COURSE PLAN

Department	: Data Science And Computer Applications
Course Name & code	**:** Database Management System Lab & MCA 4141
Semester & branch	**:** 1st Semester, M.C.A
Name of the faculty	:Vinayak Mantoor, Archana. H,Chithra K
No. of contact hours/week	**:** 3 hours/week

Continuous Evaluation	**60%**
• **Two Evaluations** • **Midterm Exam**	Record(6)+Execution(7)+Quiz(7) 20 marks **Max.Marks:60**
Lab Examination	**40%**
	Examination of 3 hours duration that includes questions based on: SQL+PL/SQL Max.Marks:40

INSTRUCTIONS TO STUDENTS

1. Students should be regular and come prepared for the lab practice.
2. In case a student misses a class, it is his/her responsibility to complete that missed exercise(s).
3. Students should bring the observation book daily for the lab.
4. They should implement the given query/program individually.
5. Students should clearly listen to the instructions given by the faculty.
6. Once the query/program gets executed, they should show the query/program and results to the instructors and copy the same in their observation book.
7. When copying down the query in the observation book the template to be followed is:
    a. Question No
    b. Question

c. Query statement
d. Output

8. When copying down the PL/SQL program in the observation book the template to be followed is:
    a. Program No
    b. Program title
    c. Program Code
    d. Output
9. Questions for lab tests and exam need not necessarily be limited to the questions in the manual, but could involve some variations and / or combinations of the questions.

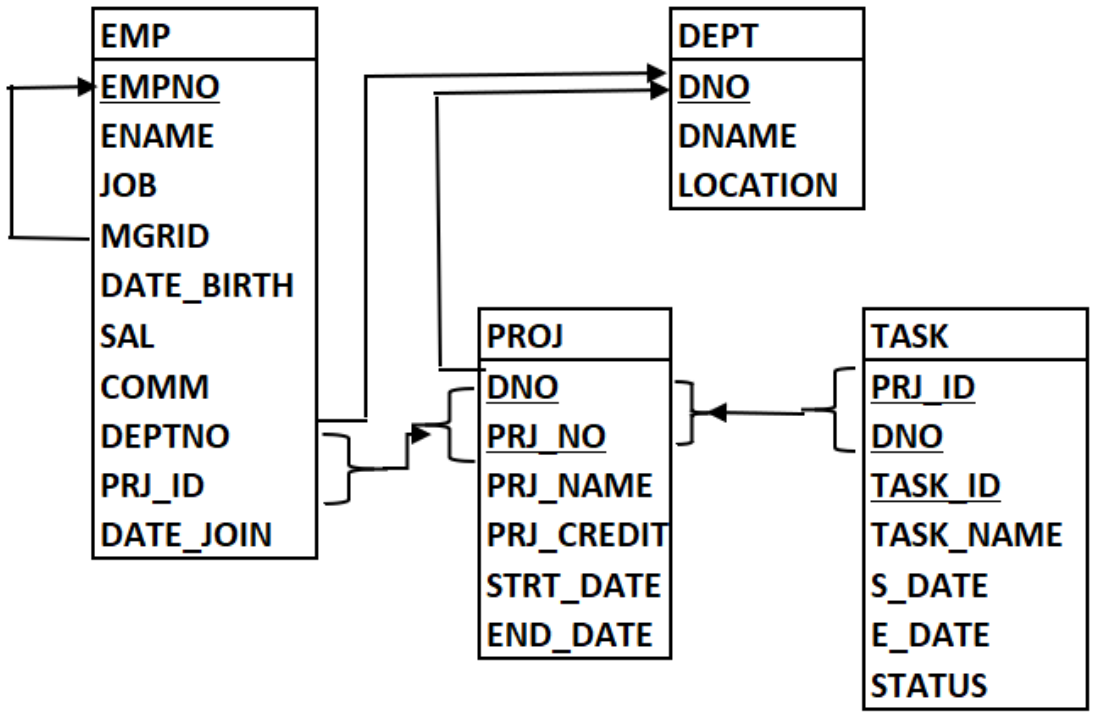
CONTENTS

Course Outcomes (COs)

