



Format No. QSP/7.1/01.F01 (B)

Issue No.05 Rev. No 5 Dated: Jan 1, 2017

UNIVERSITY OF PETROLEUM & ENERGY STUDIES

School of Computer Science

Dehradun

COURSE PLAN

Programme : B. Tech (CSE) with **Specialization in Cloud Computing &**

Virtualization Technology

Course : Advanced Database Management System Lab.

Course Code : CSEG2105

No. of credits : 1

Semester : IV

Session : 2022-23 (Jan-May-2023)

Batch : 2021-25

Instructor : Dr. Neeraj Chugh

Email : nchugh@ddn.upes.ac.in

Approved By

Faculty

UPES Campus

“Energy Acres”

P.O. Bidholi, Via Prem Nagar, Dehradun

HOD

Tel: +91-135-2770137

Fax: +91 135- 27760904

Website: www.upes.ac.in



COURSE PLAN

A. PREREQUISITE:

- a. Basic Knowledge Database and DBMS

B. PROGRAM OUTCOMES (POs) and PROGRAM SPECIFIC OUTCOMES (PSOs):

B1. 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.

PO9: 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.

B2. Program Specific Outcomes (PSOs)

PSO1: Perform system and application programming using computer system concepts, concepts of Data Structures, algorithm development, problem solving and optimizing techniques.

PSO2: Apply software development and project management methodologies using concepts of front-end and back-end development and emerging technologies and platforms.

PSO3: Understand and apply Cloud Computing architecture for scalable, secure and dynamically provisioned business oriented environment with optimized performance tuning and data reliability.

C. Course Outcomes for Advanced Database Management Systems

Course Outcomes

CO1. Implement a database schema for a given problem-domain using DDL/DML/DCL commands.

CO2. Use SQL queries in databases.

CO3. Implement Cursors & Triggers using PL\SQL.



Table: Mapping of POs and PSOs v/s COs

PO / CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	2	3	2										2	3	
CO2	2	3	2										2	3	
CO3	2	3	2										2	3	
Average	2	3	2										2	3	

1=weak

2= moderate

3=strong

Table: Correlation of POs and PSOs v/s COs

Table: Mapping of COs with POs and PSOs

Course Outcomes (COs)		Mapped PO & PSO
CO1	Implement a database schema for a given problem-domain using DDL/DML/DCL commands.	PO1, PO2, PO3, PSO1, PSO2
CO2	Use SQL queries in databases.	PO1, PO2, PO3, PSO1, PSO2
CO3	Implement Cursors & Triggers using PL\SQL.	PO1, PO2, PO3, PSO1, PSO2

COURSE OUTLINE

Expt. No.	Big Ideas/ Topics	CO	Modality
1	DDL & DML commands	1	F2F/e-lab
2	Data and Integrity constraints	2	F2F/e-lab
3	SQL subquery	3	F2F/e-lab
4	Inbuilt functions and relational algebra	2	F2F/e-lab
5	SQL clauses and Join	2	F2F/e-lab
6	Views	2	F2F/e-lab
7	Index	2	F2F/e-lab
8	Sequence	3	F2F/e-lab
9	PL/SQL Programming building block	3	F2F/e-lab
10	Function & Procedure in PL/SQL	3	F2F/e-lab
11	Implicit and Explicit Cursor	3	F2F/e-lab
12	Trigger	3	F2F/e-lab

D. PEDAGOGY

- Solution to the problems should be written and tested.
- Students are evaluated based on Performance (via efficient design, implementation), record keeping, and preparation of students (via viva-voce and quiz).

E. COURSE COMPLETION PLAN

No. of experiments planned	No. of experiments completed	No. of Quiz planned	No. of Quiz conducted	No. of internal viva planned	No. of internal viva conducted
12		1		2	

No. of lab classes per batch planned: 12

One Session = 120 minutes

F. EVALUATION & GRADING

Description	Weightage	Schedule
1. Performance Records	50%	During Lab Sessions
2. Viva Voce and Quiz	50%	One viva and quiz before mid-semester and one viva and quiz after mid-semester

F1. Performance Records: Marks 100 (50% weightage)

10 Marks for each lab, consider marks of best 10 labs out of 12	
Performance & Records	7 Marks
Lab File	3 Marks

Lab performance and record evaluation shall be a continuous process throughout the semester.

F2. Viva voce and Quiz: Marks 100 (50%)

- Viva 1: 25 Marks



- Quiz 1: 25 Marks
- Viva 2: 25 Marks
- Quiz 2: 25 Marks

Conduct viva of 10 students in one lab session. So, one complete viva will be conducted in 3 Lab Sessions.

