

Lumbini City College

(Affiliated to Tribhuvan University)

Tilottama-04 Rupandehi



LAB ASSIGNMENT (2080)

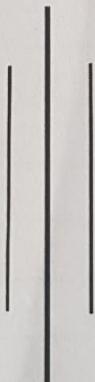
Course code: CACS 255

Course Title: Database Management System

Faculty Of Humanities and Social Science

Bachelor In Computer Application

Fourth Semester



Submitted To:

Mr. Kamal Bahadur Thapa

Submitted By:

Gaurav Karki

Table of Content

S.N.	Page	Topics	Date	Signature
01	1.	Lab Manual 1 'Movie'		
	1.1 - 1.3	a. DDL	2080/08/18	✓ JD
	1.4 - 1.5	b. DML	2080/08/19	
02	2.	Lab Manual 2 'Clinic'		
	2.1 - 2.2	a. DDL	2080/08/24	✓ JD
	2.3 - 2.4	b. DML	2080/08/24	
	2.5 - 2.6	c. DQL	2080/09/12	
03.	3.	Lab Manual 3 'Institute'		
	3.1 - 3.3	a. DDL	2080/08/24	✓ JD
	3.4 - 3.6	b. DML	2080/08/24	
	3.7 - 3.8	c. DQL	2080/09/11	
04.	4.	Lab Manual 4 'Hospital'		
	4.1 - 4.3	a. DDL	2080/11/06	✓ JD
	4.4 -	b. DML	2080/11/08	
	4.5 - 4.6	c. DQL	2080/11/09	

Name of subject teacher:

Kamal Bahadur Thapa

Final Remark

Final Remark

Lab Manual 1
Title: Movie
as Data Definition Language

Q.1. Make database cinema.

ans. sql > CREATE DATABASE cinema

Q.2. Create tables

actor (Aid, Name, gender, DOB)

director (Did, Name, gender, DOB)

movies (Mid, Mtitle, Myear, MLanguage, DID, rating)

casting (Mid, Aid, Roles)

myfav (myid, movie, myear)

ans.

create table actor(
Aid int primary key,
Name varchar(30),
gender varchar(30),
DOB date
);

create table director(
Did int primary key,
Name varchar(30),
gender varchar(30),
DOB date
);

```
create table movies (
    Mid int primary key,
    Mtitle varchar (30),
    Myear year,
    Mlanguage varchar (30),
    DID int,
    rating int
);
```

```
create table casting (
    Mid int references movies (Mid),
    Aid int references actor (Aid),
    Roles varchar (30),
    primary key (Mid, Aid)
);
```

```
create table mytav(
    myid int primary key,
    movie varchar (30),
    myyear year
);
```



Q.3. Add attributes mlanguage, budget, country and movie type in mytav table.

ans. alter table mytav
 add mlanguage varchar (30),
 add budget int,
 add country varchar (30),
 add movietype varchar (30);

Q.4. Change attribute name country to desh in mytav table.

ans. alter table mytav
 CHANGE COLUMN country desh varchar (30);

Q.5. Change table name myfav to merochoice.

ans. alter table myfav rename merochoice

Q.6. Change myid olddatatype to integer.

ans. alter table merochoice
modify myid int(4);

Q.7. Remove attribute budget and movie type from
merochoice table.

ans. alter table merochoice
drop budget,
drop movie type;

Q.8. Remove table merochoice

ans. drop table merochoice

✓
Wd

b) Data Manipulation Language

Q1. Insert 5 records in each table.