At the end of this course, the student should be able to:

		No. of Contact Hours
C01:	Create and modify database objects	6
C02:	Manipulate the data in the database	3
C03:	Design queries to retrieve the data from database	9
C04:	Perform database operations by integrating procedural language constructs	9
CO5:	Design stored programs	6

SCHEMA DIAGRAM



Course Plan

L. No.	Topics	CO
L1, L2	**1.1** Create the tables with the following columns and constraints with given constraint names:	CO1

EMP

Attribute	Type	Constraint	Constraint Name
EMPNO	NUMBER(4)	Make this as primary key	
ENAME	VARCHAR2(10)		
JOB	VARCHAR2(9)	CLRK/MGR/A.MGR/GM/CEO,	JOB_CHK_Cons
MGRID	NUMBER(4)	References EMP	MGR_FK_EMPNO_Cons
DATE_BIRTH	DATE	Must be less than joining Date	DB_Less_DBJOIN_Cons
SAL	NUMBER(7,2)	More than 20000	SAL_20KMore_Cons
COMM	NUMBER(7,2)	DEFAULT 1000	
DEPTNO	VARCHAR2(3)	References DEPT and ON DELETE CASCADE	DEPTNO_FK_DEPT_Cons
DATE_JOIN	DATE		

DEPT

Attribute	Type	Constraint	Constraint Name
DNO	VARCHAR2(3)	UNIQUE and Starts from 'D'	DNO_UNQ_Cons and D-LikeDNO_Cons
DNAME	VARCHAR2(10)	Unique	DNAME_UNQ_Cons
LOCATION	VARCHAR2(9)	BNG/MNG/MUB/HYD/CHN	VALID_LOC_Cons

PROJ

Attribute	Type	Constraint	Constraint Name
DNO	VARCHAR2(3)	References DEPT ,NOT NULL	DNO_FK_DEPT_Cons
PRJ_NO	VARCHAR2(5)	Starts from 'P' , NOT NULL	P-LikePRJNO_Cons
PRJ_NAME	VARCHAR2(10)		
PRJ_CREDIT	NUMBER(2)	Range from 1 to 10	PRJ_CREDIT_RANGE_Cons
STRT_DATE	DATE		
END_DATE	DATE	END_DATE > START_DATE	ENDDATE_GRT_STRT_Cons

TASK

Attribute	Type	Constraint	Constraint_Name
TASK_ID	NUMBER(3)		
TASK_NAME	VARCHAR(100)		
PRJ_ID	F.Key	References PROJ ON DELETE SET NULL	
DNO			
S_DATE	DATE		
E_DATE	DATE	E_DATE>S_DATE	
STATUS	VARCHAR(50)		

2.1 Make the combination of **DNO and PRJ_NO** as primary key in the table PROJ.

L. No.	Topics	CO
	2.2 Add a column to EMP table named **PRJ_ID**. Add a foreign key constraint to EMP table on **(DeptNo, Proj_Id)** referencing **PROJ**. (Indicates an employee from which department is working on which project/s.) **2.3** Add constraints(VALID_EMPNO_Cons) to the EMP table to check the EMPNO >100. **2.4** Add a new column **Dept_Budget** column of size 7 digits to the DEPT table. **2.5** Add a new column **Prj_Fund** column of size 7 digits to the PROJ table. **2.6** Add constraint to **STATUS** column of TASK table to include values {NOT STARTED,IN PROGREES,COMPLETED} **2.7** In the table TASK, make Task_Id,DNO,Prj_ID as the primary key.	
L3	**Note:** Insert records into the following tables. • **Student can enter a valid value into the columns left blank, columns with *null* must be entered with NULL value only.** • **If some records cannot be inserted because of violation of constraints, then write the reason for violation in the lab book and choose a data satisfying the constraint and insert them into the table.** • **Insert 2 records of your own**. **3.1 EMP table data.** **3.1.1** MgRID values can't be inserted using INSERT, write the reason and solution.	**CO2**

