VIDYALANKAR INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

Lab Manual

Subject : Database Management System

SEM-III

Prof. (Dr.) Deepali Vora

2019-2020

VIT
Vidyalankar Institute of Technology

Lab Manual

Subject	SQL Lab
Semester	III
Academic Year	2019-20
Software Requirements	Postgress 9, Netbeans IDE with JDK 8+
Hardware Requirements	P IV
Theory Faculty In-charge	Dr. Deepali Vora
Practical Faculty In-charge	Dr. Deepali Vora, Prof. Vidya Chitre, Prof. Rohit Brave
Laboratory	L-05
Lab Assistant	Sonal Gawade
Revised On	July 2019
Prepared By	Dr. Deepali Vora
Sign	
Endorsed By HOD	Dr. G. Bhole

Vidyalankar Institute Of Technology

DEPARTMENT OF INFORMATION TECHNOLOGY

LAB CODE

1. Students should report to the concerned labs as per the time table schedule.
2. Students who turn up late to the labs will in no case be permitted to perform the experiment scheduled for the day.
3. After completion of the experiment, certification of the concerned staff in-charge in the observation book is necessary.
4. Students should bring a note book of about 100 pages and should enter the readings/observations into the note book while performing the experiment.
5. The record of observations along with the detailed experimental procedure of the experiment performed in the immediate last session should be submitted and certified by the faculty member.
6. The group-wise division made in the beginning should be adhered to, and no mix up of student among different groups will be permitted later.
7. The components required pertaining to the experiment should be collected from the concerned Lab Assistants.
8. When the experiment is completed, students should disconnect the setup made by them, and should return all the components/instruments taken for the purpose. Any damage of the equipment or burn-out of components will be viewed seriously either by putting penalty or by dismissing the total group of students from the lab for the semester/year.
9. Students should be present in the labs for the total scheduled duration. Students are required to prepare thoroughly to perform the experiment coming to Laboratory. Procedure sheets/data sheets provided to the students groups should be maintained neatly and to be returned after the experiment.

Vision

To be recognized as a center of excellence in the field of Information Technology where learners are nurtured in a scholarly environment to evolve into competent professionals to benefit society.

Mission

- Evolve a curriculum which emphasizes on strong engineering fundamentals with the flexibility to choose advanced courses of interest and gain exposure to tools and techniques in Information Technology.
- Encourage a teaching-learning process in which highly competent faculty share a symbiotic association with the institutes of repute.
- Facilitate creation and dissemination of knowledge through a digitally-enabled learning environment.
- Develop academic and infrastructural facilities with modern equipment and other learning resources and encourage reciprocal sharing with other institutes through networking.
- Establish a center of excellence to enhance academia – industry partnership and work on collaborative projects.

Program Outcomes

1. Ability to apply knowledge of mathematics, science, Information technology principles and Standards.
2. Ability to identify problem and define the computing requirements appropriate to its solution
3. Ability to develop a software system or components or process to meet desired needs using feedback and effective communication techniques
4. Ability to conduct investigation of complex problems in IT domain that cannot be solved by routine methods, knowledge and may not have unique solution.
5. Ability to analyze user requirement by developing prototype or concept of proof using tools.
6. Ability to understand the impact of engineering solution on society and environment.
7. Ability to effectively provide IT based solutions into the user environment
8. Ability to understand professional, ethical, legal, security and social issues.
9. Ability to effectively lead and function as a member of a multidisciplinary team.
10. Ability to communicate effectively with a range of audiences using a range of modalities including written , oral and graphical
11. Ability to understand and apply the principles of management and finance in relation to IT engineering projects.
12. Ability to adapt to emerging information technology for life-long learning.

Lab Outcome

Sr. No.	Lab Outcomes
1	Construct problem definition statements for real life applications and implement a database for the same.
2	Design conceptual models of a database using ER modeling for real life applications and also construct queries in Relational Algebra.
3	Create and populate a RDBMS, using SQL.
4	Write queries in SQL to retrieve any type of information from a data base.
5	Analyse and apply concepts of normalization to design an optimal database.
6	Implement indexes for a database using techniques like B or B+ trees.

List of Experiment

Sr. No.	Name of Experiment	LO
1	Construction of detailed problem definitions for real life applications.	LO1
2	Construction of ER/EER diagrams for the identified application.	LO2
3	Construct relational model from ER diagram	LO2
4	To study and practice DDL commands and implement integrity constraints	LO3
5	To study and implement DML SQL queries on single and multiple relations	LO4
6	To study and implement aggregate functions and Date and Time functions in SQL	LO4
7	To design Complex nested Queries for update/delete data	LO4
8	To design views and illustrate joins	LO4
9	To design triggers	LO4
10	Program for JDBC database connectivity	LO4
11	Assignment for conversion of relation to different normal forms.	LO5
12	Program for Construction of B+ trees	LO6
13	Mini Project	LO1,LO2, LO3, LO4
14	To design Cursors using PL/SQL	LO4

Experiment 9 and 14 are optional for academic year 2019-20.

Experiment 1

Title: To collect the initial requirement for the design of Database Management System

---

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: Understand what are database design requirements and how to collect it.

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Students should be able to understand importance of DBMS and how to collect requirements for DBMS.

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: What is data and computer based system

---

Historical Profile:

New Concepts to be learned: Database Management system usage

Requirements:

Software & Hardware Required: PC with internet connection

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

DBMS and its Applications

DBMS:

A database management system is a software system. It allows access to the data in a database. It consists of a set of interrelated data together with a set of programs to access those data.

Objective:

The objective of a DBMS is to provide a convenient and effective method of defining, storing and retrieving the data in the database.

Purpose of DBMS:

Before the arrival of DBMS, data were processed using file processing system. File processing system consists of several application programs and each application program has its own data stored in private files. In this system the same data file cannot be shared. Hence duplication of data is required if two or more application programs have to share the same data.

Disadvantage of file processing system:

The file processing system has the following major disadvantages:

- Data redundancy and inconsistency.
- Integrity Problems.
- Security Problems
- Difficulty in accessing data.
- Data isolation.

a) Data redundancy and inconsistency:

Data redundancy means duplication of data and inconsistency means that the duplicated values are different.

b) Integrity problems:

Data integrity means that the data values in the data base should be accurate in the sense that the value must satisfy some rules.

c) Security Problem:

Data security means prevention of data accession by unauthorized users.

d) Difficulty in accessing data:

Difficulty in accessing data arises whenever there is no application program for a specific task.

e) Data isolation:

This problem arises due to the scattering of data in various files with various formats. Due to the above disadvantages of the earlier data processing system, the necessity for an effective data processing system arises. Only at that time the concept of DBMS emerges for the rescue of a large number of organizations.

Data base system applications:

Universities: For student information, course details, and grades.

Airlines: For reservations and schedule information.

Credit card transactions: For purchase on credit cards and generations of monthly statements.

Human resources: For information about employees, salaries, payroll taxes, benefits and for generations of paychecks.

Banking: For customer information, accounts, and banking transactions.

Sample Problem Definition:

It is very important to maintain efficient software to handle information of an event management system. This application provides away to record this information and to access these in a simple way.
The existing systems provide the basic functionalities needed to be handled in an event management environment. A new system is to be designed which keeps into account all the events which are going to take place, the locations, number of participants, date of the event, etc. it will keep the details of the participants, such as name, address, contact details. Apart from participants, crew details will also be stored. The sponsors for the events, their details are also maintained by the system.

This system will keep all the details required for event management in an organized manner.

Functions of the event management system:
1. Keep record of all events and their sponsors
2. Keep record of all participants and their respective events
3. Store details of the crew members who will help in events.

Required data:
Events (event_code, name, place, no of participants, no of crew, total earnings, total cost)
Participants (part_Id, FN, LN, Email, DOB, Address)
Crew (crew_Id, FN, LN, Email, DOB, Address, department, salary)
Sponsors (spon_Id, name, owner, address, amount_sponsored, no of events sponsoring)

Conclusion: Thus the database requirements for the computer based system are collected

---

Real Life Application: To collect requirements for any computer based system or application.

Viva Questions:

1. What is DBMS ?
2. What are objectives of DBMS ?
3. What is goal of dbms ?
4. List disadvantages of file system ?
5. What are different storage models ?
6. What is redundancy ?
7. What is integrity constraint ?
8. What is Atomicity ?
9. How is security in dbms?
10. What is concuurency?
11. List applications of DBMS ?

