DATABASE MANAGEMENT SYSTEM LAB GUIDE

KCS 551

Lab 1 and Lab 2 – E R Diagram

Objective: To understand a given case study, notations used for ER Diagram and design the ER Diagram using a tool of following scenarios.

A Group of students (Max 2) will be assigned a case study (from Exercise -2 moderate level) to design the ER Diagram.

Student has to submit a file named as < Group No._Branch_Lab Group_Day1_2> with the complete solution.

Students have to submit the entities, relationships, constraints, details of relationship and constraints, individual diagrams of entities, relationships with constraints and a complete diagram in the file.

Exercise-1 (Basic Level)

Case Study 1:

Draw E-R diagrams to indicate the following relationships between entity set Operator and entity setMachine: - Each Machine can be operated by many Operators but each Operator can operate only one machine. An operator can operate many machine and each machine can be operated by many Operators.

Case Study 2:

An organization having a set of employees to execute a set of projects. Each employee may be workingonmore than one project, each project is managed by a manager and a manager is also one of the employees.

Case Study 3:

Preparation of time table of an Engineering College, catering for a number of Sections (Year/Branch/Section),a number of courses, a number of faculty members teaching the courses and a number of class rooms (ignorelabs).

Case Study 4:

Construct an E-R diagram for a car-insurance company whose customers own one or more cars each. Each car has associated with it zero to any number of recorded accidents.

Case Study 5:

Consider a university database for the scheduling of classrooms for final exams. This database could be modelled as the single entity set exam, with attributes course-name, section-number, room-number, and time. Alternatively, one or more additional entity sets could be defined, along with relationship sets to replace some of the attributes of the exam entity set, as

• Course with attributes name, department, and c-number
• Section with attributes s-number and enrolment, and dependent as a weak entity set on course
• Room with attributes r-number, capacity, and building

Exercise-2 (Moderate Level)

Entity Relationship Model

Case Study 1:

Draw the ER diagram of airline Reservation Systems with the following details:-

FLIGHT_SCHEDULE (**FLT_NO**, FROM_PLACE, TO_PLACE, ETD, ETA)
AIRCRAFT (**AC_NO**, AC_TYPE, CAPACITY)
CREW (**CREW_ID**, CREW_NAME, DESIGNATION)
FLIGHT (**FLT_NO, DATE, AC_NO**, ATD, ATA)
FLT_CREW (**FLT_NO**, **DATE**, **CREW_ID**)
TICKET (**TICKET_NO**, ISSUE_DATE, FARE, P_NAME, P_ADDR, P_TEL_NO)
RESERVATION (**TICKET_NO**, **FLT_NO**, **DATE**, CONFIRMED, SEAT_NO)
CANCELLATION (**TICKET_NO**, **FLT_NO**, **DATE**, VOUCHER_NO, C_DATE)
REFUND (**VOUCHER_NO**, AMOUNT)

Case Study 2:

Draw the ER diagram of a College Management System with the following details:-

a. A college contains many departments
b. Each department can offer any number of courses
c. Many instructors can work in a department
d. An instructor can work only in one department
e. For each department there is a Head
f. An instructor can be head of only one department
g. Each instructor can take any number of courses
h. A course can be taken by only one instructor
i. A student can enrol for any number of courses
j. Each course can have any number of students

Case Study 3:

A university registrar's office maintains data about the following entities: (a) courses, including number, title, credits, syllabus, and prerequisites; (b) course offerings, including course number, year, semester, section number, instructor(s), timings, and classroom; (c) students, including student-id, name, and program; and (d) instructors, including identification number, name, department, and title. Further, the enrollment of students in courses and grades awarded to students in each course they are enrolled for must be appropriately modeled. Construct an E-R diagram forthe registrar's office. Document all

assumptions that you make about the mapping constraints. and instructor. The entity set course-offering is a weak entity set dependent on course.
The assumptions made are:
a. A class meets only at one particular place and time. This E-R diagram cannot model a class meeting at different places at different times.
b. There is no guarantee that the database does not have two classes meeting at the same place and time.

Case Study 4:

Draw the Entity- Relationship Diagram (ERD) for the following scenario: A salesperson may manage many other salespeople. A salesperson is managed by only one salespeople. A salesperson can be an agent for many customers. A customer is managed by one salespeople. A customer can place many orders. An order can be placed by one customer. An order lists many inventory items. An inventory item may be listed on many orders. An inventory item is assembled from many parts. A part may be assembled into many inventory items. Many employees assemble an inventory item from many parts. A supplier supplies many parts. A part may be supplied by many suppliers.