Empno	Ename	Job	MgRID	Date_Birth	Sal	comm	Deptno	Prj_Id	Date join
100	Ravi	MGR	111	10-10-1985	32000		D1	P1	2-10-2001
102	Raviraj	CLRK	106	10-12-1980	24000		D1	P3	12-11-2000
111	Raghu	GM	150	10-12-1974	45000	15000	null	null	3-12-1985
150		CEO	null	10-12-1970	60000	30000	null	null	3-12-1990
103		A.CLRK	111	10-12-1980			D1	P1	2-10-2001
103		CLRK	111	2-10-1980			D1	P3	2-10-2002
125	Manu	A.MGR	150	10-12-1980			D4	P2	2-10-2002
104		CLERK	125	2-10-1980			D2	P1	2-10-2005
106		MGR	111	2-10-1986			D2		2-10-1985
123	Mahesh	CLRK	150	10-12-1974	25000		D3	P2	2-10-2002
108		CLRK	106	10-12-1970			D9		2-10-1985
103		CLRK	111	10-12-1980			D1	P3	2-10-2001
null		CLRK	106	10-12-1980	18000		D5		10-12-1980

3.2 DEPT table data.

L. No.	Topics	CO

DNO	DName	Location	Dept_Budget
D1	Marketing	CHN	500000
D2	Research	MNG	300000
D3	IT	BNG	400000
D4	HR	BGG	200000
D5	Accounts	BNG	500000
Null	Corporate	HYD	700000

3.3 PROJ table data.

Dno	Prj_No	Prj_Name	Prj_Credits	Prj_Fund
D1	P1		4	400000
D2	P1		2	200000
D3	P2		3	300000
D1	P3		5	500000
D4	P2		7	700000

Note: Perform following activity and write the observation.

3.4 TASK table Data

Task_Id	Task_Name	Prj_Id	DNo
1	Design Phase	P1	D1
2	Development Phase	P1	D2
1	Design Phase	P2	D3
2	Development Phase	P3	D1
3	Testing Phase	P3	D1

3.5 Run **COMMIT** command. Delete the employee records working on project P3 and confirm the result. Type **ROLLBACK** to restore the records back if records are deleted.

3.6 Run **COMMIT** command. Delete *Accounts* department from the DEPT table and confirm the result with reason. Type **ROLLBACK** to restore the records back if records are deleted.

3.7 Run **COMMIT** command. Delete records of employees with Empno 125 and working in project P2. Type **ROLLBACK** to restore the records back, if records are deleted.

