ACHARYA INSTITUTE OF TECHNOLOGY

Acharya Dr. Sarvepalli Radhakrishnan Road, Soladevanahalli,

Bengaluru, Karnataka 560107

(*Affiliated to VTU, Jnana Sangama, Belagavi***)**

Department of Artificial Intelligence and Machine Learning

DATABASE MANAGEMENT SYSTEM

B.E - IV Semester, Artificial Intelligence and Machine Learning

[As per Choice Based Credit System (CBCS) scheme]

Subject Code - BCS403

Lab Manual – 2024-25

Name : _____

USN: _____

Branch: _____ **Section:** _____

ACHARYA INSTITUTE OF TECHNOLOGY

Acharya Dr. Sarvepalli Radhakrishnan Road, Soladevanahalli,

Bengaluru, Karnataka 560107

(*Affiliated to VTU, Jnana Sangama, Belagavi*)

Department of Artificial Intelligence and Machine Learning

DATABASE MANAGEMENT SYSTEM

B.E - IV Semester, Artificial Intelligence and Machine Learning

[As per Choice Based Credit System (CBCS) scheme]

Subject Code - BCS403

Prepared By:

Md. Tahir Mirji
Assistant Professor, Department of AIML & CSE(DS),
AIT, Acharya, Bangalore -107

ACHARYA INSTITUTE OF TECHNOLOGY MOTTO

"Nurturing Aspirations Supporting Growth"

VISION

"Acharya Institute of Technology, committed to the cause of sustainable value-based education in all disciplines, envisions itself as a global fountainhead of innovative human enterprise, with inspirational initiatives for Academic Excellence".

MISSION

"Acharya Institute of Technology strives to provide excellent academic ambiance to the students for achieving global standards of technical education, foster intellectual and personal development, meaningful research and ethical service to sustainable societal needs."

Department of Artificial Intelligence and Machine Learning

VISION

To be recognized as the leader in the field of Artificial Intelligence and Machine Learning by nurturing and producing quality next-generation academicians and researchers with human values, who are creative, innovative, and versatile in this fast-growing field.

MISSION

The Department of Artificial Intelligence and Machine Learning (AI and ML) @ Acharya Institute of Technology's mission is to produce quality students with a sound understanding of the fundamentals of the theory and practice of Artificial Intelligence and Machine Learning. The mission is also to enable students to be leaders in the industry and academia nationally and internationally. Finally, the mission is to meet the strong demands of the nation in the areas of Artificial Intelligence and Machine Learning.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Students shall have a successful professional career in industry, academia, R & D organization or entrepreneur in specialized fields of Artificial Intelligence and Machine Learning and allied disciplines.

P**EO2:** Students shall be competent, creative and valued professionals in the chosen field.

PEO3: Engage in life-long learning and professional development.

PEO4: Become effective global collaborators, leading or participating to address technical, business, environmental and societal challenges.

PROGRAM OUTCOMES (POs)

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for

sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

O9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

Table of Contents

i) Do’s and Don’ts in the lab

DO’S

1. Please leave footwear outside the laboratory at the designated place.
2. Please keep your belongings such as bags in the designated place.
3. Turn off the respective systems and arrange the chairs before you
4. leaving the laboratory.
5. Maintain Silence and Discipline inside the laboratory.
6. Wear your ID card & Carry observation and record while coming to
7. the laboratory.

DON'TS

1. Don’t use mobile cell phones during lab hours.
2. Do not eat food, chew gum in the laboratory.
3. Do not install, uninstall or alter any software on the computer.
4. Students are not allowed to work in a laboratory alone or without the
5. presence of faculty or instructor.
6. Do not move any equipment from its original position.
7. We are not responsible for any belongings left behind.
8. Do not enter the laboratory without permission.

ii) RULES FOR MAINTAINING LABORATORY RECORD

Always carry observation book to lab sessions along with laptop with proper software being installed.

Put your name, USN, and subject on the certificate front cover of the record. Put that the same information on the first page inside.