Case Study 5:

Congratulations! You have gotten a job planning databases for the European Union. Your first on job assignment is to help the various countries maintain information about their inhabitants. Your model should capture the following information:
• In each country, there are provinces, which contain towns. There cannot be two provinces with the same name in a single country. Similarly, there cannot be two towns with the same name in a single province.
• People live in towns. Men and women work in a town. Children learn in a school in a town.
• A person can be a man, a woman, or a child, and has a first-name, last-name, id, and birthday. Children are any people under the age of 18.
• A man can be married to a woman (polygamy is not allowed, i.e., one man can be married only to one woman). Although the Pope strongly disapproves, divorce, and subsequent remarriage, is possible

For each marriage, store the date of the marriage and information about who are the children of the married couple. You should assume that the parents of a child were married at the time of his birth. Draw an entity relationship diagram to model the information described above. Remember to put edge constraints and participation constraints where needed. Underline the key attributes of each entity in the diagram. If you use the ISA relationship, state any covering and overlap constraints that hold. Make any necessary and logical assumptions. State any such assumptions clearly. If there are any constraints in the problem that could not be expressed in the diagram, state these clearly.

Case Study 6:

Suppose you are given the following requirements for a simple database for the National Hockey League (NHL):

- the NHL has many teams,
- each team has a name, a city, a coach, a captain, and a set of players,
- each player belongs to only one team,
- each player has a name, a position (such as left wing or goalie), a skill level, and a set of injury records,
- a team captain is also a player,
- a game is played between two teams (referred to as host_team and guest_team) and has a date (such as May 11th, 1999) and a score (such as 4 to 2).

Construct a clean and concise ER diagram. List your assumptions and clearly indicate the cardinality mappings as well as any role indicators in your ER diagram.

Case Study 7:

Assume we have the following application that models soccer teams, the games they play, and the players in each team. In the design, we want to capture the following:

- We have a set of teams, each team has an ID (unique identifier), name, main stadium, and to which city this team belongs.
- Each team has many players, and each player belongs to one team. Each player has a number (unique identifier), name, DoB, start year, and shirt number that he uses.
- Teams play matches, in each match there is a host team and a guest team. The match takes place in the stadium of the host team.
- For each match we need to keep track of the following:
    - The date on which the game is played
    - The final result of the match
    - The players participated in the match. For each player, how many goals he scored, whether or not he took yellow card, and whether or not he took red card.
    - During the match, one player may substitute another player. We want to capture this substitution and the time at which it took place.
- Each match has exactly three referees. For each referee we have an ID (unique identifier), name, DoB, years of experience. One referee is the main referee and the other two are assistant referee.

Case Study 8:

A university database contains information about professors (identified by social security number, or SSN) and courses (identified by course_id). Professors teach courses; each of the following situations concerns the Teaches relationship set. For each situation, draw an ER diagram that describes it (assuming that no further constraints hold).

- Professors can teach the same course in several semesters, and each offering must be recorded.
- Professors can teach the same course in several semesters, and only the most recent such offering needs to be recorded (assume this condition applies in all subsequent questions)

- Every professor must teach some course.
- Every professor teaches exactly one course.
- Every professor teaches exactly one course and every course must be taught by some professor.
- Now suppose that certain courses can be taught by the team of professors jointly, but it is possible that no one professor in a team can teach the course.

Case Study 9:

Consider the following information about the university data base

- Professors have an SSN, a name, an age, a rank and a research specialty.
- Projects have a project number, a sponsor name, a starting date and ending date and the budget.
- Graduate students have an SSN, a name, an age and a degree program (MS or PhD).
- Each project is managed by one professor (known as the project's principal investigator).
- Each project is work on by one or more professors (known as the project's coinvestigators).
- Professors can manage and /or work on multiple projects.
- Each project is work on by one or more graduate students (known as the project's research assistants)
- When graduate student work on a project, a professor must supervise their work on the project. Graduate students can work on multiple projects, in which case they will have supervisor for each one (potentially different).
- Departments have a department no., a department name and main office.
- Departments have a professor known as a (chairman) who runs the department.
- Professor's work in one or more departments, and for each department that they work in, a time percentage is associated with their job.
- Graduate students have one major department in which they are working on their degree.
- Each graduate student has another, more senior graduate student (known as a student advisor) who advises him or her on what courses to take.

Design and draw an ER diagram for the following collection of data. Use only the basic ER model here, i.e., entities relationships and attributes. Be sure to indicate any key and participation constraints.

Case Study 10:

A Company has several business units. Each business unit has multiple projects. Employees must be assigned to one business unit. One or more employees are assigned to a project, but an employee may be on vacation and not assigned to any projects. One of the assigned employees will be project manager for that project.