3.8 Run **COMMIT** command .Delete all the tasks under Project P2. If deleted,type ROLLBACK to restore the records.

3.9 Update the DNO of first record in PROJ to D5.

3.10 Update the Job of employee with Empno 123 to MGR, salary to 35000 and his manager as 111.

3.11 Update the EMP table to set the default commission of all employees to Rs.10000/- who are working as managers.

L. No.	Topics		CO
L4	**4.1**	Display records of Employees who have salary more than 25000 or working in department D1 .	CO3
	4.2	List all employee with their names as "Employee Name" and their salaries as "SALARY", whose salary lies between 25200/- and 35200/- both inclusive	
	4.3	List the name of employees who is working at Locations (BNG,MUB,HYD) (using both OR , IN operator).	
	4.4	Display the records in the EMP table in the ascending order of Deptno and descending order of salary.	
	4.5	Display the task details for the project P3.	
	4.6	List out the employees who are not receiving commission.	
	4.7	Create a table **Manager** with columns Empno, Ename, Job, Deptno, Salary with structure and data copied from the *EMP* table.	
	4.8	List the Project Number, Project Name of all the projects handled by the department D2 and having project credits more than 5.	
	4.9	List all employees reporting to manager with *empno* 111.	
	4.10	Display name of employees whose 2^{nd} & 3^{rd} character is '**av**',name length is 6 and ends with '**j**'.	
	4.11	Display the task details for the task with the duration less than 1 week.	
	4.12	Find projects that have tasks starting after January 1, 2024.	
L5	**5.1**	List the minimum, maximum and average salaries and rename the column as min_sal, max_sal, avg_sal, total salary of the employees.	CO3
	5.2	List the Project names undertaken by *Marketing* Department.	
	5.3	Display the employees name in capital, lower, 1^{st} character only capital, number of characters and 3 characters from 2^{nd} position.	
	5.4	List the name of employees who are working under the manager 'Raghu'.	
	5.5	Display department name, Max salary and Min salary in each department.	
	5.6	Display number of employees working in each department and their department name.	
	5.7	List the employees whose experience is more than 5 years.	
	5.8	List the Employees who are born in the month of December and year 2000.	
	5.9	List the Departments with More Than Three Employees.	
	5.10	List the employees working under the project 'Drug Repositioning'	
	5.11	Write SQL query to illustrate ceiling, floor, truncate, and round functions on the value ***2.83***.	
	5.12	List the projects which have duration of more than 1 year.	
	5.13	List all employees along with the tasks they are associated with through their department's projects.	
	5.14	List the departments with no project.	
L6	**6.1**	List Job category and total salary paid for the each jobs category by the company.	CO3
	6.2	Display name of the department from which maximum number of employees are working on project P1.	
	6.3	Select name and job of employees working either in 'Marketing' or 'Research' department.	

L. No.	Topics	CO
	6.4 Display name and salary of employees whose salary is greater than minimum salary of the company. **6.5** Display Employees Working on the Most Projects. **6.6** Display the Deptno of the department that has highest average salary of the company. **6.7** List the name of departments which are working on more than 1 project. **6.8** Display how many employees joined after 15th of all months. **6.9** Write a query to list Employee number, name and Job of the employees who work in the same job as 'Mahesh'. **6.10** Create a View EMP_PRJ_VW to display records of employees of 'marketing' department and project in which they are working. **6.11** Display employee names and projects in which they are working using View EMP_PRJ_VW . **6.12** Calculate Total and Average Task Duration for Each Project **6.13** List out the number of employees joined in every month in ascending order. **6.14** Create an index on the columns (name and job) on EMP table.	
L7	**7.1** Write a PL/SQL block to find the sum of the digits of a given number. **7.2** Write a PL/SQL block to check an input string is palindrome or not palindrome. **7.3** Write a PL/SQL block to accept employee number and display Employee Name, Department name, salary of employees in the format – '*RAVI* works in *Marketing* department and draws *32000*/- as salary'. **7.4** Create a Table EMPSAL with fields-Empno, Empname, Sal, HRA, DA, Gross Salary, PF, Net Salary (assume appropriate datatype and size). Write a PL/SQL block to accept an employee number existing in EMP table and calculate HRA, DA, Gross Salary, PF, Net_Salary of that employee. Insert the empno, empname, Sal, HRA, DA, Gross Salary, PF, Net Salary into the table EMPSAL: Use the following formula to calculate salary components- HRA=50% of Sal DA=20% of Sal PF=12% of Sal. Gross_sal= Sal+ HRA+DA Net_Sal= Gross_sal-PF **7.5** Write a PL/SQL block to calculate the area of a circle for a value of radius varying from 3 to 7. Store the Radius and the corresponding values of calculated Area in an empty table named CIRCLE, consisting of two columns radius & area.	**CO4**

