

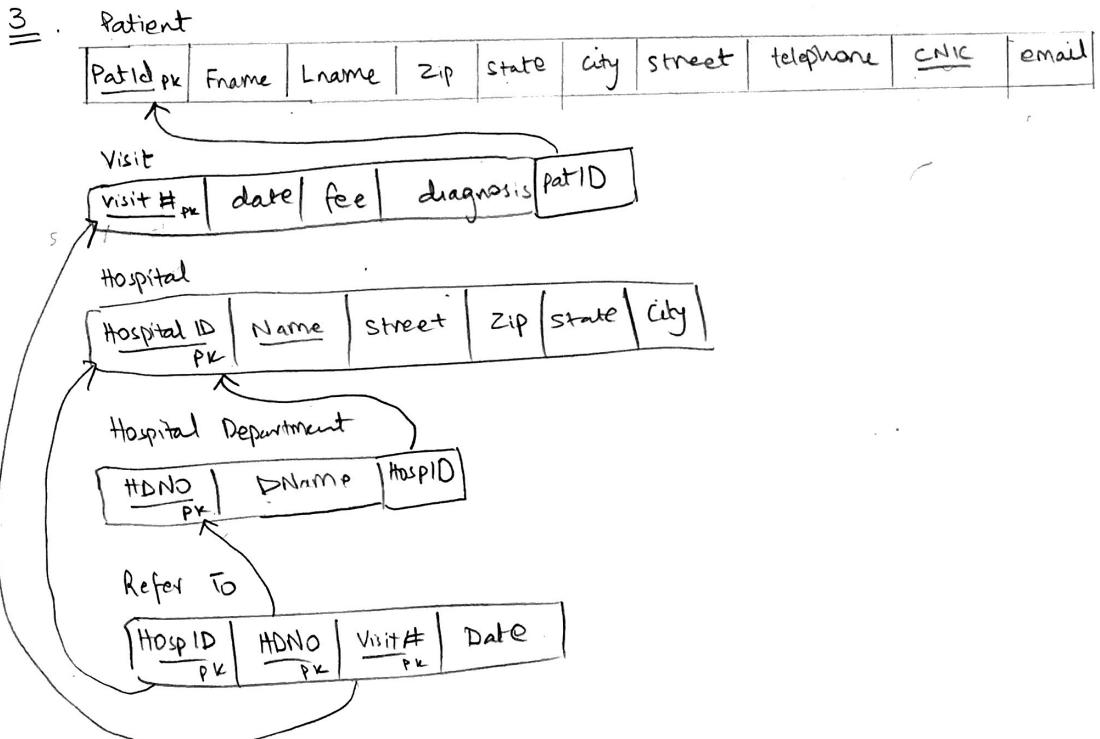
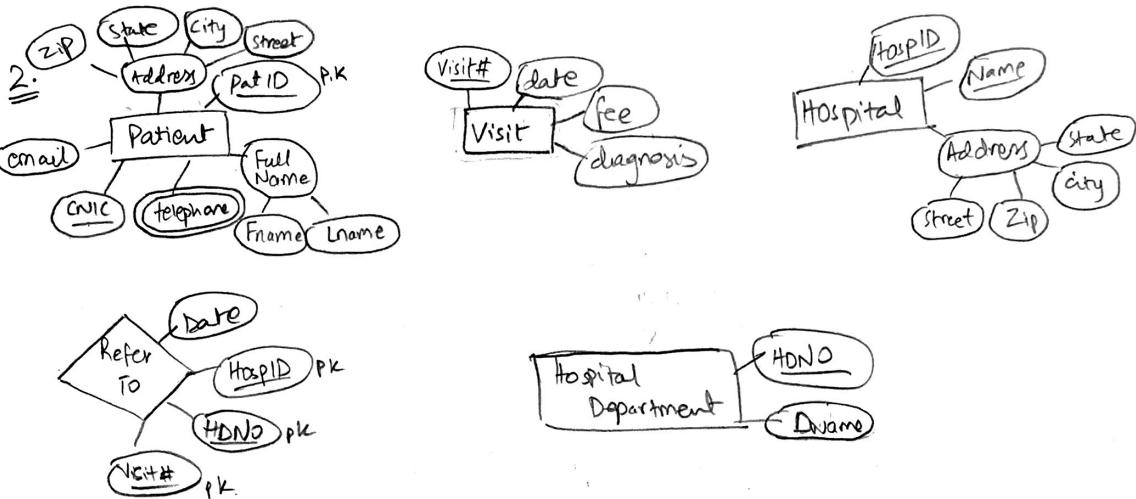
FINAL EXAM

①

Question #1

1. The ER Diagram shows a real life scenario of a Clinic's database.

A patient makes a visit to the clinic and is checked up by the doctor present there. During this visit the patient is referred to some hospital and their respective department to carry out a proper examination.