Case Study 11:

In a hospital there are different departments. Patients are treated in these departments by the doctors assigned to patients. Usually each patient is treated by a single doctor, but in rare cases they will have two or three. Healthcare assistants will also attend to patients; every department has many healthcare assistants. Each patient is required to take a variety of drugs during different parts of the day such as morning, afternoon and night.

Case Study 12:

A toy manufacturing company manufactures different types of toys. The company has several manufacturing plants. Each plant manufactures different types of toys. A customer can place the order for these toys. Each order may contain one or more toys. Each customer has multiple ship-to addresses. To promote the business, the company offers different schemes based on the order value.Survey and extend the business rules for your company to design a perfect ER model.

Case Study 13:

Draw an ER Diagram for the following scenario:

- There are several companies which provide DMAT Account for trading of shares
- Each company has different branches in different cities
- Each branch has many customers
- One customer should take only one account in a company
- One customer can take multiple accounts with different companies

Case Study 14:

Create an ERD for a car dealership. The dealership sells both new and used cars, and it operates a service facility. Base your design on the following business rules:

- A salesperson may sell many cars, but each car is sold by only one salesperson.
- A customer may buy many cars, but each car is bought by only one customer.
- A salesperson writes a single invoice for each car he or she sells.
- A customer gets an invoice for each car he or she buys.
- A customer may come in just to have his or her car serviced; that is, a customer need not buy a car to be classified as a customer.
- When a customer takes one or more cars in for repair or service, one service ticket is written for each car.
- The car dealership maintains a service history for each of the cars serviced. The service records are referenced by the car's serial number.
- A car brought in for service can be worked on by many mechanics, and each mechanic may work on many cars.
- A car that is serviced may or may not need parts (e.g., adjusting a carburetor or cleaning a fuel injector nozzle does not require providing new parts).

Case Study 15:

The following is a description of some data requirements for a chain of pharmacies. Draw the appropriate entity-relationship (E-R) diagram. Clearly show all cardinality constraints, cardinality limits, and existence dependencies.

(a) A pharmaceutical company manufactures one or more drugs, and each drug is manufactured and marketed by exactly one pharmaceutical company.

(b) Drugs are sold in pharmacies. Each pharmacy has a unique identification. Every pharmacy sells one or more drugs, but some pharmacies do not sell every drug.

(c) Drug sales must be recorded by prescription, which are kept as a record by the pharmacy. A prescription clearly identifies the drug, physician, and patient, as well as the date it is filled.

(d) Doctors prescribe drugs for patients. A doctor can prescribe one or more drugs for a patient and a patient can get one or more prescriptions, but a prescription is written by only one doctor.

(e) Pharmaceutical companies may have long-term contracts with pharmacies and a pharmacy can contract with zero, one, or more pharmaceutical companies. Each contract is uniquely identified by a contract number.

Case Study 16:

Suppose that you are designing a schema to record information about reality shows on TV. Your database needs to record the following information:

- For each reality show, its name, genre, basic_info and participants name. Any reality show has at least two or more participants.
- For each producer, the company name, company country. A show is produced by exactly one producer. And one producer produces exactly one show.
- For each television, its name, start year, head office. A television may broadcasts multiple shows. Each show is broadcasted by exactly one television.
- For each user, his/her username, password, and age. A user may rate multiple shows, and a show may be rated by multiple users. Each rating has a score of 0 to 10.

Draw an entity relationship diagram for this database.

Case Study 17:

Construct an ER Diagram for Company having following details:

• Company organized into DEPARTMENT. Each department has unique name and a particular employee who manages the department. Start date for the manager is recorded. Department may have several locations.

• A department controls a number of PROJECT. Projects have a unique name, number and a single location.

• Company's EMPLOYEE name, ssno, address, salary, sex and birth date are recorded. An employee is assigned to one department, but may work for several projects (not necessarily controlled by her dept). Number of hours/week an employee works on each project is recorded; the immediate supervisor for the employee.

• Employee's DEPENDENT are tracked for health insurance purposes (dependent name, birthdate, relationship to employee).