F3. GENERAL DISCIPLINE

Based on student's regularity, punctuality, sincerity and participation in the interactions. *The marks obtained by the students will be displayed on LMS at the end of semester.*

F4. GRADING

The overall marks obtained at the end of the semester comprising all the above two mentioned shall be converted to a grade. Students scoring less than 35 composite marks shall be awarded a 'F' grade. Students scoring 85 marks and above as composite score shall be awarded a 'O' grade. The minimum individual course grade is 'C'.

The student who is debarred due to shortage of attendance for a lab course will need to repeat the continuous evaluation during summer vacation (June-July) after the registration by payment of the prescribed fee per subject as notified by the University. The grades are awarded based on the performance of the students and capping as per the University guidelines. All other rules and regulations such as requirement of passing, etc. will remain same.

G. COURSE DELIVERY PLAN

Activities listed under 'Lab activities' should be performed F2F (face-to-face) in the allocated laboratories (either compiled in lab machines or compiled online using e-labs).

1. EXPERIMENT-1

Title: To understand DDL and DML commands

Objective: To understand the concept of designing issue related to the database with creating, populating the tables. Also familiarize students with different ways of manipulation in database.

1. **Create the tables described below:**



Table name: CLIENT_MASTER

Description: used to store client information.

Column name	data type	Size
CLIENTNO	Varchar	6
NAME	Varchar	20
ADDRESS 1	Varchar	30
ADDRESS 2	Varchar	30
CITY	Varchar	15
PINCODE	Integer	
STATE	Varchar	15
BALDUE	Number	10,2

Table Name: PRODUCT_MASTER

Description: used to store product information

Column name	data type	Size
PRODUCTNO	Varchar	6
DESCRIPTION	Varchar	15
PROFITPERCENT	Decimal	4,2
UNIT MEASURE	Varchar	10
QTYONHAND	Integer	
REORDERL VL	Integer	
SELLPRICE	Number	8,2
COSTPRICE	Number	8,2

Table Name: SALESMAN_MASTER

Description: Used to store salesman information working for the company.

Column name	data type	Size
SALESMANNO	Varchar	6
SALESMANNAME	Varchar	20
ADDRESS 1	Varchar	30
ADDRESS 2	Varchar	30
CITY	Varchar	20



PINCODE	Integer	
STATE	Varchar	20
SALAMT	Real	
TGTTTOGET	Decimal	
YTDSALES	Double	6,2
REMARKS	Varchar	60

2. **Insert the following data into their respective tables:**

a) Data for **CLIENT_MASTER** table:

Client no	Name	City	Pincode	State	BalDue
C00001	Ivan bayross	Mumbai	400054	Maharashtra	15000
C00002	Mamta muzumdar	Madras	780001	Tamil nadu	0
C00003	Chhaya bankar	Mumbai	400057	Maharashtra	5000
C00004	Ashwini joshi	Bangalore	560001	Karnataka	0
C00005	Hansel colaco	Mumbai	400060	Maharashtra	2000
C00006	Deepak sharma	Mangalore	560050	Karnataka	0

b) Data for **PRODUCT_MASTER** table:

Product No	Description	Profit percent	Unit measure	Quantity On hand	Recorder Level	Sell Price	Cost Price
P00001	T-Shirt	5	Piece	200	50	350	250
P0345	Shirts	6	Piece	150	50	500	350
P06734	Cotton jeans	5	Piece	100	20	600	450
P07865	Jeans	5	Piece	100	20	750	500
P07868	Trousers	2	Piece	150	50	850	550
P07885	Pull Overs	2.5	Piece	80	30	700	450
P07965	Denim jeans	4	Piece	100	40	350	250
P07975	Lycra tops	5	Piece	70	30	300	175
P08865	Skirts	5	Piece	75	30	450	300

c) Data for **SALESMAN_MASTER** table:



Salesman No	Name	Address1	Address2	City	Pin Code	State
S00001	Aman	A/14	Worli	Mumbai	400002	Maharashtra
S00002	Omkar	65	Nariman	Mumbai	400001	Maharashtra
S00003	Raj	P-7	Bandra	Mumbai	400032	Maharashtra
S00004	Ashish	A/5	Juhu	Mumbai	400044	Maharashtra

3. **Exercise on retrieving records from a table.**

- Find out the names of all the clients.
- Retrieve the entire contents of the Client_Master table.
- Retrieve the list of names, city and the state of all the clients.
- List the various products available from the Product_Master table.
- List all the clients who are located in Mumbai.
- Find the names of salesman who have a salary equal to Rs.3000.