Post Lab Questions:

1. What is a record in DBMS?
2. Different types of databases.
3. What are the applications of a Database?

4. What is the concept of primary key?

Experiment 2

Title: To design the model to depict the database requirement.

Estimated time to complete this experiment: 2 hours

Objective: In this experiment, student should understand to draw ER diagram from the requirements collected in previous experiment.

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Students should be able to understand the concept of ER Modeling

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: ER Notations

Historical Profile:

New Concepts to be learned: E R Modelling

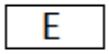
Requirements:
Software & Hardware Required: PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

Notations used in an ER Diagram

1. Entity

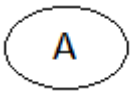


2. Relationship

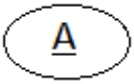
3. Link

---

4. Attribute



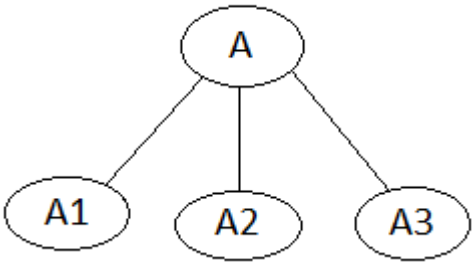
5. Primary Key



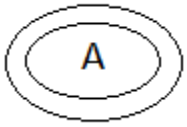
6. Derived Attribute

A

7. Composite Attribute



8. Multi-valued Attribute



9. One to Many

---

10. One to One

---

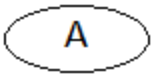
11. Many to Many

---

12. Weak Entity Set

13. Weak Relationship Set

14. Discriminator



Conclusion: Thus the ER diagram is created for computer based system

---

Real Life Application: In designing database for any computer application

---

Viva Questions:

1. What is ER diagram?
2. What is problem Definition?
3. Draw an ER diagram for any given applications.
4. What are the different notations for ER diagram?
5. What is entity?
6. Define attributes.
7. What is cardinality?
8. How to define weak relationship?
9. What is modality?
10. Define discriminator.

Post Lab Questions:

1. What is a weak entity? Give an example
2. How to identify entities from a problem definition?
3. What are the notations of weak entity and multivalued attribute?
4. How to depict cardinalities?

Experiment 3

Title: To design the database schema

Estimated time to complete this experiment: (in hours) -- 2hrs

Objective: To create relational model from ER model.

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Students will be able to create relational schema from ER diagram.

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: Relational database concepts.

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiments.)

New Concepts to be learned: Relational Schema

Requirements:
Software & Hardware Required:

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

Mapping of ER/ERR model to Relational model:

Step 1.Strong Entity Set
It reduces to Relational schema with same attributes.

Step 2.Weak Entity Set
It is mapped as a separate schema with combination of primary key from identifying strong entity set and attributes of weak entity set.
There is a foreign key constraint that each row in weak entity set there is a corresponding strong entity set row.

Step 3.Realtionships
1.One to One : Extra attribute is added to either of the two entity set schema corresponding to entity set participating in the relationship.

2.Many to One/One to Many : Addition of extra attribute to the many side of entity set schema with primary key of the one side entity set as a foreign key constraint.

3.Many to Many : The relationship is mapped as a separate schema with the primary keys of participating schema.

Step 4.Multivalued attribute
A multivalued attribute is mapped as a separate schema with attributes as primary key of Entity set and the multivalued attribute.

Step 5:Composite attribute
A composite attribute is flattened out by creating a separate attribute for each component attribute.

Program code and Output:

Sample Schema:

Events (event_code, name, place, no of participants, no of crew, total earnings, total cost)
Participants (part_Id, FN, LN, Email, DOB, Address)
Crew (crew_Id, FN, LN, Email, DOB, Address, department, salary)
Sponsors (spon_Id, name, owner, address, amount_sponsored, no of events sponsoring)
Part_cont (part_Id, contact)
Crew_cont (crew_Id, contact)
Spon_cont (spon_Id, contact)
Participation (part_Id, event_code)
Sponsorship (event_code, spon_Id)

Conclusion: Thus we have converted ER diagram into relational schema

---

Real Life Application: In designing database for any computer application.

Viva Questions:

1. What is relational schema?
2. How to convert ER diagram into relational schema.
3. What are the steps from conversion?
4. What is strong entity?
5. Define weak entity.
6. What is composite attributes?
7. What is multivalued attributes?
8. Differentiate between multivalued and composite attributes.
9. How to convert a weak entity into relational schema?
10. What are the rules for conversion of specialization or generalization?

Post Lab Questions:

1. What is a weak entity? Give an example
2. Differentiate between multivalued and composite attributes.
3. What are the steps of conversion of ER diagram to schema?
4. How to depict foreign key in schema?
5. What are the rules for conversion of specialization or generalization?

Experiment 4

Title: To study and practice DDL commands and implement integrity constraints.

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: To create relational database in Oracle using SQL.

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Students should be able to create and alter table structure

---

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: SQL DDL commands.

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiments.)

New Concepts to be learned: Data Definiton Language Commands.

Requirements:

Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

DDL Commands:

1.CREATE TABLE

The CREATE TABLE statement is used to create a table in a database.

SQL CREATE TABLE Syntax :

```
CREATE TABLE table_name
(
column_name1 data_type,
column_name2 data_type,
column_name3 data_type,
....
)
```

2. ALTER TABLE

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.

SQL ALTER TABLE Syntax

To add a column in a table, use the following syntax:

```
ALTER TABLE table_name
ADD column_name datatype
```

To delete a column in a table, use the following syntax (notice that some database systems don't allow deleting a column):

```
ALTER TABLE table_name
DROP COLUMN column_name
```

3. DROP TABLE

The DROP TABLE statement is used to delete a table.

SQL DROP TABLE Syntax :

DROP TABLE table_name

4. TRUNCATE TABLE

What if we only want to delete the data inside the table, and not the table itself?

Then, use the TRUNCATE TABLE statement:

SQL TRUNCATE TABLE Syntax :

TRUNCATE TABLE table_name

5. INSERT INTO Statement :

The INSERT INTO statement is used to insert a new row in a table.

SQL INSERT INTO Syntax :

It is possible to write the INSERT INTO statement in two forms.

The first form doesn't specify the column names where the data will be inserted, only their values:

INSERT INTO table_name
VALUES (value1, value2, value3,...)

The second form specifies both the column names and the values to be inserted:

INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)

Program Code and Output:

Query:

```
CREATE TABLE Events08
( eventcode int primary key,
name varchar(20) not null,
place varchar(10) not null,
partno int,   crewno int,
earn int,
cost int );
```

Table created

CREATE TABLE Participants08

```
( partId int primary key,
FN varchar(10),
LN varchar(10),
email varchar(25),
DOB date,
Address varchar(40) );
```

Table created

```
CREATE TABLE Crew08
( crewId int primary key,
FN varchar(10),
LN varchar(10),
email varchar(25),
DOB date,
Address varchar(40),
dept varchar(20) not null,
salary float );
```

Table created

```
INSERT INTO Events08 VALUES (01, 'Dance', 'Mumbai', 17, 22, 3500, 2000);
INSERT INTO Events08 VALUES (02, 'Music', 'Mumbai', 14, 13, 1500, 1000);
INSERT INTO Events08 VALUES (03, 'Singing', 'Mumbai', 15, 20, 3600, 2300);
INSERT INTO Events08 VALUES (04, 'Coding', 'Mumbai', 22, 03, 2500, 500);
INSERT INTO Events08 VALUES (05, 'Concert', 'Mumbai', 537, 96, 33000, 12200);
```

5 rows inserted successfully

EVENTCODE	NAME	PLACE	PARTNO	CREWNO	EARN	COST
1	Dance	Mumbai	17	22	3500	2000
2	Music	Mumbai	14	13	1500	1000
3	Singing	Mumbai	15	20	3600	2300
4	Coding	Mumbai	22	3	2500	500
5	Concert	Mumbai	537	96	33000	12200

```
INSERT INTO Participants08 VALUES (01, 'Pratik', 'Salve', 'pratik@vit.in', to_date('14-03-1995','dd-mm-yyyy'), 'Thane' );
INSERT INTO Participants08 VALUES (02, 'Shashank', 'Pawar', 'shashank@vit.in', to_date('05-06-1995','dd-mm-yyyy'), 'Thane' );
INSERT INTO Participants08 VALUES (03, 'Abhishek', 'Patil', 'abhi32@vit.in', to_date('04-05-1993','dd-mm-yyyy'), 'Mulund' );
INSERT INTO Participants08 VALUES (04, 'Pritesh', 'Divekar', 'pritesh22@vit.in', to_date('22-10-1995','dd-mm-yyyy'), 'Mumbai' );
```