Ans: INSERT INTO actor

```
VALUES (1, "Gaurav", "m", "2005/01/11"),  
(2, "Aayush", "m", "2006/02/13"),  
(3, "Manisha", "f", "2003/05/19"),  
(4, "Bishal", "m", "2002/08/30"),  
(5, "Asmi", "f", "2007/04/03");
```

INSERT INTO directors

```
VALUES (1, "Bijay", "m", "2000/09/11"),  
(2, "Samir", "m", "1998/07/25"),  
(3, "Binita", "f", "2002/06/11"),  
(4, "Rita", "f", "1990/08/15"),  
(5, "Shyam", "m", "1999/11/06");
```

INSERT INTO movies

```
VALUES (1, "Race", "2009", "English", 89, 3),  
(2, "Dama", "2004", "Nepali", 34, 4),  
(3, "Lucy", "2012", "English", 20, 3),  
(4, "Stars", "2018", "Korean", 40, 3),  
(5, "Twins", "2016", "Chinese", 30, 4);
```

INSERT INTO casting

```
VALUES (1, 1, "Leadactor"),  
(2, 2, "Son"),  
(3, 3, "Leadactress"),  
(4, 4, "Antagonist"),  
(5, 5, "Leadactress");
```

```
INSERT INTO myfavmerochoice
VALUES (1, "Sahar", "2006"),
       (2, "Gida", "2004"),
       (3, "Sholay", "1996"),
       (4, "Avenger", "2002"),
       (5, "Muglan", "2002");
```

Q.2. Change the actor name to SHARUKH KHAN and gender to male of Mid 4.

Ans. UPDATE actor
 SET name = "SHARUKH KHAN", gender = "m"
 WHERE aid = 1;

Q.3. Deduct rating of all movies by 1.

Ans. UPDATE movies
 SET rating = rating - 1;

Q.4. Make the mlanguage of Mid 5 NULL.

Ans. UPDATE movies
 SET mlanguage = NULL
 WHERE Mid = 1;

Q.5. Delete all Korean movies released in year 2005.

Ans. DELETE from movies
 WHERE mlanguage = "Korean" && myear = "2005";

Q.6. Delete all actors whose name is RAJ or HIRA.

Ans. DELETE from actor
 WHERE name = "RAJ" or "HIRA";

Q.7. Delete all directors born in SEPTEMBER.

Ans. DELETE from director
 WHERE month(DOB) = 9;

Lab Manual 2

Title : Clinic

as Data Definition Language

Q.1. Create database clinic

ans. SQL > CREATE DATABASE clinic

Q.2. Create tables

patients (PatientID, firstName, lastName, gender, age, phone number, address)

doctors (doctorid, firstname, lastname, gender, dob, phone number, specialization)

appointments (appointmentid, patientid (Foreign key referencing patients), doctorid (Foreign key referencing doctors), appointment date, appointment time)

ans.

create table patients (
 patientid int primary key,
 firstName varchar (30),
 lastName varchar (30),
 gender varchar (5),
 age int,
 phone number varchar (30),
 address varchar (30));

```
create table doctors (
    doctorid int primary key,
    firstname varchar(30),
    lastname varchar(30),
    gender varchar(5),
    dob date,
    phonenumbe varchar(30),
    specialization varchar(30)
);
```

```
create table appointments (
    appointmentid int primary key,
    patientid int references patients(patientid),
    doctorid int references doctors(doctorid),
    appointmentdate date,
    appointmenttime time
);
```

Q.3. Add attributes email, fathername and mothername in patients table.

ans. alter table patients
 add email varchar(30),
 add fathername varchar(30),
 add mothername varchar(30);

Q.4. Change the attribute mothername to amakonaam in patients table.

ans. alter table patients

CHANGE COLUMN mothername amakonaam varchar(30);

Q.5. Remove fathername and amakonaam attribute from patients table.

ans. alter table patients
 DROP fathername,
 DROP amakonaam;

b) Data Manipulation Language

Q.1. Insert 5 records in each table.

ans. INSERT INTO patients VALUES

```
(1, "Gaurav", "Karki", "m", "18", "9812345678", "Btw1", "gk@gmail.com"),
(2, "Bishal", "Baral", "m", "19", "9812345687", "ktm", "bb@gmail.com"),
(3, "Anu", "Thapa", "f", "20", "9812345687", "Pokhara", "at@gmail.com"),
(4, "Asha", "Khati", "f", "21", "9812345786", "Dharan", "ak@gmail.com"),
(5, "Samir", "Neupane", "m", "25", "9812123434", "ktm", "sn@gmail.com");
```