4. **Exercise on updating records in a table**

- Change the city of ClientNo 'C00005' to 'Bangalore'.
- Change the BalDue of ClientNo 'C00001' to Rs.1000.
- Change the cost price of 'Trousers' to rs.950.00.
- Change the city of the salesman to Pune.

5. **Exercise on deleting records in a table**

- Delete all salesman from the Salesman_Master whose salaries are equal to Rs.3500.
- Delete all products from Product_Master where the quantity on hand is equal to 100.
- Delete from Client_Master where the column state holds the value 'Tamil Nadu'.

6. **Exercise on altering the table structure**

- Add a column called 'Telephone' of data type integer to the Client_Master table.
- Change the size off SellPrice column in Product _Master to 10, 2.

7. **Exercise on deleting the table structure along with the data**

- Destroy the table Client_Master along with its data.

8. **Exercise on renaming the table**

- Change the name of the Salesman_Master to sman_mast.



2. EXPERIMENT-2

Title: 2. To understand and apply the concept of Constraints.

Objective: To understand the concept of data constraints that is enforced on data being stored in the table. Focus on Primary Key and the Foreign Key.

1. Create the tables described below:

Table name: CLIENT_MASTER_1

Description: used to store client information.

Column name	data type	Size	Constraints
CLIENTNO	Varchar	6	Primary key / first letter must start with 'C'
NAME	Varchar	20	Not Null
ADDRESS 1	Varchar	30	
ADDRESS 2	Varchar	30	
CITY	Varchar	15	
PINCODE	Integer	8	
STATE	Varchar	15	
BALDUE	Decimal	10,2	

Table Name: PRODUCT_MASTER_1

Description: used to store product information

Column name	data type	Size	Attributes
PRODUCTNO	Varchar	6	Primary Key/ first letter must start with 'P'
DESCRIPTION	Varchar	15	Not Null
PROFITPERCENT	Decimal	4,2	Not Null
UNIT MEASURE	Varchar	10	Not Null
QTYONHAND	Integer	8	Not Null
REORDERL VL	Integer	8	Not Null
SELLPRICE	Decimal	8,2	Not Null
COSTPRICE	Decimal	8,2	Not Null

Table Name: SALESMAN_MASTER_1

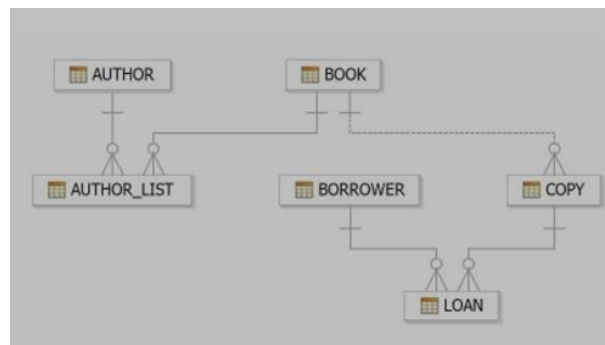
Description: used to store salesman information working for the company.

Column name	data type	Size	Attributes
-------------	-----------	------	------------

SALESMANNO	Varchar	6	Primary Key/ first letter must start with 'S'
SALESMANNAME	Varchar	20	Not Null
ADDRESS 1	Varchar	30	Not Null
ADDRESS 2	Varchar	30	
CITY	Varchar	20	
PINCODE	Integer	8	
STATE	Varchar	20	
SALAMT	Real	8,2	Not Null , Cannot be 0
TGTTGET	Decimal	6,2	Not Null , Cannot be 0
YTDSALES	Double	6,2	Not Null
REMARKS	Varchar	60	

2. Reinsert the data in these two tables based upon Lab 1.

3. Display the contents of each table.



4. Create table AUTHOR = { Author_ID , Lastname, Firstname, Email, City, Country}

Where:

Author_ID – text data type, 5 characters, primary key

Lastname – text data type, 15 characters, not null

Firstname – text data type, 15 characters, not null

Email – text data type, 40 characters,

City – text data type, 15 characters,

Country – text data type, 15 characters,

5. Create Table BOOK={ Book_ID, Book_Title, Copies)

Where :



Book_ID – text data type, 5 characters Primary Key Start With Character B

Book_Title - Text data Type Not Null

Copies- No.of copies Data Type int always greater the 2

6. Create table AUTHOR_LIST = {Author_ID ,Book_ID , Role}

Where:

Author_ID – text data type, 5 characters, referenced by Author_ID from AUTHOR table

Book_ID – text data type, 5 characters

Role – text data type, 15 characters

and primary key is: Author_ID, Book_ID

7. Add four records in each tables AUTHOR, BOOK, BOOK_LIST.

8. Alter structure of table AUTHOR_LIST add the field Publisher data type of 30 Character.

3. EXPERIMENT-3

Title: 3. To understand and use SQL Sub-Query

Objective: To understand the use of sql subquery.