INSERT INTO Participants08 VALUES (05, 'Shraddha', 'Mhetre', 'shraddha@vit.in', to_date('03-12-1995','dd-mm-yyyy'), 'Kalwa');

5 rows inserted successfully

PARTID	FN	LN	EMAIL	DOB	ADDRESS
1	Pratik	Salve	pratik@vit.in	14-MAR-95	Thane
2	Shashank	Pawar	shashank@vit.in	05-JUN-95	Thane
3	Abhishek	Patil	abhi32@vit.in	04-MAY-93	Mulund
4	Pritesh	Divekar	pritesh22@vit.in	22-OCT-95	Mumbai
5	Shraddha	Mhetre	shraddha@vit.in	03-DEC-95	Kalwa

INSERT INTO Crew08 VALUES (01, 'Yash', 'Chopra', 'yashc@vit.in', to_date('09-05-1985','dd-mm-yyyy'), 'Thane', 'Publicity', 5000.00);
INSERT INTO Crew08 VALUES (02, 'Aditya', 'Chopra', 'adichop@vit.in', to_date('12-06-1988','dd-mm-yyyy'), 'Wadala', 'Publicity', 5800.00);
INSERT INTO Crew08 VALUES (03, 'Priyanka', 'Chopra', 'peecee@vit.in', to_date('22-08-1982','dd-mm-yyyy'), 'Mumbai', 'Designing', 7000.00);
INSERT INTO Crew08 VALUES (04, 'Parineeti', 'Chopra', 'pari@vit.in', to_date('22-10-1985','dd-mm-yyyy'), 'Kurla', 'Sponsorship', 9000.00);
INSERT INTO Crew08 VALUES (05, 'Prem', 'Chopra', 'prem@vit.in', to_date('27-01-1980','dd-mm-yyyy'), 'Thane', 'Publicity', 4580.00);

5 Rows inserted successfully.

CREWID	FN	LN	EMAIL	DOB	ADDRESS	DEPT	SALARY
1	Yash	Chopra	yashc@vit.in	09-MAY-85	Thane	Publicity	5000
2	Aditya	Chopra	adichop@vit.in	12-JUN-88	Wadala	Publicity	5800
3	Priyanka	Chopra	peecee@vit.in	22-AUG-82	Mumbai	Designing	7000
4	Parineeti	Chopra	pari@vit.in	22-OCT-85	Kurla	Sponsorship	9000
5	Prem	Chopra	prem@vit.in	27-JAN-80	Thane	Publicity	4580

Conclusion: Thus we have implemented the logical design with integrity constraints.

Real Life Application: For creation of table structure in databases for any computer application

Viva Questions:

1. What are objectives of DBMS ?
2. What is syntax of CREATE TABLE command ?
3. What is syntax of ALTER TABLE command ?
4. What are DML commands uused for ?
5. What is syntax of DROP command ?
6. What is syntax of TRUNCATE TABLE command ?
7. What is difference between DROP and TRUNCATE command ?
8. How do u add a column to an existing table?
9. How to modify datatype of a column?
10. How to drop a column from existing table ?

Post Lab Questions:

1. List the DDL commands
2. Syntax of CREATE table
3. What is the difference between TRUNCATE and DROP?
4. Syntax of Truncate table.
5. Need of ALTER table command with syntax.

Experiment 5

Title: To study and implement DML SQL queries on single and multiple relations

Estimated time to complete this experiment: (in hours) -- 2hrs

Objective: To insert and manipulate data in the table

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Adding and manipulating data in the database

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: SQL commands .

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiment.)

New Concepts to be learned: Data Manipulation Language Commands.

Requirements:

Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

1.SELECT Statement

The SELECT statement is used to select data from a database.

The result is stored in a result table, called the result-set.

SQL SELECT Syntax

```
SELECT column_name(s)
FROM table_name
WHERE condition1 AND/OR condition2
ORDER BY attr1 asc/desc
```

```
SELECT * FROM table_name
```

Tip: The asterisk (*) is a quick way of selecting all columns!

- DISTINCT keyword :

It is used in SELECT clause when we do not wish to display the repeat rows.

- BETWEEN AND

It is used in WHERE clause to specify the range of values.

```
SELECT  *

FROM tablename

ORDER BY column_name asc/desc
```

It is used to display the rows in ascending or descending order.

Program Code and output:

Q.1 Find the names of events who has more than 20 participants.

Ans.
```
Select name
From Events08
Where partno > 20
```

NAME
Coding
Concert

Q.2 Find the number of Crew members assigned to event name 'Concert'.

Ans.
```
Select Crewno
From Events08
Where name = 'Concert'
```

CREWNO
96

Q.3 List the phone numbers of all the participants.

Ans.
```
Select contact
From Part_cont08
```

CONTACT
9987654328
9987654961
9987823693
9757729428
9987658462

Q.4 Find the name of crew whose salary is more than 6000.

Ans.
```
Select FN, LN
From Crew08
Where Salary > 6000
```

FN	LN
Priyanka	Chopra
Parineeti	Chopra

Q.5 List the name of events whose total earning is less than 3000.

Ans.
```
Select name
From Events08
Where earn < 3000
```

NAME
Music
Coding

Q.6 Find name of crew whose department is Publicity

Ans.
```
Select LN, FN
From Crew08
Where dept = 'Publicity'
```

FN	LN
Yash	Chopra
Aditya	Chopra
Prem	Chopra

Q.7 List the Sponsors who sponsored event 'Dance'.

Ans.
```
Select  Sponsors08.name, owner
From    Sponsors08, Events08, Sponsoring08
Where   Sponsors08.sponid = Sponsoring08.sponid And
        Events08.eventcode = Sponsoring08.eventcode And
        Events08.name = 'Dance'
```

NAME	OWNER
McDonalds	Varun
Dominos	Shraddha
Nike	Barney

Q.8 Display the name of owner of sponsorship whose name start with 'V'.

Ans.
```
Select owner
From Sponsors08
Where owner like 'V%'
```

OWNER
Varun
Vikram

Q.9 Display contacts of all the sponsors sponsoring more than Rs. 4000.

Ans.
```
Select name, owner, contact
From   Sponsors08, Spon_cont08
Where        Sponsors08.sponid = Spon_cont08.sponid And
       Amnt > 4000
```

NAME	OWNER	CONTACT
Pepsi	Rahul	9692986645
Nike	Barney	7321831476

Q.10 List the participants who participated in event 'Coding'.

Ans.
```
Select FN, LN
From   Participation08, Participants08, Events08
Where        Events08.eventcode = Participation08.eventcode And
       Participation.partid = Participants08.partid And
       Name = 'Coding'
```

FN	LN
Pritesh	Divekar

Conclusion: Thus we have designed and implemented queries on single and multiple relations.

Real Life Application: While writing application program which uses data from database.

Viva Questions:

1.What is use of DML commands?
2.What is Syntax and use of SELECT Statement ?
3.What is Syntax and use of INSERT Statement ?
4.What is Syntax and use of DELETE Statement ?
5.What is Syntax and use of UPDATE Statement ?
6.What is the use of Where clause in DML statements?
7.What are different wildcard characters used with string Operations ?
8.What are the logical operators?
9.Explain in and not in operators?
10.Explain use of between and not between operators ?

Post Lab Questions:

1. List the DML commands
2. Syntax of INSERT INTO
3. What is the difference between ALTER and UPDATE query
4. Explain use of between and not between operators?
5. What is the use of Where clause in DML statements?

Experiment 6

Title: To study and implement aggregate functions and Date and Time functions in SQL

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: To manipulate data in the table

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Adding and manipulating data in the database

---

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: SQL commands .

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiment.)

New Concepts to be learned: Data Manipulation Language Commands.

Requirements:
Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

DML Commands :

1. **Aggregate Functions:**

There are 5 aggregate functions as:
Count – to count number of rows in table
Sum – to sum up the values in the table column
Avg – To calculate average of values in column
Min – To calculate minimum value
Max -- To calculate maximum value

Generalized Syntax
SELECT aggregate function
FROM table1, Table 2..
WHERE [Condition]
GROUP BY [attribute name]
HAVING [condidtion]

2. **Subquery**

SQL provides a mechanism for the nesting of subqueries. A subquery is a select-from-where expression that is nested within another query. A common use of subqueries is to perform tests for set membership, set comparisons, and set cardinality.