- Update the Table of Contents every time you start each new experiment or topic.
- Always use a black pen and write neatly and clearly (punctuations missing are seen as errors).
- Obvious care should be taken to make it readable, even if you have bad handwriting.
- Start each new topic (experiment, notes, calculation, etc.) on a right-side (odd-numbered) page.
- Date to be written every page on the top right-side corner.
- On each left side page
    - Write the output
- On each left side page
    - Write the experiment/procedure
- Use labels and captions for figures and tables if any.
- Strictly observe the instructions given by the Teacher/ Lab Instructor.

INDEX

Exp. No.	List of Experiments Planned	CO mapped
1	Create a table called Employee & execute the following. **Employee(EMPNO,ENAME,JOB, MANAGER_NO, SAL, COMMISSION)** 1. Create a user and grant all permissions to the user. 2. Insert any three records in the employee table containing attributes EMPNO,ENAME JOB, MANAGER_NO, SAL, COMMISSION and use rollback. Check the result. 3. Add primary key constraint and not null constraint to the employee table. 4. Insert null values to the employee table and verify the result.	
2	Create a table called Employee that contain attributes EMPNO,ENAME,JOB, MGR,SAL & execute the following. 1. Add a column commission with domain to the Employee table. 2. Insert any five records into the table. 3. Update the column details of job 4. Rename the column of Employee table using alter command. 5. Delete the employee whose Empno is 105.	
3	Queries using aggregate functions(COUNT,AVG,MIN,MAX,SUM),Group by,Orderby. **Employee(E_id, E_name, Age, Salary)** 1. Create Employee table containing all Records E_id, E_name, Age, Salary. 2. Count number of employee names from employee table 3. Find the Maximum age from employee table. 4. Find the Minimum age from employee. 5. Find salaries of employee in Ascending Order. 6. Find grouped salaries of employees.	

4	Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old & new Salary. **CUSTOMERS(ID,NAME,AGE,ADDRESS,SALARY)**	
5	Create cursor for Employee table & extract the values from the table. Declare the variables, Open the cursor & extrct the values from the cursor. Close the cursor. **Employee(E_id, E_name, Age, Salary)**	
6	Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped.	
7	Install an Open Source NoSQL Data base MangoDB & perform basic CRUD(Create, Read, Update & Delete) operations. Execute MangoDB basic Queries using CRUD operations.	
Additional Exercise		
8	**Activity-Based Learning (Suggested Activities in Class)/Practical-Based learning Mini Project:** Project-Based Learning (CRUD Operation with PHP)	

PRACTICAL COMPONENT OF IPCC

Sl.NO	Experiments
1	Create a table called Employee & execute the following. **Employee(EMPNO,ENAME,JOB, MANAGER_NO, SAL, COMMISSION)** 1. Create a user and grant all permissions to theuser. 2. Insert the any three records in the employee table contains attributes EMPNO,ENAME JOB, MANAGER_NO, SAL, COMMISSION and use rollback. Check the result. 3. Add primary key constraint and not null constraint to the employee table. 4. Insert null values to the employee table and verify the result.
2	Create a table called Employee that contain attributes EMPNO,ENAME,JOB, MGR,SAL & execute the following. 1. Add a column commission with domain to the Employeetable. 2. Insert any five records into the table. 3. Update the column details of job 4. Rename the column of Employ table using alter command. 5. Delete the employee whose Empno is 105.
3	Queries using aggregate functions(COUNT,AVG,MIN,MAX,SUM),Group by,Orderby. **Employee(E_id, E_name, Age, Salary)** 1. Create Employee table containing all Records E_id, E_name, Age, Salary. 2. Count number of employee names from employeetable 3. Find the Maximum age from employee table. 4. Find the Minimum age from employeetable. 5. Find salaries of employee in Ascending Order. 6. Find grouped salaries of employees.
4	Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old & new Salary. **CUSTOMERS(ID,NAME,AGE,ADDRESS,SALARY)**
5	Create cursor for Employee table & extract the values from the table. Declare the variables ,Open the cursor & extrct the values from the cursor. Close the cursor. **Employee(E_id, E_name, Age, Salary)**
6	Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped.
7	Install an Open Source NoSQL Data base MangoDB & perform basic CRUD(Create, Read, Update & Delete) operations. Execute MangoDB basic Queries using CRUD operations.