M.ADIL FAYYAZ
18I-0613 (C)
CS 210
Database systems
Adil Fayyaz

(2)

4. drop table Patient cascade constraints;

CREATE TABLE Patient (

PATID	VARCHAR2(10),
Fname	VARCHAR2(20),
LName	VARCHAR2(20),
Zip	VARCHAR2(10),
State	VARCHAR2(20),
City	varchar2(20),
Street	Var char2(20),
telephone	INT,
CNIC	VARCHAR2(13) UNIQUE,
email	varchar2(30) UNIQUE
CONSTRAINT Pat-PK PRIMARY KEY(PATID),	
CONSTRAINT check-cnic check(CNIC between 12 AND 14));	

drop table visits cascade constraints;

CREATE TABLE visits (

VISIT #	VARCHAR2(10),
date	DATE,
fee	INT,
diagnosis	VARCHAR2(20),
PAT ID	VARCHAR2(10)

CONSTRAINTS	V-fic FOREIGNKEY(PATID) references Patient (PATID),
CONSTRAINTS	v-pk PRIMARY KEY (VISIT#));

drop table Hospital cascade constraints;

Create table HOSPITAL (

Hospital ID	VARCHAR2(20),
Name	VARCHAR2(50) UNIQUE,
Street	Varchar2(10),
city	Varchar2(25),
State	Varchar2(30),
Zip	Varchar2(10),

CONSTRAINT	pk - H PRIMARY KEY (Hospital ID));
------------	------------------------------------

(3) Drop table Hospital Department cascade constraints;

Create table hospital Department(

HONO varchar2 (20),

Dname varchar2 (50),

HospitalID varchar2 (20),

CONSTRAINTS pk-Hd PRIMARY KEY (HONO),

CONSTRAINTS fk-Hd FOREIGN KEY (HospitalID) REFERENCES Hospital (HospitalID);

Drop table Refer to cascade constraints

Create table ReferTo(

HospitalID varchar2 (20),

HONO varchar2(20),

Visit# varchar2 (20),

Date DATE,

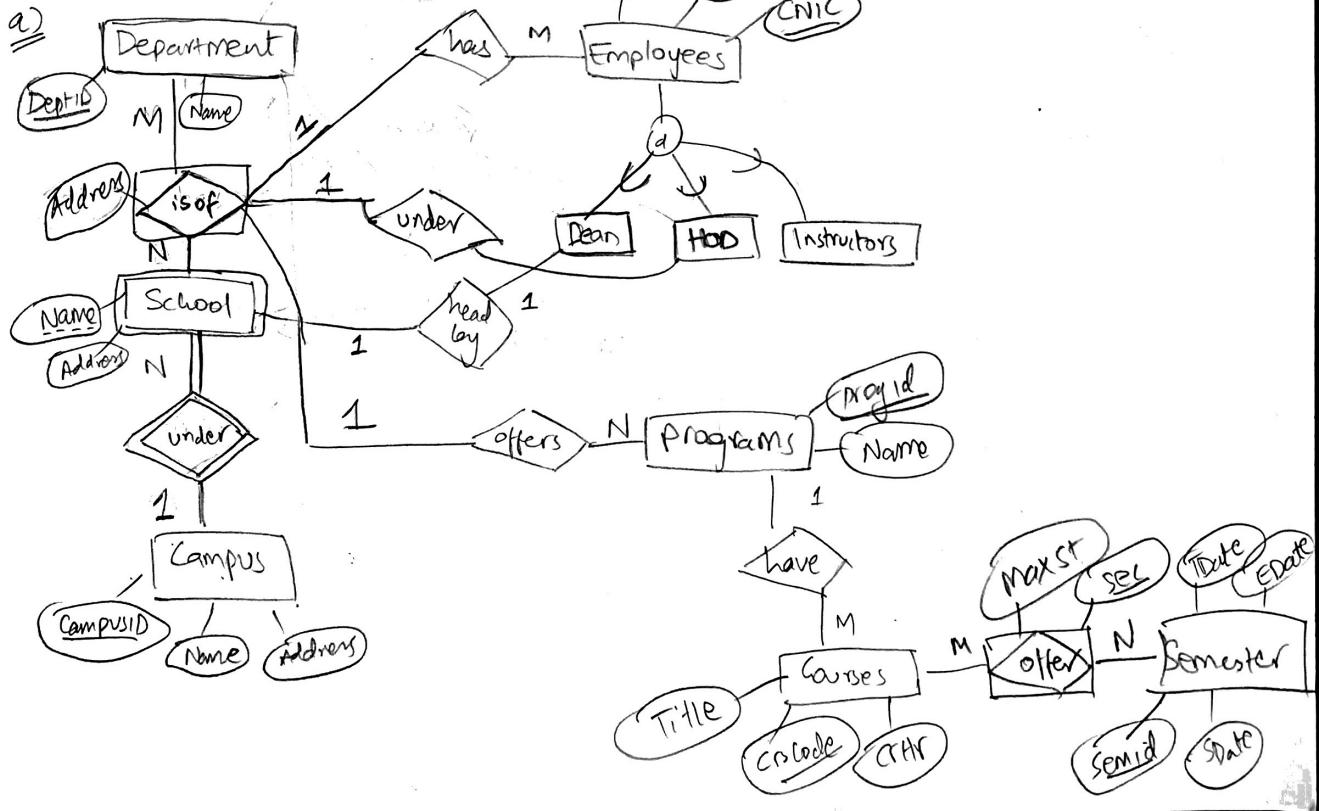
CONSTRAINTS pk-Ref PRIMARY KEY (HospitalID, HONO, Visit#),