Three ways to construct subqueries:

- Set membership
    - IN and NOT IN
- Set Comparison
    - SOME
    - ALL
- Tests for empty relations
    - EXISTS
    - NOT EXISTS

Example Queries:

```
select distinct customer_name
        from borrower
        where customer_name in (select customer_name
                                    from depositor )


select branch_name
        from branch
        where assets > some
                (select assets
                 from branch
                 where branch_city = 'Brooklyn')


select customer_name
from borrower
where  exists (
                select *
                   from depositor
                where  depositor.customer_name = borrower.customer_name
                 )
```

Program Code and output:

Q.1 Find the number of events whose name start with 'C'.

Ans.
```
Select Count (name)
From Events08
Group by name
Having name like 'C%'
```

COUNT(NAME)
1
1

Q.2 Find the total sponsorship amount.

Ans.
```
Select count (amnt)
From Sponsorship08
```

SUM(AMNT)
46054.74

Q.3 Find the total earning for all events

Ans.
```
Select sum(earn)
From Events08
```

SUM(EARN)
44100

Q.4 Find the name of the events whose earning is above average earning of all events

Ans.
```
Select name
From Events08
Group by name
Having earn > ( Select Avg (earn)
                From Events08 )
```

NAME
Concert

Q.5 Find the number of Crew members whose salary is less than average salary.

Ans.
```
Select count (crewid)
From Crew08
Where salary < ( Select Avg (salary)
                 From Crew08)
```

COUNT(CREWID)
3

Q.6 Find the participants who have participated in more than 1 events.

Ans.
```
Select FN, LN
From   Participants08
Where  PartId in (
       Select PartId
       From Participation08
       Group by partid
       Having count (*) > 1 )
```

FN	LN
Pratik	Salve
Shashank	Pawar

Q.7 Find the name of sponsors who are sponsoring more than 3 event.

Ans.
```
Select  Name, Owner
From    Sponsors08
Where   SponId in (
        Select SponId
        From Sponsoring08
        Group by SponId
        Having count (*) > 2 )
```

NAME	OWNER
McDonalds	Varun
Sahara	Vikram
Nike	Barney

Q.8 Find the name, email & DOB of crew having max salary.

Ans.
```
Select FN, LN, Email, DOB
From Crew08
Where Salary = (Select Max(Salary)
                From Crew08 )
```

FN	LN	EMAIL	DOB
Parineeti	Chopra	pari@vit.in	22-OCT-85

Q.9 Find the name & address of sponsor sponsoring minimum amount.

Ans.
```
Select Name, Owner, Address
From Sponsors08
Where Amnt = (Select Min(Amnt)
              From Sponsors08)
```

NAME	OWNER	ADDRESS
McDonalds	Varun	Main office, Delhi

Q.10 Find the number of Crews whose Birthday is in October.

Ans.
```
Select count (CrewId)
From Crew08
Where to_char(DOB,'mm')='05'
```

COUNT(CREWID)
1

Conclusion: Thus we have designed and implemented complex queries to retrieve data.

Real Life Application: While writing application program which uses data from database.

Viva Questions:

1. What is use of DML commands?
2. Which are aggregate functions?
3. Explain GROUP BY clause.
4. Explain HAVING clause.
5. What is difference between Having and Where clause?
6. Can we use attribute names in Select clause with aggregate functions?
7. Explain natural join operation with example.
8. Explain outer joins with example.

Post Lab Questions:

1. List the types of aggregate functions.
2. Need of GROUP BY clause
3. Need of HAVING clause
4. Types of joins?
5. What is difference between Having and Where clause?

Experiment 7

Title: To design Complex and nested Queries for update/delete data

---

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: To manipulate data in the table

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Adding and manipulating data in the database

---

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: SQL commands.

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiment.)

New Concepts to be learned: Data Manipulation Language Commands.

Requirements:

Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

DML Commands:

1. UPDATE Statement :

The UPDATE statement is used to update existing records in a table.

SQL UPDATE Syntax :

UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value

Note: Notice the WHERE clause in the UPDATE syntax. The WHERE clause specifies which record or records that should be updated. If you omit the WHERE clause, all records will be updated!

2. DELETE Statement :

The DELETE statement is used to delete rows in a table.

SQL DELETE Syntax

DELETE FROM table_name
WHERE some_column=some_value

Delete All Rows

It is possible to delete all rows in a table without deleting the table. This means that the table structure, attributes, and indexes will be intact:

DELETE FROM table_name
or
DELETE * FROM table_name

Program Code and output:
Queries –

Q.1 Insert a sponsor with sponId 500 with name 'Café Coffee Day', owner Rajesh, having address Bandra (W), amount 1200 and having 2 no of events.

Ans. Insert into Sponsors08 values (500, 'Cafe Coffee Day', 'Rajesh', 'Bandra (W)', 1200, 2);

1 Row Inserted

SPONID	NAME	OWNER	ADDRESS	AMNT	NOEVENTS
501	Pepsi	Rahul	Main office, Mumbai-06	4363.64	2
502	McDonalds	Varun	Main office, Delhi	2345.43	4
503	Dominos	Shraddha	Mumbai-22	2345.67	1
504	Sahara	Vikram	Mumbai	3000	4
505	Nike	Barney	32 street, San Diego	34000	5
500	Cafe Coffee Day	Rajesh	Bandra (W)	1200	2

Q.2 Update the salary of Prem Chopra as 7000.

Ans. Update Crew08
Set salary = '7000'
Where FN = 'Prem'

CREWID	FN	LN	EMAIL	DOB	ADDRESS	DEPT	SALARY
5	Prem	Chopra	prem@vit.in	27-JAN-80	Thane	Publicity	7000

Q.3 Delete tuple from Sponsors whose ID is 500.

Ans Delete from Sponsors08
Where sponId = 500

SPONID	NAME	OWNER	ADDRESS	AMNT	NOEVENTS
501	Pepsi	Rahul	Main office, Mumbai-06	4363.64	2
502	McDonalds	Varun	Main office, Delhi	2345.43	4
503	Dominos	Shraddha	Mumbai-22	2345.67	1
504	Sahara	Vikram	Mumbai	3000	4
505	Nike	Barney	32 street, San Diego	34000	5

Q.4 Illustrate left outer join.

Ans. Select * from Participants08 left outer join Part_cont08
on Participants08.PatIdrtId=Part_cont08.PartId

PARTID	FN	LN	EMAIL	DOB	ADDRESS	PARTID	CONTACT
1	Pratik	Salve	pratik@vit.in	14-MAR-95	Thane	1	9987654328
2	Shashank	Pawar	shashank@vit.in	05-JUN-95	Thane	2	9987654961
3	Abhishek	Patil	abhi32@vit.in	04-MAY-93	Mulund	3	9987823693
4	Pritesh	Divekar	pritesh22@vit.in	22-OCT-95	Mumbai	4	9757729428
5	Shraddha	Mhetre	shraddha@vit.in	03-DEC-95	Kalwa	5	9987658462

Q.5 Delete tuple from Crew where first name is Uday.

Ans. Delete from Crew08
Where FN = 'Uday'

CREWID	FN	LN	EMAIL	DOB	ADDRESS	DEPT	SALARY
1	Yash	Chopra	yashc@vit.in	09-MAY-85	Thane	Publicity	5000

2	Aditya	Chopra	adichop@vit.in	12-JUN-88	Wadala	Publicity	5800
3	Priyanka	Chopra	peecee@vit.in	22-AUG-82	Mumbai	Designing	7000
4	Parineeti	Chopra	pari@vit.in	22-OCT-85	Kurla	Sponsorship	9000

Q.6 Replace Prem Chopra by Uday Chopra.

Ans. Update Crew08
Set FN = 'Uday'
Where FN = 'Prem'

CREWID	FN	LN	EMAIL	DOB	ADDRESS	DEPT	SALARY
1	Yash	Chopra	yashc@vit.in	09-MAY-85	Thane	Publicity	5000
2	Aditya	Chopra	adichop@vit.in	12-JUN-88	Wadala	Publicity	5800
3	Priyanka	Chopra	peecee@vit.in	22-AUG-82	Mumbai	Designing	7000
4	Parineeti	Chopra	pari@vit.in	22-OCT-85	Kurla	Sponsorship	9000
5	Uday	Chopra	prem@vit.in	27-JAN-80	Thane	Publicity	7000

Q.7 Update the Address of Shraddha.