Assessment Rubrix Details (both CIE)

CIE for the practical component of the IPCC

● **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.

● On completion of every experiment/program in the laboratory, the students shall be evaluated including viva voce and marks shall be awarded on the same day.

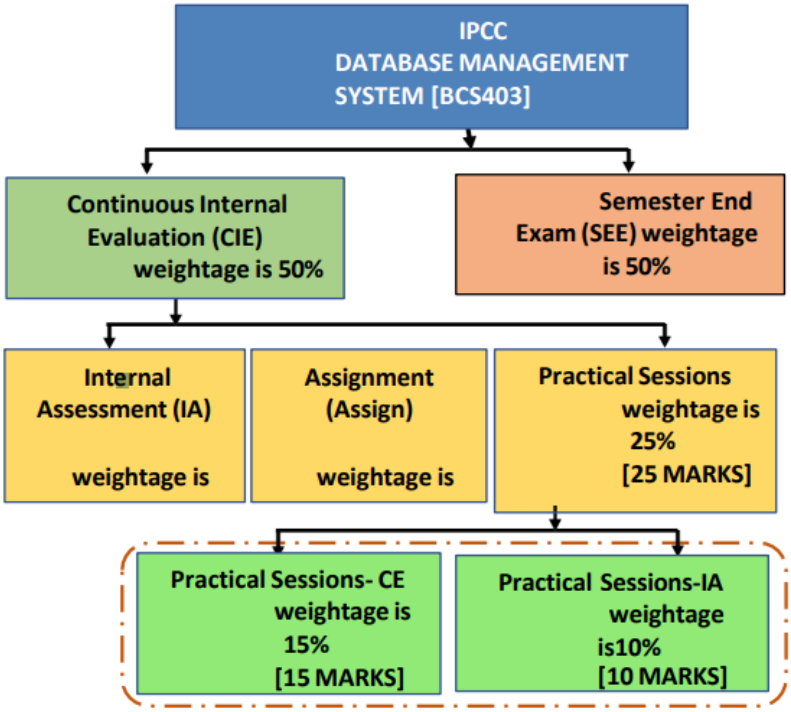
● The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.

● The laboratory test **(duration 02/03 hours)** after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks.**

● Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**.

● The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

ASSESSMENT PROCESS FOR INTEGRATED PROFESSIONAL CORE COURSE (IPCC)



Practical (50 Marks):

1. Test: 50 Marks Conduction
2. Experiment Conduction, Observation, and Record: 15 Marks

Practical Evaluation(15):

Each experiment is evaluated as follows:

- Attendance (if present: 2, if absent: 0)
- Observation (2/1)
- Record (4/2)
- Viva (2/1)

DBMS LABORATORY EXERCISES

DBMS Lab Theory

Instructions: Theory and commands to be written on lined(right) side pages output on the left side.

SQL (Structured Query Language):

- Data Definition Language (DDL): Creating, altering, and dropping tables, views, indexes, etc.
- Data Manipulation Language (DML): Inserting, updating, deleting, and retrieving data.
- Data Control Language (DCL): Granting and revoking user privileges.
- Transaction Control Language (TCL): Managing transactions (COMMIT, ROLLBACK).
- Complex Queries: Using joins, subqueries, aggregate functions, and other advanced SQL features.
- DBMS Software
  e.g., MySQL, PostgreSQL, Oracle

Data Definition Language (DDL):

- **Table Creation:** CREATE TABLE (defining table structure, attributes, data types).
- **Constraints:**
  - PRIMARY KEY: Enforcing uniqueness and non-null values for identifying records.
  - NOT NULL: Ensuring that specific columns cannot contain null values.
  - Adding and altering columns: ALTER TABLE ADD COLUMN, ALTER TABLE MODIFY COLUMN.
  - Renaming columns: ALTER TABLE RENAME COLUMN.
- **User Management:** CREATE USER, GRANT (managing user permissions and privileges).
- **Dropping data:** DELETE FROM (removing records).