INSERT INTO doctors VALUES

```
(1, "Rom", "Shah", "m", "06/06/2002", "9812347812", "Neurosurgeon"),
(2, "Shyam", "Thapa", "m", "09/11/1982", "9852621879", "Dentist"),
(3, "Geeta", "Pandey", "f", "07/12/1990", "9812312354", "Cardiologist"),
(4, "Padma", "Magar", "f", "04/02/1994", "9812314562", "Physician"),
(5, "Bikash", "Sharma", "m", "08/06/1998", "9854321023", "Allergist");
```

INSERT INTO appointments VALUES

```
(1, 1, 1, "12/14/2023", "10:45"),
(2, 2, 2, "12/16/2023"),
(3, 3, 3, "12/18/2023"),
(4, 4, 4, "12/20/2023"),
(5, 5, 5, "12/24/2023").
```

Q.2. Change the age of patients with id 5, 8 and 20 to 25.

ans. UPDATE patients

SET age = 25

WHERE patientid = 5 or 8 or 20;

Q.3. Remove all patients whose age is between 80 to 100.

ans. DELETE FROM patients

WHERE patientid > 80 AND age < 100;

Q.4. Change the name to RAJ TAMANG , gender to male and phonenumbe to 9877734435 of doctorid 7.

ans. UPDATE doctors

SET firstname = "RAJ" , lastname = "TAMANG" , gender = "male",
phonenumbe = "9877734435"

WHERE doctorid = 7;

✓
Key

c) Data Retrieval / Query Language

1. Show patients details whose firstname's 3rd character is M.

ans. Select * from patients where firstname LIKE "___M%";

2. Show doctor's firstname born in March 2000.

ans. Select firstname from doctors where date of birth >= "2000-3-1" and date of birth <= "2000-3-30";

3. Show doctor id & total appointments

ans. Select doctorid, count(*) from appointments where doctors.doctorid = appointments.appointmentid group by doctorid;

4. Show patient firstname & phone whose age is same as of patient SITA

ans. Select firstname, phone from patients where age = (Select age from patients where firstname = "SITA");

5. Show places whose patients average age is less than 35.

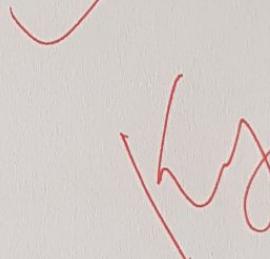
ans. Select address from patients group by address having avg(age) < 35;

6. show all doctors firstname whose specialization is not known.

ans. Select firstname from doctors where specialization is NULL;

7. Show doctor's firstname who have no appointment in 2023-12-12.

ans. Select distinct firstname from doctors, appointment where doctors.doctorid NOT IN (Select doctorid from appointment where appointmentdate = "2023-12-12");



Lab Manual 3
Title: Institute
as Data Definition Language

Q.1. Create database INSTITUTE.

ans. SQL > CREATE DATABASE INSTITUTE

Q.2. Create tables

courses (courseid, firstname, coursename, description, duration,
instructor, fee)

students (studentid, firstname, lastname, gender, dateofbirth, phone
number, email, address)

enrollments (enrollmentid, studentid (F.K), courseid (F.K),
enrollmentdate)

instructors (instructorid, firstname, lastname, phonenumbers,
email)

classrooms (classroomid, location, capacity)

schedule (scheduleid, courseid (F.K), instructorid (F.K), classroomid
(F.K), starttime, endtime, dayofweek)

ans.
create table courses (
courseid int primary key,
coursename varchar(30),
description varchar(30),
duration int,
instructor varchar(30),
fee int
);

```
create table students (
    studentid int primary key,
    firstname varchar(30),
    lastname varchar(30),
    gender varchar(30),
    dateofbirth date,
    phonenumber varchar(30),
    email varchar(30),
    address varchar(30)
);
```

```
create table enrollments (
    enrollmentid int primary key,
    studentid int references students(studentid),
    courseid int references courses(courseid),
    enrollmentdate date
);
```

```
create table instructors (
    instructorid int primary key,
    firstname varchar(30),
    lastname varchar(30),
    phonenumber varchar(30),
    email varchar(30)
);
```

```
create table classrooms (
    classroomid int primary key,
    location varchar(30),
    capacity int
);
```