Ans. Update Participants08
Set Address = 'Kurla'
Where FN = 'Shraddha'

PARTID	FN	LN	EMAIL	DOB	ADDRESS
1	Pratik	Salve	pratik@vit.in	14-MAR-95	Thane
2	Shashank	Pawar	shashank@vit.in	05-JUN-95	Thane
3	Abhishek	Patil	abhi32@vit.in	04-MAY-93	Mulund
4	Pritesh	Divekar	pritesh22@vit.in	22-OCT-95	Mumbai
5	Shraddha	Mhetre	shraddha@vit.in	03-DEC-95	Kurla

Q.8 Insert a participant with partId 6 with first name ‘Akshaya’, last name ‘Badhe’, email id as akshaya@vit.in, DOB as 23 Mar 1996 and address Bandra (W).

Ans. Insert into Participants08 values
(6, 'Akshaya', 'Badhe', 'akshaya@vit.in', to_date('23-03-1996','dd-mm-yyyy'), 'Bandra (W)');
1 Row inserted

PARTID	FN	LN	EMAIL	DOB	ADDRESS
1	Pratik	Salve	pratik@vit.in	14-MAR-95	Thane
2	Shashank	Pawar	shashank@vit.in	05-JUN-95	Thane
3	Abhishek	Patil	abhi32@vit.in	04-MAY-93	Mulund
4	Pritesh	Divekar	pritesh22@vit.in	22-OCT-95	Mumbai
5	Shraddha	Mhetre	shraddha@vit.in	03-DEC-95	Kurla
6	Akshaya	Badhe	akshaya@vit.in	23-MAR-96	Bandra (W)

Q.9 Illustrate right outer join.

Ans. Select * from Crew08 right outer join Crew_cont08
on Crew08.CrewId = Crew_cont08.CrewId

CREW ID	FN	LN	EMAIL	DOB	ADDRESS	DEPT	SALARY	CREW ID	CONTACT

1	Yash	Chopra	yashc@vit.in	09-MAY-85	Thane	Publicity	5000	1	9987666462
2	Aditya	Chopra	adichop@vit.in	12-JUN-88	Wadala	Publicity	5800	2	9697666454
3	Priyanka	Chopra	peecee@vit.in	22-AUG-82	Mumbai	Designing	7000	3	9933456423
4	Parineeti	Chopra	pari@vit.in	22-OCT-85	Kurla	Sponsorship	9000	4	9687896477
5	Prem	Chopra	prem@vit.in	27-JAN-80	Thane	Publicity	7000	5	9692986645

Q.10 Increase the salary of all crew by 10%.

Ans. Update Crew08
Set salary = 1.1*salary

CREWID	FN	LN	EMAIL	DOB	ADDRESS	DEPT	SALARY
1	Yash	Chopra	yashc@vit.in	09-MAY-85	Thane	Publicity	5500
2	Aditya	Chopra	adichop@vit.in	12-JUN-88	Wadala	Publicity	6380
3	Priyanka	Chopra	peecee@vit.in	22-AUG-82	Mumbai	Designing	7700
4	Parineeti	Chopra	pari@vit.in	22-OCT-85	Kurla	Sponsorship	9900

Conclusion: Thus we have designed and implemented the queries to change the data.

Real Life Application: While writing application program which uses data from database.

Viva Questions:

1. What is difference between SOME and ALL Operators
2. Which are the set operations?
3. Explain difference between union and union all
4. Explain the IN construct.
5. Explain EXISTS construct.

Post Lab Questions:

1. List the different ways to construct sub queries.
2. The use of IN operator
3. Different ways to use LIKE operator
4. Explain the use of EXISTS and NOT EXISTS

Experiment 8

Title: To design views and illustrate joins

Estimated time to complete this experiment: (in hours) -- 2hrs

Objective: To perform operations on database

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: To perform operations on database

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: Creating a Database and Populating data in tables and writing Queries

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiments.)

New Concepts to be learned: VIEWS and Triggers

Requirements:
Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

I. SQL CREATE VIEW Statement

A SQL View is a virtual table, which is based on SQL SELECT query. Essentially a view is very close to a real database table (it has columns and rows just like a regular table), except for the fact that the real tables store data, while the views don't. The view's data is generated dynamically when the view is referenced. A view references one or more existing database tables or other views. In effect every view is a filter of the table data referenced in it and this filter can restrict both the columns and the rows of the referenced tables. A view can be referenced and used from another view, from a SQL query, and from stored procedure. You reference a view as you would reference any real SQL database table.

You can add SQL functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from one single table.

SQL CREATE VIEW Syntax :

```
CREATE VIEW view_name AS
SELECT column_name(s)
FROM table_name
WHERE condition
```

Note: A view always shows up-to-date data! The database engine recreates the data, using the view's SQL statement, every time a user queries a view.

SQL Updating a View

You can update a view by using the following syntax:

SQL CREATE OR REPLACE VIEW Syntax

```
CREATE OR REPLACE VIEW view_name AS
SELECT column_name(s)
FROM table_name
WHERE condition
```

SQL Dropping a View

You can delete a view with the DROP VIEW command.

SQL DROP VIEW Syntax

DROP VIEW view_name

II. JOINS

Join operations take two relations and return as a result another relation.These additional operations are typically used as subquery expressions in the from clause

Join condition – defines which tuples in the two relations match, and what attributes are present in the result of the join.

Join type – defines how tuples in each relation that do not match any tuple in the other relation (based on the join condition) are treated

Join types
inner join
left outer join
right outer join
full outer join

Join Conditions
natural
on <predicate>
using $(A_1, A_1, \ldots, A_n)$

Program Code and output:

Queries –

Q.1 Define a view which shows Event and its location.

Ans. Create view Event_Location as
Select eventcode, name, place
From Events08

View Created

EVENTCODE	NAME	PLACE
1	Dance	Mumbai
2	Music	Mumbai
3	Singing	Mumbai
4	Coding	Mumbai
5	Concert	Mumbai

Q.2 Define a view which contains contact information of participants.

Ans. Create view Contact_Part as
Select FN, LN, Email, Contact
From Participants08 p, Part_cont08 c
Where p.partId = c.partId

View Created

FN	LN	EMAIL	CONTACT
Pratik	Salve	pratik@vit.in	9987654328
Shashank	Pawar	shashank@vit.in	9987654961
Abhishek	Patil	abhi32@vit.in	9987823693
Pritesh	Divekar	pritesh22@vit.in	9757729428
Shraddha	Mhetre	shraddha@vit.in	9987658462

Q.3 Define a view which contains details of Sponsons.

Ans. Create view Sponsor_detail as
Select Name, Owner, Amnt, Contact
From Sponsors08 s, Spon_cont08 c
Where s.sponId = c.sponId

NAME	OWNER	AMNT	CONTACT
Pepsi	Rahul	4363.64	9692986645
McDonalds	Varun	2345.43	7749927445
Dominos	Shraddha	2345.67	7682348923
Sahara	Vikram	3000	7692382382
Nike	Barney	34000	7321831476

Q.4 Illustrate left outer join.

Ans. Select * from Participants08 left outer join Part_cont08
on Participants08.PatIdrtId=Part_cont08.PartId

PARTID	FN	LN	EMAIL	DOB	ADDRESS	PARTID	CONTACT
1	Pratik	Salve	pratik@vit.in	14-MAR-95	Thane	1	9987654328
2	Shashank	Pawar	shashank@vit.in	05-JUN-95	Thane	2	9987654961
3	Abhishek	Patil	abhi32@vit.in	04-MAY-93	Mulund	3	9987823693
4	Pritesh	Divekar	pritesh22@vit.in	22-OCT-95	Mumbai	4	9757729428
5	Shraddha	Mhetre	shraddha@vit.in	03-DEC-95	Kalwa	5	9987658462

Q.5 Illustrate right outer join.