Data Manipulation Language (DML):

- **Data Insertion:** INSERT INTO (adding new records).
- **Data Update:** UPDATE (modifying existing records).
- **Data Retrieval:** SELECT (querying and retrieving data).
- **Transactions:** ROLLBACK (undoing changes).

Aggregate Functions:

- COUNT(): Counting the number of rows.
- AVG(): Calculating the average value.

- MIN(): Finding the minimum value.
- MAX(): Finding the maximum value.
- SUM(): Calculating the sum of values.

Querying and Sorting:

- GROUP BY: Grouping rows based on specified columns.
- ORDER BY: Sorting rows in ascending or descending order.

Triggers:

- Row-level triggers: Executing code for each affected row during INSERT, UPDATE, or DELETE operations.
- Detecting changes in data values.

Cursors:

- Declaring, opening, fetching, and closing cursors: Iterating through result sets row by row.
- Parameterized cursors: Using variables in cursor queries.
- Data merging using cursors.

PL/SQL (Procedural Language/SQL):

- Writing blocks of procedural code within the database.
- Implementing conditional logic and loops.

NoSQL (MongoDB):

- Basic CRUD operations (Create, Read, Update, Delete).
- Understanding document-oriented databases.
- Executing basic NoSQL queries.

Experiment 1. Create a table called Employee & execute the following.

Employee(EMPNO,ENAME,JOB,MANAGER_NO,SAL,COMMISSION)

a. Create a user and grant all permissions to the user.

b. Insert the any three records in the employee table contains attributes EMPNO,ENAME JOB, MANAGER_NO, SAL, COMMISSION and use rollback. Check the result.

c. Add primary key constraint and not null constraint to the employee table.

d. Insert null values to the employee table and verify the result.

SOLUTION.

Step 1: Create a user and grant all permissions:

```
CREATE USER 'new_user'@'localhost' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON *.* TO 'new_user'@'localhost';
```

This command reloads the grant tables, ensuring that any changes to user privileges take effect immediately.

```
FLUSH PRIVILEGES;
```

Step 2: Create the Employee table and insert records using rollback:

```
CREATE TABLE Employee
( EMPNO   INT,
 ENAME   VARCHAR (50),
JOB         VARCHAR (50),
MANAGER_NO   INT,
SAL       DECIMAL (10,2)
COMMISSION    DECIMAL (10,2)
);
```

Step 3: Inserting three records:

START TRANSACTION;

INSERT INTO Employee (EMPNO, ENAME, JOB, MANAGER_NO, SAL,

COMMISSION) VALUES (1, 'John Doe', 'Manager', NULL, 50000.00, 1000.00),

(2, 'Jane Smith', 'Developer', 1, 40000.00, NULL),

(3, 'Alice Johnson', 'Salesperson', 1, 30000.00, 500.00);

display records inserted recently

SELECT * FROM Employee;

Use rollback to undo the insertions:

ROLLBACK;

Check that the recent records is rolled back (should return 0 rows)

SELECT * FROM Employee;

Step 4: Add primary key constraint and not null constraint:

Alter table to add primary key constraint on EMPNO:

ALTER TABLE Employee **ADD CONSTRAINT** pk_employee **PRIMARY KEY** (EMPNO);

Alter table to add NOT NULL constraints on required columns:

ALTER TABLE Employee **MODIFY** (ENAME VARCHAR2(50) NOT NULL,

JOB VARCHAR2(50) NOT NULL, SAL NUMBER NOT NULL);

Step 5: Insert null values to the employee table and verify the result

Attempt to insert a record with a NULL value in a NOT NULL column (ENAME):

INSERT INTO Employee (EMPNO, ENAME, JOB, MANAGER_NO,

SAL, COMMISSION) **VALUES** (4, NULL, 'Tester', 2, 3000, 200);

This insert will fail due to the NOT NULL constraint on ENAME

Check the result by querying the table:

SELECT * FROM Employee;