L. No.	Topics	CO
L8	**8.1** Write a cursor to display ENAME and SAL of all employees drawing salary more than 30000/-. **8.2** Write a cursor PL/SQL block to allocate incentives to employees and to the corresponding departments handling a project from the Prj_Fund reserved. According to the Prj_Credits a department is having on the Project, a portion of Prj_Fund is given to the department on that project. E.g. Credit is 2 , allocate 20%. Corresponding department has to share 70% of amount received among all employees who are working on that project equally. Remaining 30% is added to Dept_Budget. **8.3** Write a PL/SQL block using a cursor FOR LOOP to list all projects and their associated tasks. **8.4** Write a parameterized cursor to display employee with the user given parameters- job and deptno (Using Cursor for loop) **8.5** Write a parameterized cursor to display first two highest paid employees details (Name, Salary, Department Name) working on a project. The Prj_Id is the user given parameter.	CO4
L9	**9.1** Write a PL/SQL program to demonstrate predefined exceptions- INVALID_NUMBER, TOO_MANY_ROWS, NO-DATA_FOUND. Do following operations in the PL/SQL block and handle pre-defined exceptions correspondingly by displaying proper messages. - Convert ename into number using to_number( ) function - Display ename of employee corresponding to user entered empno. Handle exception if entered empno does not exists. - Display ename of employee corresponding to user entered deptno. Handle exception. **9.2** Write a PL/SQL block to accept, Principle, Interest rate and duration (in years) to calculate Interest to be paid. Handle the exceptions if Principle <=1000, interest rate <5, year <1 and display proper error message for each. **9.3** Write a PL/SQL block to accept employee number from user and display employee details such as Empno, Name, and Sal. Handle the exception raised through user defined messages – **(i)** If user entered a non–existing employee number. **(ii)** If the salary more than 25000/- If employee exists and salary is less than 25000/- then update that salary to 25000/-	CO4
L10	**10.1** Write a PL/SQL trigger to fire when there is an updation of salary of any employee and record the Empno, Dept. Name and Old Salary, date on which salary is modified and user who modified information in the table SAL_MOD (Empno, Dname, Old_Sal, Mod_Date, Modifier). **10.2** Write a trigger to be invoked when a department on a project updates Prj_fund by adding additional fund. Depending on project duration lapsed, display the message for additional fund for which department is eligible on that project. The project duration lapsed is calculated as difference between current_date (user entered) and start_date. Additional fund eligible is calculated according to the following criteria:	CO5

L. No.	Topics	CO
	If project duration lapsed is less than 50% then display- You are eligible for Rs. xxxx(xxxx is 70% of additional fund requested) else display- You are eligible for Rs.xxxx (xxxx is40% of additional fund requested). **10.3** Create a trigger that ensures an employee's salary cannot be reduced. If an update attempt is made to reduce the salary, the trigger should prevent it and raise an error.	
L11	**11.1** Write a procedure to calculate age and service experience of all employees and return these values to calling PL/SQL block and display. **11.2** Write a procedure to take department name as input to display project being handled by the department and name of the employees working under those projects belonging to the department. **11.3** Write a PL/SQL function to get the status of a project based on the completion status of its tasks. **11.4** Write a function to find total project fund on different projects that every department has received. In main program call function for every deptno fetched from Dept. Display following message format for every deptno in main program- (use cursor) ***Marketing*** department has received Rs. ***1600000***/- **11.5** Write a package containing procedure to find sum of salary of employees working in a given Department Name and a function to find number of employees working under a given Project Name.	**CO5**
L12	**END SEMESTER LAB EXAM**	

References

1. Ivan Bayross, "SQL, PL/SQL-The Programming Language of ORACLE", 4th Edition, BPB Publications,.
2. Satish Asnani, "Oracle Database 11g", PHI, 2010.
3. Scott Urman, "ORACLE – PL/SQL Programming", Oracle Press.

Submitted by: Vinayak Mantoor, Archana.H & Chithra K

(Signature of the faculty)

Date: 02/07/2024

Approved by: Dr. Radhika M Pai

(Signature of HOD)

Date: 02/07/2024