Ans. Select * from Crew08 right outer join Crew_cont08
on Crew08.CrewId = Crew_cont08.CrewId

CREWID	FN	LN	EMAIL	DOB	ADDRESS	DEPT	SALARY	CREWID	CONTACT
1	Yash	Chopra	yashc@vit.in	09-MAY-85	Thane	Publicity	5000	1	9987666462

2	Aditya	Chopra	adichop@vit.in	12-JUN-88	Wadala	Publicity	5800	2	9697666454
3	Priyanka	Chopra	peecee@vit.in	22-AUG-82	Mumbai	Designing	7000	3	9933456423
4	Parineeti	Chopra	pari@vit.in	22-OCT-85	Kurla	Sponsorship	9000	4	9687896477
5	Prem	Chopra	prem@vit.in	27-JAN-80	Thane	Publicity	7000	5	9692986645

Conclusion: Thus we have designed and implemented Views and joins.

Real Life Application:

Viva Questions:

Views :

1. What is the use of View ?
2. Views are a part of which level in the database ?
3. What is the Syntax of VIEW STATEMENT ?
4. List some applications where views are useful ?
5. Can DML operations performed on Views ?
6. What is the Syntax to DROP a VIEW?
7. What is the Synatx to ALTER a VIEW ?
8. Does the VIEW table reflect changes made in the original table?
9. Are view tables stored physically or created on fly ?
10. Views occupy less space ?Justify ?

Joins:

1. Explain natural join with example
2. Explain the use of outer join.
3. What is difference between natural join and inner join?
4. What is Cartesian product?

Post Lab Questions:

1. Can DML operations performed on Views ?

2. What is the use of View ?
3. Does the VIEW table reflect changes made in the original table?
4. Difference between inner join and natural join?
5. What is the use of joins?
6. Why joins are important?

Experiment 9

Title: To design triggers on database schema.

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: To perform operations on database

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: To perform operations on database

---

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: Creating a Database and Populating data in tables and writing Queries

---

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiments.)

New Concepts to be learned: Triggers

Requirements:

Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

I.TRIGGERS :

SQL triggers

The SQL CREATE TRIGGER statement provides a way for the database management system to actively control, monitor, and manage a group of tables whenever an insert, update, or delete operation is performed. The statements specified in the SQL trigger are executed each time an SQL insert, update, or delete operation is performed. An SQL trigger may call stored procedures or user-defined functions to perform additional processing when the trigger is executed.

Unlike stored procedures, an SQL trigger cannot be directly called from an application. Instead, an SQL trigger is invoked by the database management system on the execution of a triggering insert, update, or delete operation. The definition of the SQL trigger is stored in the database management system and is invoked by the database management system, when the SQL table, that the trigger is defined on, is modified. An SQL trigger can be created by specifying the CREATE TRIGGER SQL statement.

Parts of a Trigger

A trigger has three basic parts:

- A triggering event or statement
- A trigger restriction
- A trigger action

The Triggering Event or Statement

A triggering event or statement is the SQL statement, database event, or user event that causes a trigger to fire. A triggering event can be one or more of the following:

- An INSERT, UPDATE, or DELETE statement on a specific table (or view, in some cases)
- A CREATE, ALTER, or DROP statement on any schema object
- A database startup or instance shutdown
- A specific error message or any error message
- A user logon or logoff

Trigger Restriction

A trigger restriction specifies a Boolean expression that must be true for the trigger to fire. The trigger action is not run if the trigger restriction evaluates to false or unknown.

Trigger Action

A trigger action is the procedure (PL/SQL block, Java program, or C callout) that contains the SQL statements and code to be run when the following events occur:

- A triggering statement is issued.
- The trigger restriction evaluates to true.

Trigger Syntax:

```
CREATE OR REPLACE TRIGGER trigger_name
  BEFORE|AFTER|INSTEAD OF
  INSERT|DELETE|UPDATE ON table_name
        [OF column_names]
  [REFERENCING [NEW AS new_cols] [OLD AS old_cols]]
  [FOR EACH ROW [WHEN (where_condition)]]
  [DECLARE]
  BEGIN
  EXCEPTION
  END;
```

SQL DROP TRIGGER STATEMENT

DROP TRIGGER removes the specified trigger.

Syntax

DROP TRIGGER TriggerName

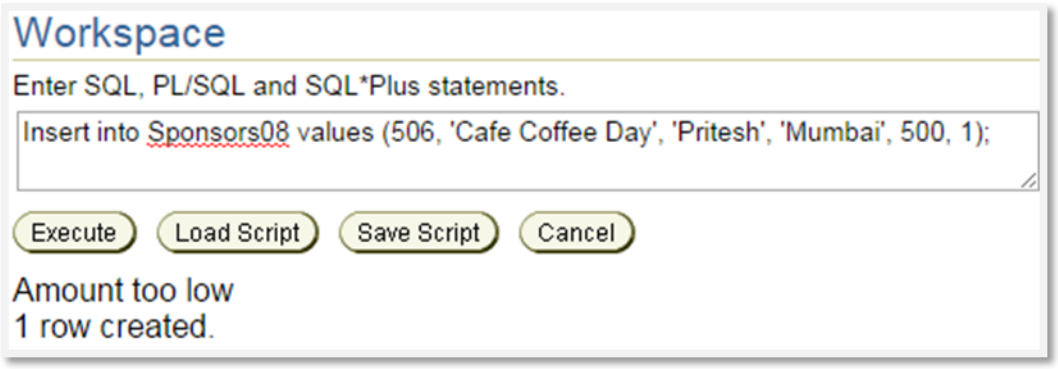
Statement Dependency System

When a table is dropped, all triggers on that table are automatically dropped. (You don't have to drop a table's triggers before dropping the table.)

Program Code and output:
Queries –

Q.1 Write a trigger that will ensure new amount is not less than 1000.

Ans.
```
Set Serveroutput on;
Create or replace trigger tiger08
Before insert on Sponsors08
for each row
When (New.Amnt < 1000)
begin
Dbms_Output.put_line ('Amount too low');
end;
Tigger Created
```



Conclusion: Thus we have designed and implemented Triggers.

Real Life Application:

Viva Questions:

Triggers :

1. What is a Trigger?
2. Explain the model used for trigger?
3. What is the Syntax for creating a trigger?
4. What is the Syntax to DROP a trigger?
5. What is the use of ":" in trigger body?
6. List few applications of triggers ?
7. What is an ASSERTION?
8. What is difference in assertion and trigger?
9. Write an example of trigger?
10. What is difference between row level and table level trigger?

Post Lab Questions:

1. What is an ASSERTION?
2. What is difference in assertion and trigger?
3. Write an example of trigger?
4. Trigger is static or dynamic constraint? Explain.

Experiment 10

Title: To use JDBC(Java Database Connectivity) to study connectivity with individual databases.

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: To perform operations on database and form the java connectivity.

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: To perform java connectivity with database

---

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: Creating a Database and Populating data in tables

---

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiments.)

New Concepts to be learned: Java database connectivity.

Requirements:

Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

JDBC driver

A JDBC driver is a software component enabling a Java application to interact with a database . JDBC drivers are analogous to ODBC drivers, ADO.NET data providers, and OLE DB providers.

To connect with individual databases, **JDBC (the Java Database Connectivity API)** requires drivers for each database. The JDBC driver gives out the connection to the database and implements the protocol for transferring the query and result between client and database.

JDBC technology drivers fit into one of four categories
Type 1 Driver - JDBC-ODBC bridge
Type 2 Driver - Native-API Driver
Type 3 Driver - Network-Protocol Driver(MiddleWare Driver)
Type 4 Driver - Database-Protocol Driver(Pure Java Driver)

JDBC-ODBC bridge driver can be explained in detail as follows:

JDBC-ODBC bridge

The JDBC type 1 driver, also known as the JDBC-ODBC bridge, is a database driver implementation that employs the ODBC driver to connect to the database. The driver converts JDBC method calls into ODBC function calls.

The driver is platform-dependent as it makes use of ODBC which in turn depends on native libraries of the underlying operating system the JVM is running upon. Also, use of this driver leads to other installation dependencies; for example, ODBC must be installed on the computer having the driver and the database must support an ODBC driver.

Advantages:

Almost any database for which ODBC driver is installed, can be accessed.

Disadvantages:

1] Performance overhead since the calls have to go through the jdbc Overhead bridge to the ODBC driver, then to the native db connectivity interface (thus may be slower than other types of drivers).

2] The ODBC driver needs to be installed on the client machine.

3] Not suitable for applets, because the ODBC driver needs to be installed on the client.