Case Study 18:

The following are the requirements for the Gym Fitness Database

• For each MEMBER we keep track of the unique MemdID, a well as Name, Zip, and the Date the membership was paid
• For each MEMBERSHIP type we keep track of the unique Mid, as well as MName and Price
• For each PASS CATEGORY we keep track of the unique PassCatID, as well as PCName and Price
• For each ONE DAY PASS we keep track of the unique PassID and Date
• For each MERCHANDISE item we keep track of the unique MrchID, as well as Name and Price
• For each sale TRANSACTION we keep track of the unique Tid and Date
• Each member pays for exactly one membership type; each membership type has at least one member but can have many members
• Each member can buy many day passes but does not have to buy any, each day pass was bought by exactly one member
• Each day pass belongs to exactly one pass category; a pass category can have many individual day passes issued for it but does not have to have any
• Each sale transaction involves exactly one member; each member can be involved in many sale transactions but does not have to be involved in any
• Each merchandise item is sold via at least one sale transaction but it can be sold via many sale transactions; each sale transaction involves at least one merchandise item but can involve many merchandise items
• Every time a merchandise item is sold via a sale transaction, we keep track of the quantity (how many instances of that particular merchandise item were sold via that particular sale transaction)
Your task is to create and ER Diagram based on these requirements.

Bonus Case Study:

You have been asked to develop a new Course Registration System for a college. The college wants a web-based system to replace its manual system. The college provides education in the various streams. In any stream, the entire graduation is divided into semesters.

The new system should allow the aspirants to submit their applications online. Once their applications have been approved and they have been admitted into the college in a Branch, the system should send an automatic welcome email along with the login id and the password

to the email addresses of the students. The email address is specified as part of the application. For students without any email address, the system shall print the welcome letters to be posted. Each Branch will have a set of courses, which are mandatory, and a certain number of elective courses.

At the beginning of the semester, the Head of the Department will create the necessary classes and do the allocation of lecturers to the classes for his department. The HOD may make changes in the allocation during the progress of the course. The system maintains a history of all the instructors who have conducted the classes throughout the semester.

The instructor will use the system to update the marks of the student (project, assignment, internal test marks and the semester end examination marks). The instructor will also mark the attendance of the students through the system.

The student can view his / her marks and attendance through the system.

In addition to the above, the system also keeps track of the residential status of the student. The student may be a hostelite or a day scholar. If he is a hostelite, the system will maintain his / her hostels name, room number and the fees pertaining to the same.

Lab 3 & Lab 4 – DDL Commands & Constraints

Objective: At the end of the assignments, participants will be able to understand basic DDL, Create table with constraints, Alter, Truncate, Drop and Rename

Student has to submit a file named as < Group No._Branch_Lab Group_Day3> with the complete solution.

Exercise -1:

1. Complete the GRADUATE CANDIDATE table instance chart. Credits is a foreign-key column referencing the requirements table.

Column Name	student_id	last_name	first_name	credits	graduation_date
Key Type					
Nulls/Unique					
FK Column					
Datatype	NUMBER	VARCHAR2	VARCHAR2	NUMBER	DATE
Length	6			3	

2. Write the syntax to create the grad_candidates table.

3. Confirm creation of the table.

4. Create a new table using grad_candidates with the following syntax:
   CREATE TABLE o_grad_candidates AS (SELECT * FROM grad_candidates);

5. Create a new table using a subquery. Name the new table your first name -- e.g., gaurav_table. Using a subquery, copy grad_candidates into gaurav_table.

6. In your o_grad_candidates table, enter a new column called “adm_date.” The datatype for the new column should be VARCHAR2. Set the DEFAULT for this column as SYSDATE.

7. In your o_grad_candidates table, increase the length of last_name column by 10 and remove the credits column.

8. Create a new column in the smith_table table called start_date. Use the TIMESTAMP WITH LOCAL TIME ZONE as the datatype.