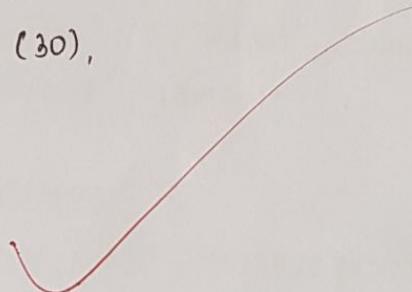
create table schedules (

scheduleid int primary key,
 courseid int references courses (courseid),
 instructorid int references instructors (instructorid),
 classroomid int references classrooms (classroomid),
 starttime time,
 endtime time,
 dayofweek varchar (30)

) ;

Q.3. Add attributes acfacility, waterfacility, roombreadth & roomheight in classrooms table.

ans. ALTER TABLE classrooms
 ADD acfacility varchar (30),
 ADD waterfacility varchar (30),
 ADD roombreadth int,
 ADD roomheight int;



Q.4. Change the attribute name acfacility to acORnot.

ans. ALTER TABLE classrooms
 CHANGE COLUMN acfacility acORnot varchar (30);

Q.5. Remove attributes roombreadth, roomheight & waterfacility from classrooms table.

ans. ALTER table classrooms
 DROP waterfacility,
 DROP roombreadth,
 DROP roomheight;

Q.6. change the attribute acorNOT datatype to Boolean.

ans. ALTER TABLE classrooms
 MODIFY acORnot boolean;



b> Data Manipulation Language

Q.1. Insert 5 records in each table.

Ans. INSERT INTO courses VALUES

```
(1, "BCA", "computer application", "4", "Manish", 12000),
(2, "BIM", "Information management", "3", "Gaurav", 14000),
(3, "BBS", "Business study", "4", "Santa", 18000),
(4, "BBA", "Business Administration", "5", "Rahul", 10000),
(5, "Bsc", "Science", "4", "Nirmal", 17000);
```

INSERT INTO students VALUES

```
(1, "Gaurav", "KC", "m", "01/11/2005", "9812435782", "gk@gmail.com", "Butwal"),
(2, "Samir", "Thapa", "m", "02/22/2003", "9814327802", "st@gmail.com", "Benganga"),
(3, "Rahul", "Bhusal", "m", "06/28/2002", "9814562682", "rb@gmail.com", "Ktm"),
(4, "Sima", "Bista", "f", "09/12/2003", "9814624203", "sb@gmail.com", "Dharan"),
(5, "Anju", "Poudel", "f", "08/11/2004", "9814145273", "ap@gmail.com", "pokhara");
```

INSERT INTO enrollments VALUES

```
(1, 1, 1, "12/16/2080"),
(2, 2, 2, "11/18/2079"),
(3, 3, 3, "01/23/2078"),
(4, 4, 4, "08/12/2079"),
(5, 5, 5, "06/18/2080");
```

Q.3. Change the attribute motherName to

INSERT INTO instructors VALUES {

```
(1, "Mahish", "Poudel", "9812341234", "mp@gmail.com"),
(2, "Gaurav", "Karki", "9843214321", "gk@gmail.com"),
(3, "Santa", "Chaudary", "9854321234", "sc@gmail.com"),
(4, "Rahul", "Tharu", "9802420864", "rt@gmail.com"),
(5, "Nirmal", "Shah", "9806428248", "ns@gmail.com");
```

INSERT INTO classrooms VALUES

```
(1, "Butwal", 30),
(2, "KTM", 20),
(3, "Pokhara", 40),
(4, "Birgunj", 20),
(5, "Butwal", 20);
```

INSERT INTO Schedules VALUES

```
(1, 1, 1, 1, "10:00", "11:00", "sunday"),
(2, 2, 2, 2, "11:00", "12:00", "monday"),
(3, 3, 3, 3, "10:00", "12:00", "tuesday"),
(4, 4, 4, 4, "1:00", "2:00", "friday"),
(5, 5, 5, 5, "9:00", "11:00", "saturday");
```

Q.2. Change the course name to JAVA PROGRAMMING and fee to 50,000 of courseid 4.