Program Code:

```
/STEP 1. Import required packages
import java.sql.*;

public class JDBCExample {
  // JDBC driver name and database URL
  static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
  static final String DB_URL = "jdbc:mysql://localhost/";

  //  Database credentials
  static final String USER = "username";
  static final String PASS = "password";

  public static void main(String[] args) {
  Connection conn = null;
  Statement stmt = null;
  try{
    //STEP 2: Register JDBC driver
    Class.forName("com.mysql.jdbc.Driver");

    //STEP 3: Open a connection
    System.out.println("Connecting to database...");
    conn = DriverManager.getConnection(DB_URL, USER, PASS);

    //STEP 4: Execute a query
    System.out.println("Creating database...");
    stmt = conn.createStatement();

    String sql = "CREATE DATABASE STUDENTS";
    stmt.executeUpdate(sql);
    System.out.println("Database created successfully...");
  }catch(SQLException se){
    //Handle errors for JDBC
    se.printStackTrace();
  }catch(Exception e){
    //Handle errors for Class.forName
    e.printStackTrace();
  }finally{
    //finally block used to close resources
    try{
      if(stmt!=null)
        stmt.close();
    }catch(SQLException se2){
    }// nothing we can do
    try{
```

```
        if(conn!=null)
           conn.close();
      }catch(SQLException se){
        se.printStackTrace();
      }//end finally try
    }//end try
    System.out.println("Goodbye!");
}//end main
}//end JDBCExample
```

Conclusion: Thus studied and implemented java database connectivity.

Real Life Application:

Viva Questions:

1. How to perform java database connectivity?
2. What is use of database connectivity?
3. What are the steps for forming the connectivity?
4. What is advantage of java connectivity?
5. What do you mean by JDBC?
6. What is JDBC-ODBC Bridge?
7. What is the use of Connection class?
8. What is the use of recordset class?
9. Explain the methods of recordset class.
10. Explain the methods of statement class.

Experiment 13

Title: To write a program to implement B+tree

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: Understand the working of B+ tree

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Indexing using B+ tree.

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: B+ tree and B tree working .

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiments.)

New Concepts to be learned: How to write PLSQL commands.

Requirements:

Software & Hardware Required: ORACLE, PC

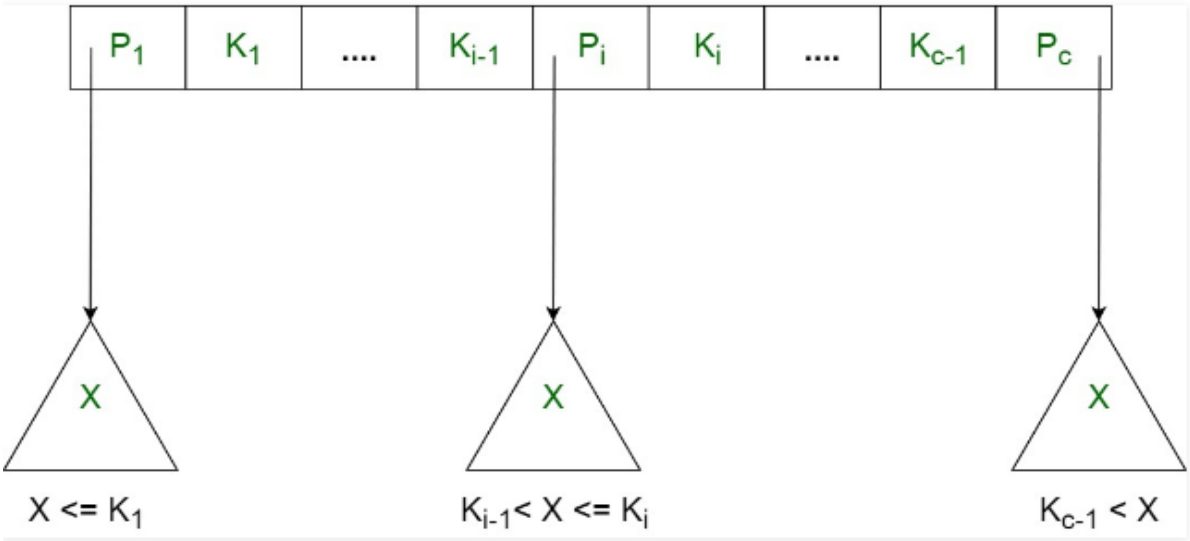
Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

In order, to implement dynamic multilevel indexing, B-tree and B+ tree are generally employed. B+ tree stores data pointers only at the leaf nodes of the tree. Thus, the structure of leaf nodes of a B+ tree is quite different from the structure of internal nodes of the B+ tree. It may be noted here that, since data pointers are present only at the leaf nodes, the leaf nodes must necessarily store all the key values along with their corresponding data pointers to the disk file block, in order to access them. Moreover, the leaf nodes are linked to provide ordered access to the records. The leaf nodes, therefore form the first level of index, with the internal nodes forming the other levels of a multilevel index. Some of the key values of the leaf nodes also appear in the internal nodes, to simply act as a medium to control the searching of a record.

The structure of the internal nodes of a B+ tree of order 'a' is as follows:

1. Each internal node is of the form :
   $<P_1, K_1, P_2, K_2, ..., P_{c-1}, K_{c-1}, P_c>$
   where c <= a and each P_i is a tree pointer (i.e points to another node of the tree) and, each K_i is a key value (see diagram-I for reference).
2. Every internal node has : $K_1 < K_2 < .... < K_{c-1}$
3. For each search field values 'X' in the sub-tree pointed at by P_i, the following condition holds :
   $K_{i-1} < X <= K_i$, for $1 < i < c$ and,
   $K_{i-1} < X$, for $i = c$
4. Each internal nodes has at most 'a' tree pointers.
5. The root node has, at least two tree pointers, while the other internal nodes have at least \ceil(a/2) tree pointers each.
6. If any internal node has 'c' pointers, c <= a, then it has 'c – 1' key values.



B+ Tree

Program code and output:

```
#include<stdio.h>
#include<conio.h>
#include<iostream>
using namespace std;
struct BTreeNode
{
   int *data;
   BTreeNode **child_ptr;
   bool leaf;
   int n;
}*root = NULL, *np = NULL, *x = NULL;
BTreeNode * init()
{
   int i;
   np = new BTreeNode;
   np->data = new int[5];
   np->child_ptr = new BTreeNode *[6];
   np->leaf = true;
   np->n = 0;
   for (i = 0; i < 6; i++)
   {
      np->child_ptr[i] = NULL;
   }
   return np;
}
void traverse(BTreeNode *p)
{
   cout<<endl;
   int i;
   for (i = 0; i < p->n; i++)
```

```
    {
        if (p->leaf == false)
        {
            traverse(p->child_ptr[i]);
        }
        cout << " " << p->data[i];
    }
    if (p->leaf == false)
    {
        traverse(p->child_ptr[i]);
    }
    cout<<endl;
}
void sort(int *p, int n)
{
    int i, j, temp;
    for (i = 0; i < n; i++)
    {
        for (j = i; j <= n; j++)
        {
            if (p[i] > p[j])
            {
                temp = p[i];
                p[i] = p[j];
                p[j] = temp;
            }
        }
    }
}
int split_child(BTreeNode *x, int i)
{
    int j, mid;
    BTreeNode *np1, *np3, *y;
    np3 = init();
    np3->leaf = true;
    if (i == -1)
    {
        mid = x->data[2];
        x->data[2] = 0;
        x->n--;
        np1 = init();
        np1->leaf = false;
        x->leaf = true;
        for (j = 3; j < 5; j++)
        {
            np3->data[j - 3] = x->data[j];
            np3->child_ptr[j - 3] = x->child_ptr[j];
            np3->n++;
            x->data[j] = 0;
            x->n--;
```

```
        }
        for(j = 0; j < 6; j++)
        {
            x->child_ptr[j] = NULL;
        }
        np1->data[0] = mid;
        np1->child_ptr[np1->n] = x;
        np1->child_ptr[np1->n + 1] = np3;
        np1->n++;
        root = np1;
    }
    else
    {
        y = x->child_ptr[i];
        mid = y->data[2];
        y->data[2] = 0;
        y->n--;
        for (j = 3; j < 5; j++)
        {
            np3->data[j - 3] = y->data[j];
            np3->n++;
            y->data[j] = 0;
            y->n--;
        }
        x->child_ptr[i + 1] = y;
        x->child_ptr[i + 1] = np3;
    }
    return mid;
}
void insert(int a)
{
    int i, temp;
    x = root;
    if (x == NULL)
    {
        root = init();
        x = root;
    }
    else
    {
        if (x->leaf == true && x->n == 5)
        {
            temp = split_child(x, -1);
            x = root;
            for (i = 0; i < (x->n); i++)
            {
                if ((a > x->data[i]) && (a < x->data[i + 1]))
                {
                    i++;
                    break;
```

```
            }
            else if (a < x->data[0])
            {
               break;
            }
            else
            {
               continue;
            }
         }
         x = x->child_ptr[i];
      }
      else
      {
         while (x->leaf == false)
         {
         for (i = 0; i < (x->n); i++)
         {
            if ((a > x->data[i]) && (a < x->data[i + 1]))
            {
               i++;
               break;
            }
            else if (a < x->data[0])
            {
               break;
            }
            else
            {
               continue;
            }
         }
            if ((x->child_ptr[i])->n == 5)
            {
               temp = split_child(x, i);
               x->data[x->n] = temp;
               x->n++;
               continue;
            }
            else
            {
               x = x->child_ptr[i];
            }
         }
      }
   }
   x->data[x->n] = a;
   sort(x->data, x->n);
   x->n++;
}
```

```
int main()
{
   int i, n, t;
   cout<<"enter the no of elements to be inserted\n";
   cin>>n;
   for(i = 0; i < n; i++)
   {
      cout<<"enter the element\n";
      cin>>t;
      insert(t);
   }
   cout<<"traversal of constructed tree\n";
   traverse(root);
   getch();
}
```