9. Write syntax to change the name of credit column by grad_credit.

10. Insert 5 tuples in gaurav_table.

11. Truncate the gaurav_table table. Then do a SELECT * statement. Are the columns still there?

12. What the distinction is between TRUNCATE and DROP for tables?

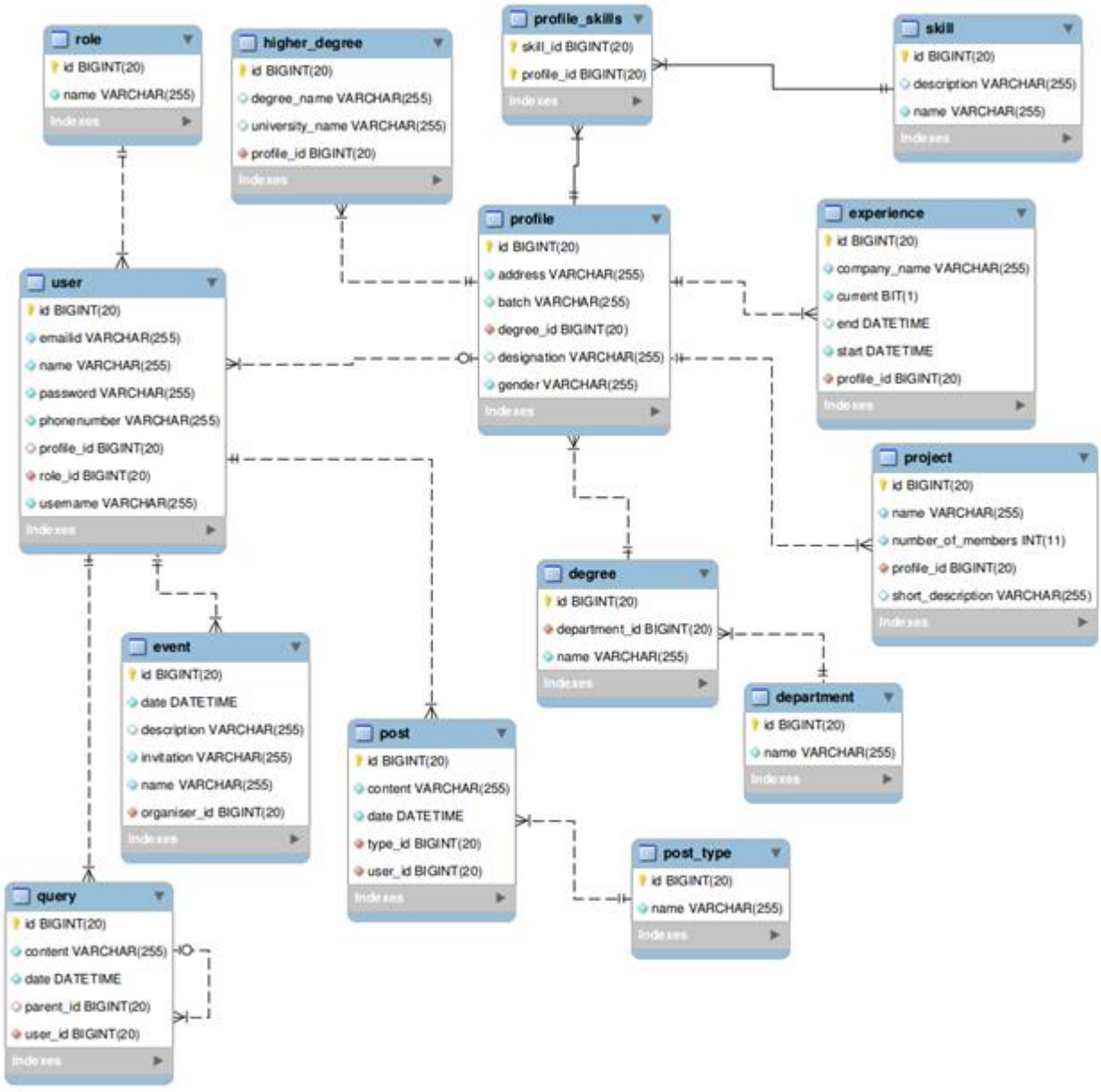
13. List the changes that can and cannot be made to a column.

14. Rename o_grad_candidates to n_grad_candidates.

Exercise -2:

1. Convert the ER Diagram into a neat and clean relational model.
2. Create all the tables using SQL commands.
3. Show all the tables with the constraints of your case study. Explain the reason of implementing that constraint on the required columns.

Bonus Exercise -3:



Q-1) Write a query to create profile_skills table.
Q-2) Write a query to create user table.
Q-3) Write a query to create role table.
Q-4) Write a query to create department table.
Q-5) Write a query to create degree table.
Q-6) Write a query to create profile table.
Q-7) Write a query to create higher_degree table.
Q-8) Write a query to create experience table.
Q-9) Write a query to create skill table.
Q-10) Write a query to add a new column named description of type varchar (255) to role table.
Q-11) Write a query to change the type of field description in the role table to varchar (500).
Q-12) Write a query to remove the column description from the role table.

Lab 5 -DML Commands

Objective: At the end of the assignments, participants will be able to understand basic DML, Insert rows using insert command, Update using update command and delete using delete command.

Exercise -1:

1. Insert 5 rows into the tables for the database created of your case study.

Exercise -2:

1. Create the tables mentioned below and insert the rows as shown. Please assume the datatype and constraints as required.