Ans. UPDATE courses

```
SET coursename = "JAVA PROGRAMMING", fee = 50000
WHERE courseid = 4;
```

Q.3. change the address of studentid 2 and 9 to BUTWAL.

Ans. UPDATE Student

```
SET address = "BUTWAL"
WHERE studentid = 2 or 5 or 9;
```

Q.4. Delete all classrooms with capacity less than 50.

Ans. DELETE FROM classrooms
WHERE capacity < 50;

Q.5. Delete all schedules of Saturday

Ans. DELETE FROM schedules
WHERE dayofweek = "saturday";

✓
Why

Page: 3. 9
Date: 11 Paush
2080.

c> Data Retrieval / Query Language

Q.1. Show course name and fee

Ans. `SELECT coursename, fee from courses;`

Q.2. Show girl students from palpa and nawalparasi.

Ans. `SELECT firstname, lastname from students
WHERE gender = "female" and (address = "palpa" or address
= "nawalparasi");`

Q.3. Show all student firstname enrolled in PTE

Ans. ~~`SELECT firstname from students, enrollment, courses
WHERE coursename = "PTE" and students.studentid = enrollment.
enrollmentid and courses.courseid = students.studentid;`~~

Q.4. Show students firstname & lastname whose first
name starts with S and lastname ends with R.

Ans. `SELECT firstname, lastname from students
WHERE firstname LIKE "S%" and lastname LIKE "%R";`

Q.5. Show course name and no. of students enrolled in
that course.

Ans. `Select coursename, count(*) from courses, enrollment
where courses.courseid = enrollment.enrollmentid
group by coursename;`

Q.6. Show highest & lowest fee.

Ans: SELECT max(fee), min(fee) from courses;

Q.7. Show coursename with lowest fees

Ans: SELECT coursename from courses

WHERE fee = (SELECT min(fee) from courses);

Q.8. Show all classroom id with more than 5 courses.

Ans: SELECT classroomid from schedules

GROUP by classroomid

HAVING courses(*) > 5;

Q.9. Show all unique addresses of students

Ans: SELECT DISTINCT address from students

Q.10. Show all course names taught betn 8AM to 2PM.

Ans: SELECT coursename from courses, schedules

WHERE courses.courseid = schedules.scheduleid and starttime

$\geq "08:00:00"$ and endtime $\leq "02:00:00"$

GROUP by coursename;

* WHERE courses.courseid = schedules.scheduleid and starttime

$\geq "08:00:00"$ and endtime $\leq "02:00:00"$

GROUP by coursename;

Lab Manual 4
Title : Hospital

1. Create database aspatal

ans. sql> CREATE DATABASE aspatal

2. Create tables

i. doctor (Did, Name, dob, gender, address)

ans. Create table doctor (
Did int primary key,
Name varchar (5),
dob date,
gender varchar (5)
address varchar (5));

ii. patient (Pid, Name, age, dob, gender, address)

ans. create table patient (
Pid int primary key,
Name varchar (5),
age int,
dob date,
gender varchar (5),
address varchar (5));

iii. appointment (Did, Pid, apdate, atime)

ans. create table appointment (

Did int references doctor(Did),
 Pid int references patient(Pid),
 apdate date,
 atime time,
 primary key (Did, Pid));

iv. hospital (Hid, Name, address, phone)

ans. create table hospital (

Hid int primary key,
 Name varchar(5),
 address varchar(5),
 phone varchar(5));

