be retrieved from two views, the all_triggers view and the all_trigger_cols view. Most of the time you will find all the information you need in all_triggers. The all_trigger_cols view contains a list of all database columns referenced in the trigger. This view is sometimes useful when you are troubleshooting because it can show you which triggers reference or modify any given database column.

To find out whether any triggers have been defined on table, query all_triggers as shown in ex .1 for 'Listing the names of triggers on a table' given below.

Ex.1

```
SET VERIFY OFF
COLUMN description FORMAT A40 WORD_WRAPPED
```

```

COLUMN status FORMAT A10
SELECT description, status
FROM all_triggers
WHERE table_owner =UPPER ('&owner')
AND table_name=UPPER ('&table_name');

```

Ex.2 shows a script that will describe a single trigger in detail.

The script's output is a CREATE TRIGGER statement that may be used to re-create the trigger. The FOLD_AFTER option is used in the column commands to force each column to begin a new line of output. SET PAGESIZE 0 GETS RID OF any page titles and column headings that would otherwise clutter the output. The trigger _body column is of type LONG, so SET LONG 5000 ensures that you will see at least the first 5000 bytes of a trigger body. Use a higher value if your triggers are longer than that.

Ex.2

```

SET VERIFY OFF
SET LONG 5000
SET PAGESIZE 0
COLUMN create_stmt FOLD_AFTER
COLUMN description FOLD_AFTER
COLUMN when_clause FOLD_AFTER

SELECT 'CREATE OR REPLACE TRIGGER'create_stmt
Description,
CASE WHEN when_clause IS NOT NULL THEN
'WHEN ('||when_clause||')'
ELSE
',
END when_clause,
Trigger_body
FROM all_triggers
WHERE owner =UPPER ('&trigger_name');
AND trigger_name=UPPER ('&trigger_name');

SET PAGESIZE 14

```

The following invocation of Ex.2 shows the definition for the trigger emp_delete_check.

```

SQL>@ex2-2
Enter value for owner:gennick
Enter value for trigger_name:emp_delete_check
CREATE OR REPLACE TRIGGER
Emp_delete_check
BEFORE DELETE on employee
FOR EACH ROW

BEGIN
IF(:OLD .employee_termination_date IS NULL)
OR (OLD.employee_termination_date >=TRUNC (SYSDATE) +1) THEN
RAISE_APPLICATION_ERROR (-20001,'you must terminate an employee before deleting
his record');
END IF;
END;

```

8. Lab Exercises:

Exercise No 8: (2 Hours) –1 Study Practical

Study of GRANT and REVOKE DBA commands

SQL provides the GRANT and REVOKE statements to allow security to be set up on the tables in the database. The security mechanism is based on the concepts of authorization, identifiers, ownership and privileges.

Assignment:

- Give the user with authorization identifier Manager full privileges to staff table.
- Give users Personal and Director privileges SELECT and UPDATE on column salary of the staff table.
- Give all users the privilege SELECT on the Branch table.
- Revoke the privilege SELECT on the Branch table from all users
- Revoke all privileges you have given to Director on the Staff table.

9. Lab Exercises:

Exercise No 9: (2 Hours) – 1 Study Practical

Normalization in oracle

It is a technique for producing a set of relations with desirable properties given the data requirements of an enterprise. Normalization is often performed as a series of tests on a relation to determine whether it satisfies or violates the requirements of a given normal form. Three normal forms are mostly used called First (1 NF), Second (2 NF) and Third (3 NF). Subsequently, R. Boyce introduced a stronger definition of third normal form called Boyce-Codd normal form(BCNF) .All these normal forms are based on functional dependencies among the attributes of a relation .Higher normal form Fourth(4 NF) and Fifth (5 NF) are rarely used.

Assignment

1. A client gives a set of printed Challans to an analyst or database designer. The printed invoices include information like:

| | | |
|-----------------|---------------|-----------------------|
| Challan number | Challan date | Purchase Order number |
| Supplier number | Supplier name | Supplier address |

As well as item details like:

| | |
|-------------------------|--------------------|
| Raw material number | Raw material name |
| Raw material unit Price | Quantity delivered |

As the analyst designer, create a mother matrix (single table) based on the above information.

2. Decompose the mother matrix generated in Exercise 1 into tables and eliminate all repeating groups of data.

10. Quiz on the subject:

Quiz should be conducted on tips in the laboratory, recent trends and knowledge of the subject. The quiz questions should be formulated such that questions are normally from the scope outside of the books. However twisted questions and self formulated questions by the faculty can be asked but correctness of it is necessarily to be thoroughly checked before the conduction of the quiz.

11. Conduction of Viva-Voce Examinations:

Teacher should conduct oral exams of the students with full preparation. Normally, the objective questions with guess are to be avoided. To make it meaningful, the questions should be such that depth of the students in the subject is tested. Oral examinations are to be conducted in co-cordial environment amongst the teachers taking the examination. Teachers taking such examinations should not have ill thoughts about each other and courtesies should be offered to each other in case of difference of opinion, which should be critically suppressed in front of the students.