AD_ACADEMIC_SESSIONS:

ID	NAME
100	SPRING SESSION
200	FALL SESSION
300	SUMMER SESSION

AD_DEPARTMENTS:

ID	NAME	HEAD
10	ACCOUNTING	MARK SMITH
20	BIOLOGY	DAVE GOLD
30	COMPUTER SCIENCE	LINDA BROWN
40	LITERATURE	ANITA TAYLOR

AD_PARENT_INFORMATION: (Hint: must return to READ/WRITE status)

ID	PARENT1_FN	PARENT1_LN	PARENT2_FN	PARENT2_LN
600	NEIL	SMITH	DORIS	SMITH
610	WILLIAM	BEN	NITA	BEN
620	SEAN	TAYLOR	RHEA	TAYLOR
630	DAVE	CARMEN	CATHY	CARMEN
640	JOHN	AUDRY	JANE	AUDRY

2. Add 2 new rows in AD_ACADEMICS_SESSIONS table with name as “Summer Break Session” and “Winter Break Session”.
3. Update the name “Computer Science” with “Computer Science and Engineering” in AD_DEPARTMENTS table.
4. Update the PARENT1_LN as NULL for ID 620 in AD_PARENT_INFORMATION table.
5. Delete the 2 new rows added in question 2.

Lab 6 –DQL and Sorting Data

Objective: At the end of the assignments, participants will be able to understand basics of DQL commands, Select, Conditional retrieval, operators, pattern matching, order by clause.

Use the default schema of EMP Table & DEPT Table of the database and implement the listed queries:

Exercise -1: Queries based on Conditional Retrieval of Rows

1. List department names and location from the department table.

2. List the employees belonging to the department 20.

3. List the name and salary of the employees whose salary is more than 1000.

4. List the employee number and name of managers.

5. List the name of clerks working in the department 20.

6. List the names of analysts and salesmen.

7. List the details of the employees who have joined before the end of September 1981.

8. List the names of employees who are not managers.

Exercise -2: Special Operators IN and BETWEEN

1. List the name of the employee whose employee numbers are 7369,7521,7839,7934, 7788.

2. List the employee details not belonging to the department 10, 30 and 40.

3. List the employee name and salary, whose salary is between 1000 and 2000.

4. List employee names, who have joined before 30th June 81 and after December 81.

Exercise -3: DISTINCT Clause with SELECT

1. List the different jobs (designations) available in the EMP table.

Exercise -4: Working with NULL Values

1. List the employee names, who are not eligible for commission.

2. List the name of the employees and designation (job) of the employee, who does not report to anybody (manager is NULL).

3. List the employees not assigned to any department.

4. List the employees who are eligible for commission.

5. List the details of employees, whose salary is greater than 2000 and commission is NULL.

Exercise -5: Matching Pattern with Column

1. List the employees whose names start with an “S”.

2. List the employees names ending with an “S”.

3. List the names of employees whose names have exactly 5 Characters.

4. List the employee names having “I” as the second character**.**

Exercise -6: Using Expression with Column

1. List the name, salary and PF amount of all the employees (PF is calculated as 10% of salary).

2. List the names of employees, who are more than 2 years old in the organization.

Exercise -7: Ordering the Results of a query

ORDER BY clause to impose an order on the result of a query,
ORDER BY clause used with **SELECT** statement.
SYNTAX: SELECT [DISTINCT] <col list>| <exp> FROM table name WHERE cond ORDER BY col [ASC|DESC].
One or more column can be specified in ORDER BY clause.

1. List the empno, ename, sal in ascending order of salary.
2. List the employee name and hiredate in descending order of Hiredate.
3. List the employee name, salary, job and department No. in ascending order of Department No. in ascending order of Department No and then on descending descending order of salary.

Lab 7 –Subquery and Nested Query

Exercise -1:

To solve the below mentioned queries (1-10), please upload the SMS database script provided by your faculty.

1. Find the names of faculty who are working in the same department in which Gagan Kumar Verma is working.
2. Find the status of research project/s in which Ram Mohan Prasad is working.
3. Find the names of students who have registered for mini-project.
4. Find the names of faculty who are working in more than two research projects.
5. Find the number of students registered in each course of a department.
6. Create a report that displays the faculty id, last name, and salary of all faculty who earn more than the average salary. Sort the results in ascending order of salary.
7. Write a query that displays the faculty id and last name of all employees who are working in a department with any employee whose last name includes letter **t.**
8. Find the first name and salary of all faculty who are reporting to head of the department Ayush Giri.
9. Find the names of students who are taking courses of faculty Mr. Ram Mohan Prasad.
10. Find the department in which a faculty has joined most recently.