1. Create the following table.

Supplier-(scode,sname,scity,turnover)

Part-(pcode,weigh,color,cost,sellingprice)

Supplier_Part-(scode,pcode,qty)

2. Populate the table

3. Write appropriate SQL Statement for the following:

1. Get the supplier number and part number in ascending order of supplier number.
2. Get the details of supplier who operate from Bombay with turnover 50.
3. Get the total number of supplier.
4. Get the part number weighing between 25 and 35.
5. Get the supplier number whose turnover is null.
6. Get the part number that cost 20, 30 or 40 rupees.
7. Get the total quantity of part 2 that is supplied.
8. Get the name of supplier who supply part 2.
9. Get the part number whose cost is greater than the average cost.



10. Get the supplier number and turnover in descending order of turnover.

4. EXPERIMENT-4

Title: 4. Use of Inbuilt functions and relational algebra operation

Objective: To understand the use of inbuilt function and relational algebra with sql query.

1. Create the following two tables (EMP and DEPT)

EMP TABLE

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
-----	-----	-----	-----	-----	-----	-----	-----
7369	SMITH	CLERK	7902	17-DEC-80	500	800	20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-DEC-82	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	12-JAN-83	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

DEPT TABLE



DEPTNO	DNAME	LOC
-----	-----	-----
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

Write the Nested Queries for the following queries.

1. List the details of the emps whose Salaries more than the employee BLAKE.
2. List the emps whose Jobs are same as ALLEN.
3. List the Empls whose Sal is same as FORD or SMITH in desc order of Names.
4. List the emps Whose Jobs are same as MILLER or Sal is more than ALLEN.
5. Find the highest paid employee of sales department.
6. List the employees who are senior to most recently hired employee working under king.
7. List the names of the emps who are getting the highest sal dept wise.
8. List the emps whose sal is equal to the average of max and minimum
9. List the emps who joined in the company on the same date.
10. Find out the emps who joined in the company before their Managers.

5. EXPERIMENT-5

Title: 5. Use of different SQL clauses and join

Objective: To understand the use of group by and having clause and execute the SQL commands using JOIN

1. Write the SQL Queries for the following queries (use emp_table and dept_table of Experiment 4).

1. List the Deptno where there are no emps.
2. List the No.of emp's and Avg salary within each department for each job.
3. Find the maximum average salary drawn for each job except for 'President'.
4. List the department details where at least two emps are working.
5. List the no. of emps in each department where the no. is more than 3.
6. List the names of the emps who are getting the highest sal dept wise.
7. List the Deptno and their average salaries for dept with the average salary less than the averages for all departments.



2. Execute the experiment 4 using sql join.

6. EXPERIMENT-6

Title: 6. To understand the concepts of Views.

Objective: Students will be able to implement the concept of views.

1. Create table of table name: **EMPLOYEES** and add 6 rows

Column Name	Data Type	Width	Attributes
Employee_id	Character	10	PK
First_Name	Character	30	NN
Last_Name	Character	30	NN
DOB	Date		
Salary	Number	25	NN
Department_id	Character	10	

2. Execute the following view related queries:

- 1) Create View of name emp_view and the column would be Employee_id, Last_Name, salary and department_id only.:
- 2) Insert values into view(remove the NOT NULL constraint and then insert values):
- 3) Modify, delete and drop operations are performed on view.:
- 4) Creates a view named salary_view. The view shows the employees in department 20 and their annual salary.

7. EXPERIMENT-7

Title: 7. To understand the concepts of Index.

Objective: Students will be able to implement the concept of index.

1. Execute the following index related queries:

- 1) Create an index of name employee_idx on EMPLOYEES with column Last_Name, Department_id
- 2) Find the ROWID for the above table and create a unique index on employee_id column of the EMPLOYEES.
- 3) Create a reverse index on employee_id column of the EMPLOYEES.
- 4) Create a unique and composite index on employee_id and check whether there is duplicity of tuples or not.
- 5) Create Function-based indexes defined on the SQL functions UPPER(column_name) or LOWER(column_name) to facilitate case-insensitive searches(on column Last_Name).
- 6) Drop the function based index on column Last_Name.

8. EXPERIMENT-8



Title: 8. To understand the concepts of Sequence.

Objective: Students will be able to implement the concept of sequence.