Input & Output:

```
enter the no of elements to be inserted
5
enter the element
12
enter the element
45
enter the element
32
enter the element
67
enter the element
89
traversal of constructed tree

 12 32 45 67 89
```

Conclusion: Thus the B+ tree program is implemented and the working is understood.

Real Life Application: Indexing is used in every application where data is more to make search effective for example Railway Reservation System.

Viva Questions:

1. What is Indexing?
2. Why Indexing is important?

3. Types of indexing?

Post Lab Questions:

1. What is advantage of B+ tree indexing over B tree?
2. Drawbacks of B+ tree.
3. What is search tree?

Experiment 14

Title: To design Cursors using PL SQL query block

Estimated time to complete this experiment: (in hours) -- 2hrs

---

Objective: Understand the programming SQL language.

PEO to be achieved:

PEO1. Graduates will lead a successful career in diverse domains.

PEO2. The graduate will analyze real life problems and design technically sound, economically feasible and socially acceptable solutions

PEO3. Graduates will exhibit professionalism, ethics, teamwork, social awareness and agility through lifelong learning.

Expected Outcome of Experiment: Imposing constraints on database to maintain integrity

---

Books/ Journals/ Websites referred:

- Korth, Silberchatz, Sudarshan, "Database System Concepts", McGraw-Hill
- Dr P.S.Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press
- www.w3schools.com

Pre Lab/ Prior Concepts: Relational database Storage structure .

Historical Profile:

(Write name of person invented the logic/ algorithm/ law for this current experiments.)

New Concepts to be learned: How to write PLSQL commands.

Requirements:

Software & Hardware Required: ORACLE, PC

Flow Chart: (for INFT & CMPN)

Explanation/ Stepwise-Procedure/ Algorithm:

PL/SQL: **Procedural Language/Structured Query Language (PL/SQL)** is Oracle Corporation's procedural extension language for SQL and the Oracle relational database.
PL/SQL supports variables, conditions, loops ,exceptions and arrays .

Variable declaration in PL/SQL:

Syntax:
<varname><datatype>[not null]:=initial value

Conditional statements in PL/SQL:

1] If –else :-

Syntax:

```
If condition then
Statements;
Else
Statements;
End if;
```

2] Case:-

Syntax:

```
When condition_1 Then
 Sequence of statements_1;
When condition_2 Then
Sequence of statements_2;
When condition_3 Then
Sequence of statements_3;
Else
Sequence of statements_N;
End Case;
```

Looping:

Syntax of looping:

```
Loop
Statements;
Exit; or exit when condition
End loop;
```

1] While loop:

Syntax:

```
WHILE condition
```

```
LOOP
  Statements;
END LOOP;
```

2] For loop:

Syntax:

```
For counter in [reverse]<lower>..<upper>loop
Statements;
End loop;
```

The total PLSQL block can be represented as:

```
Declare
                    (Declarations)

Begin
                  (program
statements)

Exception

                    (exceptions)
End;
```

Program code and output:

Queries:

PL/SQL 1: Write a PL/SQL block to check whether the Record is present or not.

```
set serveroutput on;
declare
        id Events08.eventcode%type;
        ec Events08.name%type;
        n int;
begin

select eventcode,name
into id,ec
from Events08
where eventcode=&n;

if ec='Dance' then
dbms_output.put_line(id || '=' || ec);
```

```
else
dbms_output.put_line('Rec not found');
end if;
end;
```

Output:

1:

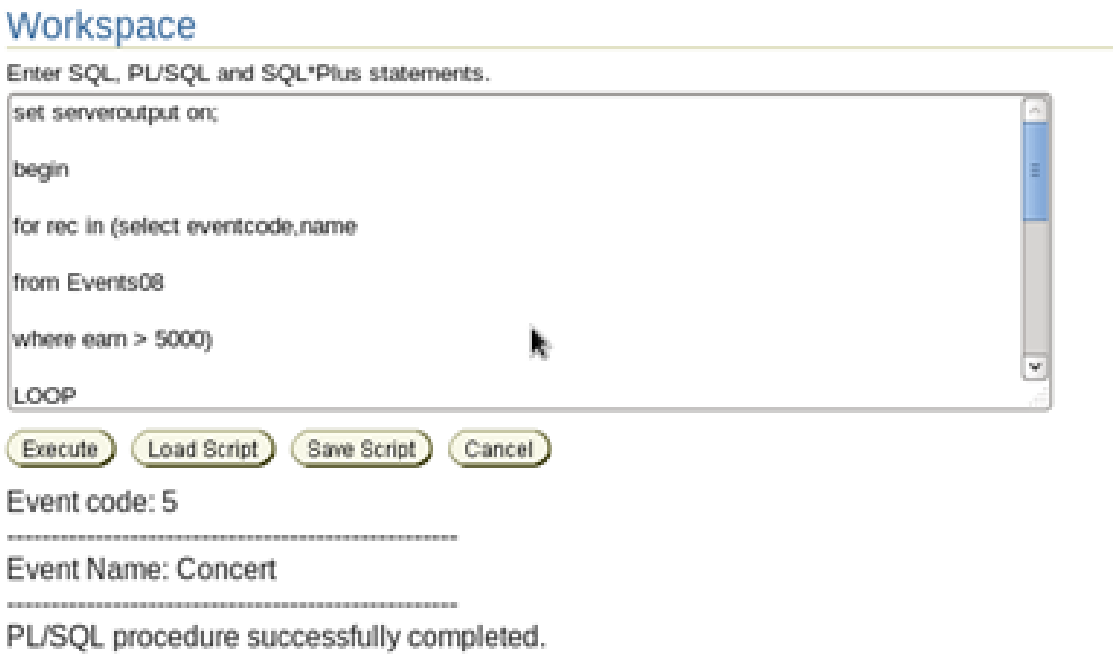
2.

PL/SQL 2: Write a PL/SQL block to display all the id and name of members whose age is more than 19.

```
set serveroutput on;
```

```
begin
for rec in (select eventcode,name
from Events08
where earn > 5000)
LOOP
DBMS_OUTPUT.PUT_LINE ('Event code: ' || rec.eventcode);
DBMS_OUTPUT.PUT_LINE ('------------------------------------------------');
DBMS_OUTPUT.PUT_LINE ('Event Name: ' || rec.name);
DBMS_OUTPUT.PUT_LINE ('------------------------------------------------');
END LOOP;
end;
```

Output:



Conclusion: Thus we have designed and implemented Cursor using PLSQL block.

Real Life Application: Used in advanced database programming. Also used in writing transactions and triggers in the database.

Viva Questions:

4. What is PL/SQL?
5. What PL/SQL stands for?
6. What is the structure for PL/SQL?
7. Write the PL/SQL code for “HELLO WORLD”?
8. What are advantages and disadvantages of PL/SQL?

Post Lab Questions:

4. What is the structure for PL/SQL?
5. What are advantages and disadvantages of PL/SQL?
6. Use of PL/SQL?
7. What is the basic structure of PL/SQL block?
8. How to declare a variable with column data type in PL/SQL?
9. What are the basic control statements in PL/SQL.