CONSTRAINTS FK-Ref FOREIGN KEY (HospitalID) REFERENCES Hospital (HospitalID),

CONSTRAINTS FK-Ref2 FOREIGN KEY (HONO) REFERENCES HospitalDepartment (HONO),

CONSTRAINTS FK-Ref3 FOREIGN KEY (Visit#) REFERENCES Visit (Visit#));

Question 2



b) Campus

CampusID	Campus Name	Address
CP001	ISB Campus	H-11/4 ISB
CP002	Lhr Campus	Model Town Lhr
CP003	Khi Campus	Korangi Karachi
CP004	PWR Campus	PWR Town Peshawar.

School

CampusID	SchoolID	Name	Address	Head Name
CP001	SC001	Computing	NU Comp-FAST	Dr Atif
CP001	SC002	Electrical	NUElec.-FAST	Dr Waqas
CP002	SC001	Computing	NU Comp-FAST	Dr Atif
CP003	SC002	Computer	NU Comp-FAST	Dr Atif

⇒ Fix campusid references
Campus (CampusID).

Department

Dept ID	Name
D001	CompScience
D002	Engineering Electrical
D003	Business

ISOF	Dept ID	School ID	Campus ID	Address	HOD
D001	D001	SC001	CP001	ISB	Dr Ali
D001	D001	SC001	CP002	Khi	Dr Ahmed
D002	D002	SC002	CP001	ISB	Dr Asif
D002	D002	SC003	CP003	PWR	Dr Ali

Programs

Empld	Name	DOB	ENIC
E001	Ali	2/4/90	2344
E002	Ahmed	2/4/91	2944
E003	Aymen	7/8/98	2992