- 1) Create a sequence by name EMPID_SEQ starting with value 100 with an interval of 1.
- 2) Write a SQL command for finding the current and the next status of EMPID_SEQ.
- 3) Change the Cache value of the sequence EMPID_SEQ to 20 and maxvalue to 1000.
- 4) Insert values in employees table using sequences for employee_id column.
- 5) Drop sequence EMPID_SEQ.
- 6) Create a sequence called REVERSE to generate numbers in the descending order from 10000 to 1000 with a decrement of 5.

9. EXPERIMENT-9

Title: 9. To understand the concepts of PL/SQL programming.

Objective: Students will be able to implement the basic concepts of PL/SQL.

- 1) Write a PL/SQL code to accept the value of A, B & C display which is greater.
- 2) Using PL/SQL Statements create a simple loop that display message “Welcome to PL/SQL Programming” 20 times.
- 3) Write a PL/SQL code block to find the factorial of a number.
- 4) Write a PL/SQL program to generate Fibonacci series.
- 5) Write a PL/SQL code to find the sum of first N numbers

10. EXPERIMENT-10

Title: 10. To understand the concepts of function and procedure in PL/SQL.

Objective: Students will be able to implement the PL/SQL programs using function and procedure.

- 1) Implement the experiments of lab 9 using functions and procedures

11. EXPERIMENT-11

Title: 11. To understand the concepts of implicit and explicit cursor.

Objective: Students will be able to implement the concept of implicit and explicit cursor.

1. Using implicit cursor update the salary by an increase of 10% for all the records in EMPLOYEES table, and finally display how many records have been updated. If no records exist display the message “**No Change**”.
2. Using explicit cursor fetch the employee name, employee_id and salary of all the records from EMPLOYEES table.
3. Using explicit cursor Insert the records from EMPLOYEES table for the columns employee_id, Last_Name and salary for those records whose salary exceeds 2500 into a new table TEMP_EMP

12. EXPERIMENT-12



Title: 12. To understand the concepts of Trigger.

Objective: Students will be able to implement the concept of trigger.

CUSTOMER Table:

ID	NAME	AGE	ADDRESS	SALARY
1	Ramesh	32	Ahmedabad	2000.00
2	Khilan	25	Delhi	1500.00
3	Kaushik	23	Kota	2000.00
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
6	Komal	22	MP	4500.00

- 1) 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 values and new values.

H. SUGGESTED READINGS:

H1. TEXT BOOK:

1. Sql, Pl/Sql the programming language of Oracle, iii- edition, Ivan Bayross

I. GUIDELINES

Cell Phones and other Electronic Communication Devices: Cell phones and other electronic communication devices (such as Blackberries/Laptops) are not permitted in classes during Tests or the Mid/Final Examination. Such devices **MUST** be turned off in the class room.

E-Mail and online learning tool: Each student in the class should have an e-mail id and a pass word to access the LMS system regularly. Regularly, important information – Date of conducting class tests, guest lectures, via online learning tool. The best way to arrange meetings with us or ask specific questions is by email and prior appointment. All the assignments preferably should be uploaded on online learning tool. Various research papers/reference material will be mailed/uploaded on online learning platform time to time.

Attendance: Students are required to have **minimum attendance of 75%** in each subject. Students with less than said percentage shall **NOT** be allowed to appear in the end semester examination.

Passing criterion: Student has to score total (Performance & Viva voce & Quiz) 35 marks.

J. COURSE OUTCOME ASSESSMENT

To assess the fulfilment of course outcomes two different approaches have been decided. Degree of fulfillment of course outcomes will be assessed in different ways through direct assessment and indirect assessment. In Direct Assessment, it is measured through quizzes, tests, assignment, Mid-term and/or End-term examinations. It is suggested that each examination is designed in such a way that it can address one or two outcomes (depending upon the course completion). Indirect assessment is done through the student survey which needs to be designed by the faculty (sample format is given below) and it shall be conducted towards the end of course completion. The evaluation of the achievement of the Course Outcomes shall be done by analyzing the inputs received through Direct and Indirect Assessments and then corrective actions suggested for further improvement.

NAME:
ENROLLMENT NO:
SAP ID:
COURSE: B. Tech. 3 rd Semester
PROGRAM: CSE (GG)

Please rate the following aspects of course outcomes of Advanced Database Management Systems Lab
Use the scale 1-4*

Sr. No.	Course Outcome	1	2	3	4
1	CO1. Implement a database schema for a given problem-domain using DDL/DML/DCL commands.				
2	CO2. Use SQL queries in databases.				
3	CO3. Implement Cursors & Triggers using PL\SQL.				



Below Average



Good



Average



Very Good