Exercise -2:

To solve the below mentioned queries (1-10), please create the below mentioned database.

depositor:

customer_id	account_number
C001	A001
C002	A002
C003	A010
C004	A033
C005	A012

borrower:

customer_id	loan_number
C001	L001
C003	L002
C005	L010
C006	L033

customer:

customer_id	customer_name	customer_city
C001	Korth	NOIDA
C002	Sudarshan	Ghaziabad
C003	Navathe	NOIDA
C004	Leon	Lucknow
C005	Sudarshan	Ghaziabad
C006	C.J.Date	Greater NOIDA

account:

account_number	branch_name	balance
A001	NOIDA MAIN	20000
A002	NOIDA MAIN	30000
A010	MOHAN NAGAR	15000
A012	GREATER NOIDA	40000
A033	TILAK NAGAR	50000

branch:

branch_name	branch_city	assets
NOIDA MAIN	NOIDA	200000
MOHAN NAGAR	Ghaziabad	500000
GREATER NOIDA	NOIDA	700000
TILAK NAGAR	Lucknow	400000

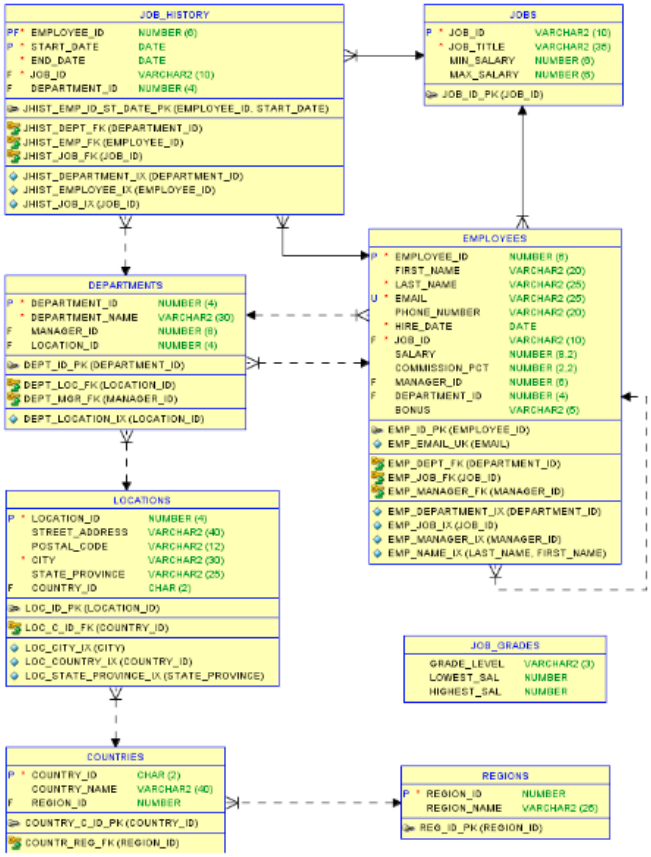
1. Find customer ids of those customers who are borrower from the banks and who appear in the list of account holders.
2. Find those customer names who are borrower.
3. Find the name of the customers who have a loan from the bank, but do not have an account at the bank. (Hint: use NOT IN)
4. Get the Customer Id and name of those customers who have both account and loan from the bank.
5. Get Branch Name of the branch having highest average balance amongst all branches.
6. Find the names of all branches that have assets greater than those of at least one branch located in NOIDA. Use some
7. Find the names of all branches that have assets greater than that of each branch located in NOIDA. (Use All)
8. Get the names of the customers who have account in each branch located in Noida.

Lab 8 – Joins using multiple tables

To solve the below mentioned queries, please upload the employee_department script provided by your faculty.

To understand the employees/departments database, please refer below mentioned table design

employees/departments Table Design



Exercise -1: CROSS-JOIN and NATURAL JOIN

1. Create a cross-join that displays the last name and department name from the employees and departments tables.
2. Create a query that uses a natural join to join the departments table and the locations table. Display the department id, department name, location id, and city.
3. Create a query that uses a natural join to join the departments table and the locations table. Restrict the output to only department IDs of 20 and 50. Display the department id, department name, location id, and city.

Exercise -2: INNER JOIN

1. Join the database locations and departments table using the location_id column. Limit the results to location 1400 only.
2. Display the city, department name, location ID, and department ID for departments 10, 20, and 30 for the city of Seattle.
3. Display country name, region ID, and region name for Americas.
4. Write a statement joining the employees and jobs tables. Display the first and last names, hire date, job id, job title, and maximum salary. Limit the query to those employees who are in jobs that can earn more than $12,000.

5. Display job title, employee first name, last name, and email for all employees who are stock clerks.
6. Write a statement that displays the employee ID, first name, last name, manager ID, manager first name, and manager last name for every employee in the employees table. Hint: this is a self-join.
7. Display the location ID, city, and department name for all Canadian locations.
8. Query and display manager ID, department ID, department name, first name, and last name for all employees in departments 80, 90, 110, and 190.
9. Display employee ID, last name, department ID, department name, and hire date for those employees whose hire date was June 7, 1994.