Prog ID	Name
P001	BS
P002	MS
P003	PHD

Courses

Course Code	Name	Credit Hrs
CS001	Blockchain	3
CS002	PF	2
CS003	DS	2

Semester

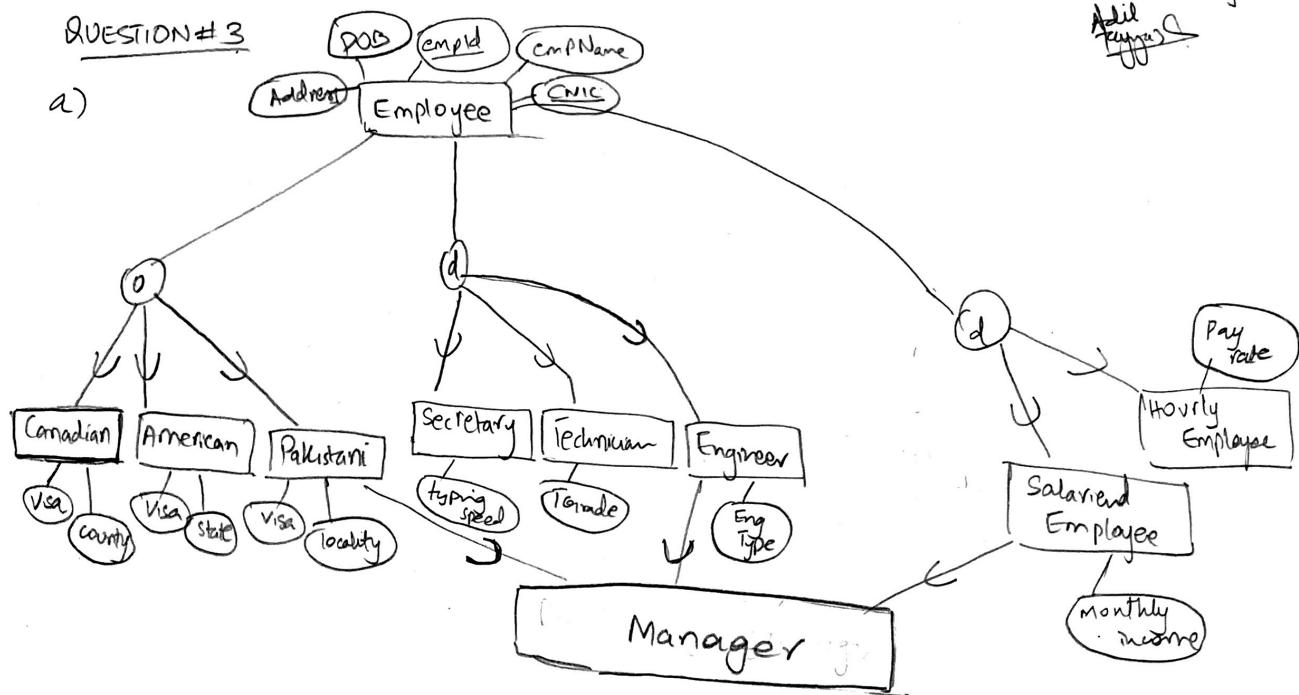
Sem ID	Sdate	Fdate
S01	2/7/01	3/9/01
S02	2/7/02	3/9/02
S03	2/7/03	3/9/03

Offer

CS Code	Sem ID	Sec	Max St
CS001	S01	A	20
CS002	S02	A	70
CS001	S01	B	100
CS001	S01	C	135

QUESTION # 3

a)



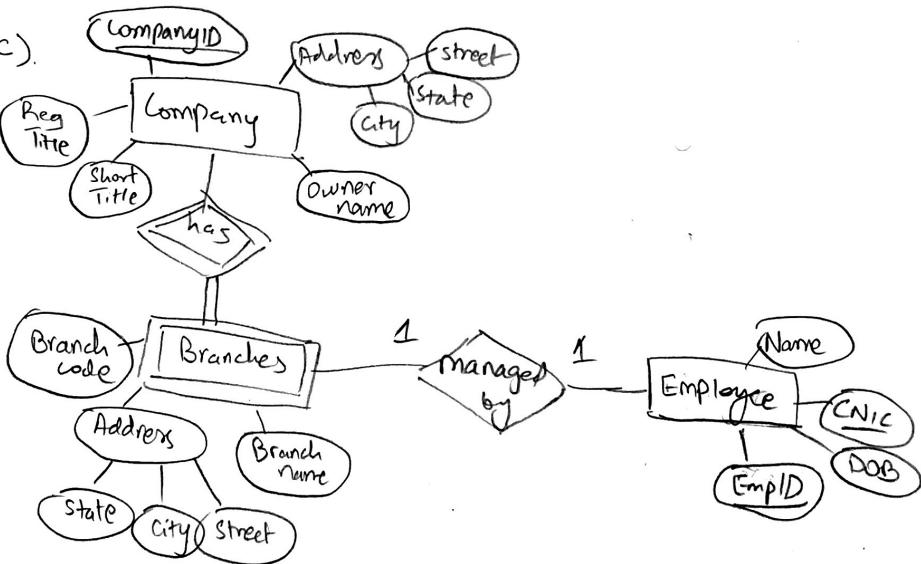
b).

Insurance (Insrid ^{PK} → type, description).

Payments (PaymentNo ^{PK}, Date, Insrid, ^{FK}Amount, Tax).

Insrid is Foreign Key in Payments table referencing primary key Insrid of Insurance table.

c).



Company

<u>CompanyID</u> ^{PK}	Street	state	City	owner name	RegTitle	shortTitle
--------------------------------	--------	-------	------	------------	----------	------------

Branches

<u>CompanyID</u> ^{PK}	<u>Sno</u> ^{PK}	Branchcode	Branchname	State	City	street	MgrID
--------------------------------	--------------------------	------------	------------	-------	------	--------	-------

Employee

<u>EmpID</u> ^{PK}	Name	CNIC	DOB
----------------------------	------	------	-----

d). prod \Rightarrow sold items \Rightarrow Invoice \Rightarrow Customer.