3. Insert 2 records in hospital.

ans. INSERT INTO hospital VALUES

(1, "Gaurav", "Butwal", "9800000000"),
 (2, "Samir", "Pokhara", "9812345678");

4. Delete table hospital.

ans. DROP table hospital

5. Insert attributes martialStatus, specification, fatherName, motherName, college in doctor table.

ans. ALTER Table hospital doctor

add martialStatus varchar(5),
 add specification varchar(5),
 add fatherName varchar(5),
 add motherName varchar(5)
 add college varchar(5);

6. Insert attribute fees in between 500 to 5000 only and doctorShift to Morning, Day and Night only in doctor table.

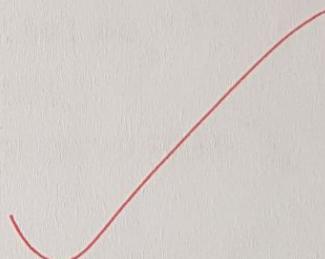
ans. ALTER table doctor

add fees int CHECK (fees > 500 and fees < 5000),
add doctorShift varchar (5) CHECK (doctorShift = "morning" or doctorShift = "day" or doctorShift = "night");

7. Remove attribute fatherName, motherName, college
and doctorShift from doctor table.

ans. ALTER table doctor

drop fatherName,
drop motherName,
drop college
drop doctorShift;



KWJ

Page: 4-4
Date: 8 Falgun
2080

b. Data Manipulation Language

1. Insert 1 record in each table.

Ans: SQL> INSERT INTO doctor VALUES
(1, "Gaurav", "2060-09-27", "male", "Butwal");
INSERT INTO patient VALUES
(1, "Samir", 20, "2060-08-04", "male", "Kopawa");
INSERT INTO appointment VALUES
(1, 1, "2080-11-07", "10:30:00");
INSERT INTO hospital VALUES
(1, "Rahul", "Butwal", "9800402030");

2. Change doctor name to ASHISH TAMANG, gender to MALE & address to GORKHA of doctorid=2.

Ans: UPDATE doctor set name = "ASHISH TAMANG",
gender = "MALE", address = "GORKHA" where Did = 2;

3. Change all appointment date after 24 February 2024 to next date.

Ans: UPDATE appointment SET apdate = apdate + 1 where
apdate > "2024-02-24";

4. Delete all appointment of palpal patient.

Ans: DELETE FROM appointment where pid IN (SELECT
pid from patient where address = "palpa");

5. Delete all appointment of June 2024.

Ans: DELETE * FROM appointment where month(apdate) = 6
and year(apdate) = 2024;

WJ

c. Data Query Language

1. Show all information of MALE patient.

ans. `SELECT * from patient where gender = "male";`

2. Show name and specialization of highest fees doctor.

ans. `SELECT name, specialization from doctor where fees = (Select max(fees) from doctor);`

3. Show patient & doctor name in which both are female.

ans. ~~`SELECT patient.name, doctor.name FROM patient, doctor, appointment where patient.Pid = appointment.Pid, and doctor.Pid = appointment.Did and gender = "female"; GROUP BY name;`~~

4. Show patients Name, age of patients whose name contains word "AD".

ans. `SELECT name, age FROM patient where name LIKE '%AD%' GROUP BY age;`

5. Show patient address and number of patient where there is at least 5 patient in month of January.

ans. `SELECT add.patient.address, count(Pid) FROM patient, appointment WHERE patient.Pid = appointment.Pid and month(update) = 1 GROUP BY address HAVING count(Pid) >= 5;`

6. Show doctor's name which is not in patient name.
ans. ~~SELECT DISTINCT doctor.name FROM doctor, patient, appointment
WHERE doctor.Did = appointment.Did and patient.Did = appointment.
Did. GROUP BY name;~~
7. Show only 1 patient name, age and address in decreasing
order according to age.
ans. ~~SELECT name, age, address FROM patient
ORDER BY age DESC;~~

✓ WD