Exercise – 3: OUTER JOIN

1. Return the first name, last name, and department name for all employees including those employees not assigned to a department.
2. Return the first name, last name, and department name for all employees including those departments that do not have an employee assigned to them.
3. Return the first name, last name, and department name for all employees including those departments that do not have an employee assigned to them and those employees not assigned to a department.

Exercise – 4: SELF JOIN

1. Display the employee's last name and employee number along with the manager's last name and manager number. Label the columns: Employee, Emp#, Manager, and Mgr#, respectively.
2. Modify question 1 to display all employees and their managers, even if the employee does not have a manager. Order the list alphabetically by the last name of the employee.
3. Display the names and hire dates for all employees who were hired before their managers, along with their managers' names and hire dates. Label the columns Employee, Emp Hired, Manager and Mgr Hired, respectively.

Lab 9 – Restricting Data Using Group By Clause

Exercise -1:

Write the answers of following questions in practical file

1 What is the difference between WHERE clause and HAVING clause?
2 Why do we need to use the same columns that are selected in the SELECT list in the GROUP BY Clause?
3 Why do we need to have columns that are not part of GROUP BY Clause as part of aggregate function?
4 Will fixing the error by adding the columns to Group By clause, result in correct output?
5 In SQL, what's the difference between the having clause and the group by statement?

Exercise -2:

Solve the following queries on the basis of SMS database.

1. Write a query in SQL to find the number of faculty in each department along with the department id.
2. Write a query in SQL to find the sum of the allotment of salary amount of all departments.
3. Write a SQL query to display the average salary amount in each department along with their id.
4. Write a SQL query to find the number of faculty getting salary more than or equal to Rs.60000.
5. Write a SQL query to find the number of faculty teaching a course and number of students registered in that course.
6. Write a SQL query to find the number of faculty of each department who are working in any research project.
7. Write a SQL query to find the first name, faculty id and department of all faculty who are working in more than two research projects.
8. Write a SQL query to display the name and id of faculty who has maximum experience.
9. Write a SQL query to find the first name and department of all faculty who have completed any research project.
10. Write a SQL query to find the student names and their departments who have registered in mini projects of Artificial Intelligence domain.

Bonus Exercise – 3:

Solve the following queries on the basis of schema given.

SCHEMA: S_ORD (CUS_ID, DATE_ORDER, DATE_SHIPMENT, SP_NO, TOTAL, PAYMENT)

CUS	DATE_ORDE	DATE_SHIP SAL	TOTAL	PAYMEN
100	31-AUG-92	10-SEP-92 11	601100	CREDIT
101	31-AUG-92	15-SEP-92 14	8056.6	CREDIT
102	01-SEP-92	08-SEP-92 15	8335	CREDIT
103	02-SEP-92	22-SEP-92 15	377	CASH
104	03-SEP-92	23-SEP-92 15	32430	CREDIT
105	04-SEP-92	18-SEP-92 11	2722.24	CREDIT
106	07-SEP-92	15-SEP-92 12	15634	CREDIT
107	07-SEP-92	21-SEP-92 15	142171	CREDIT
108	07-SEP-92	10-SEP-92 13	149570	CREDIT
109	08-SEP-92	28-SEP-92 11	1020935	CREDIT
110	09-SEP-92	21-SEP-92 11	1539.13	CASH
111	09-SEP-92	21-SEP-92-11	2770	CASH
97	28-AUG-92	17-SEP-92 12	84000	CREDIT.
98	31-AUG-92	10-SEP-92 14	595	CASH
99	31-AUG-92	18-SEP-92 14	7707	CREDIT
112	31-AUG-92	10-SEP-92 12	550	CREDIT

1. Write SQL query that displays the total and average payments of all the credit orders.
2. Write SQL query that displays the total and average payments grouped by type of payment.
3. How many order dates are represented compared to the total number of orders?
4. How many customers and sales representative are represented compared to the total number of orders?
5. Write SQL query that displays the lowest and highest payments of all the orders.
6. What is the average amount of the order for each sales representative?
7. Write an SQL query to display the order dates and how many orders were on each date.
8. Write SQL query to display the order amount by payment type for each sales representative
9. Query to display the highest and lowest order for each order date where more than one order was placed.
10. SQL query to display the average order for each order date where more than one order was placed and the average order is greater than 1000. Display them in order of average order.
11. Display the customer number with more than one order. Arrange alphabetically by customer id.