$\pi_{C.CustName, S.Qty, P.ProdName}$

$\ominus S.ProdID = P.ProdID$

$\pi_{S.ProdID, S.Qty, C.CustName}$

$\ominus I.Inv\# = S.Inv\#$

$\pi_{I.Inv\#, C.CustName}$

$\ominus C.WstID = I.WstID$

$\pi_{C.WstID, C.CustName}$

$\pi_{S.Inv\#, S.Qty}$

Product P

Sold Items S

$\pi_{I.CustID}$

$\ominus I.Inv = '66481'$

Invoice I

Customer C

(7)

Question #4

$S=100 \rightarrow D \Rightarrow \text{Dropped} = 3, \Delta = 2r.$

a). Log

$\langle 0, T_1, \text{start} \rangle$
 $\langle 1, T_1, \text{Read}, S \rangle$
 $\langle 2, T_2, \text{start} \rangle$
 $\langle 3, T_2, \text{Read}, S \rangle$
 $\langle 4, T_2, \text{Read}, \Delta \rangle$
 $\langle 5, T_2, \text{Read}, D \rangle$
 $\langle 9, T_2, S, 100, 100 \rangle$
 $\langle 10, T_2, \Delta, 2, 1 \rangle$
 $\langle 11, T_2, D, 3, 4 \rangle$
 $\langle 12, T_1, \text{Read}, D \rangle$
 $\langle 14, T_1, D, 4, 6 \rangle$
 $\langle 15, T_3, \text{start} \rangle$
 $\langle 16, T_4, \text{start} \rangle$
 $\langle 17, T_3, \text{Read} \rangle$
 $\langle 19, T_4, \text{Read} \rangle$
 $\langle 21, T_4, D, 6, 7 \rangle$
 $\langle 22, T_3, \Delta, 1, 0 \rangle$
 $\langle 23, T_1, \text{commit} \rangle$
 $\langle 24, T_4, \text{commit} \rangle$
 $\langle 25, T_3, \text{commit} \rangle$
 $\langle 26, T_2, \text{commit} \rangle$

- b). 1. T_1 and T_2 need to be undone, because they have started and not committed.
2. None, no transaction has committed yet.

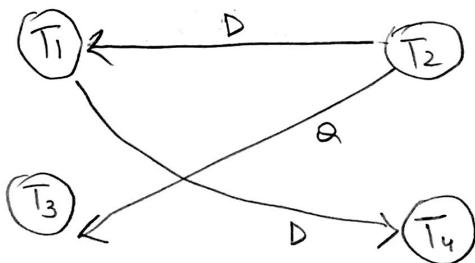
(8) c) 1. T_1, T_2 and T_3 need to be undone because they have started and not committed.

2. None, no transaction has committed yet.

d) 1. T_2 and T_3 need to be undone because they have not committed

2. T_1 and T_4 need to be redone because they have committed.

e) Graph



The serializability of a schedule is termed as a schedule of n transactions if it is equivalent to some schedule of the same n transactions. A precedence graph with a loop shows that it is not serializable, while one without loops shows that is serializable.

f) Transaction T_2 will be completed first as it uses D and Q both then T_1 will be completed followed by T_4 and lastly T_3 . This is the equivalent serial schedule

(9)

Question #5

M. ADIL FAYYAZ
RI-0613(CC)
CS219
Database Systems
Adil
Fayyaz

a). Select count(C.CUS-CODE).

from Customer C, Invoice I

where C.CUS-AREACODE = & Areacode

and C.Cus-code = I.Cuscode ;

b) Select C.CUS-LNAME, C.CUS-FNAME, C.CUS-PHONE
FROM Customer C

where C.CUS-FNAME LIKE 'O%'

ORDER BY C.CUS-LNAME ASC ;

c). Select Sum(C.CUS-BALANCE) Total Balance, min(C.CUS-BALANCE) Minimum Balance,
MAX(C.CUS-BALANCE) Maximum Balance, AVG(C.CUS-BALANCE) Average Balance.
From Customer ;

d). Select P.P_code, P.P-DESCRIPT
FROM PRODUCT P, VENDOR V

where P.V_code = V.V-Code (+) ;

Note : Using a Left outer join in the above Query.

e) Select I.INV-NUMBER, I.INV-DATE, SUM(L.LINE-PRICE * L.LINE-UNITS)
as INVOICE as INDATE
as \$ Total Value.
from INVOICE I, LINE L

where I.INV-NUMBER = L.INV-NUMBER

group by I.INV-NUMBER, I.INVDATE

having sum(L.LINE-PRICE * L.LINE-UNITS) > 100;