National University of Computer and Emerging Sciences, Lahore Campus

Course Name:	Database Systems	Course Code:	CS203
Program:	BS(CS)	Semester:	Fall 2018
Duration:	3 Hours	Total Marks:	70
Paper Date:	Mon 24-Dec-2018	Weight	45%
Section:	ALL	Page(s):	8
Exam Type:	Final		

Student : Name:_____ Roll No._____ Section:_____

Instructions/Notes:
1. Scratch sheet can be used for rough work however, all the questions and steps are to be shown on question paper. No extra/rough sheets should be submitted with question paper.
2. You will not get any credit if you do not show proper working, reasoning and steps as asked in question statements.

Q1. *(5 points)* Consider the relation *R (A, B, C, D, E)*, with FDs *{A → BC, C → D, E → D, BE → A}*. List all the possible keys of *R*. Show the intermediate steps of your derivation. Also Identify the best normal form that R satisfies.

ANSWER: *Keys are {AE} and {BE}. Best normal form is 1NF.*

Q2. *(5 points)* Consider a relation with schema *R (A, B, C, D, E, G)*, with FDs *F = {D → G, C → A, CDB → E, A → B}*.
Identify the best normal form that R satisfies (1NF, 2NF, 3NF, or BCNF). Justify your answer. If R is not in BCNF, decompose it into a set of BCNF relations and show your steps. Indicate which dependencies if any are not preserved by the BCNF decomposition.

ANSWER:
Key is {CD} and Highest Normal Form is 1NF. It violates 2NF due to PFD1 & PFD2.
BCNF relation schemas are R1(C, D, E) with FD3, R2(D, G) with FD1, R3(C, A) with FD2, R4(A, B) with FD4; All FDs are preserved.

Q3. *(5 points)* Consider the following relation and compute the closure of $\{A\}^+$, $\{B\}^+$, $\{C\}^+$, $\{D\}^+$, and $\{CD\}^+$. Show your work.

R

A	B	C	D
1	2	3	4
4	2	3	4
5	3	3	4

ANSWER:
$A^+=\{ABCD\}$, $B^+=\{BCD\}$, $C^+=\{CD\}$, $D^+=\{CD\}$, and $CD^+=\{CD\}$.

Q4. *(5 points)* Consider the following two sets of FDs. Check whether or not they are equivalent. Provide proper reason.

F1 = {A→C, B→C, C→AB} and *F2 = {A→BC, B→A, C→A}.*

ANSWER: *They are equivalent.*

Q5. *(15 points)* Consider the following tables from an internet movie database. Primary keys are underlined and the column *fid* is a foreign key in the table *ratings*. The ratings are on the scale 1-5, where 5 is best and 1 is worst.

Films

fid	title	yrrelease
f1	Title1	2016
f2	Title2	2018
f3	Title3	2018
f4	Title4	2017
f5	Title5	2017
f6	Title6	2018

Ratings

viewerid	fid	rating
v10	f1	1
v30	f1	1
v50	f1	5
v20	f3	1
v30	f3	5
v50	f3	1
v60	f3	5
v10	f5	1
v90	f5	3
v50	f5	1
v70	f5	5
v80	f5	1

Write the result of the following queries for the database state given above.

a. SELECT films.title, COUNT(rating) AS worstRatings FROM films JOIN ratings ON films.fid = ratings.fid
WHERE yrrelease > 2016 AND rating=1 GROUP BY film.title HAVING COUNT(rating) >= 2
ORDER BY COUNT(rating);

b. Result ← $\sigma_{\text{viewerid= v90 OR viewerid=NULL}}$ (films ⋈ ratings)

c. Write SQL statement to **create** the ***Ratings* Table** given above. Also specify PK constraint on *(viewerid, fid)* columns, FK constraint on *fid* column and CHECK constraint on *rating* column that ensures that rating is between 1 and 5 inclusively.

ANSWER:

a)

title	worstCount
Title3	2
Title5	3

b)

fid	title	yrrelease	viewerid	fid	rating
f5	Title5	2017	v90	f5	3
f2	Title2	2018	null	null	null
f4	Title4	2017	null	null	null
f6	Title6	2018	null	null	null

c) self.

Q6. *(10 points)* For the above movie database, Specify the **SQL and RA statements** to answer the following questions:
a. Which are the newest films? *Do not hardcode SQL/RA statement. It must be generic.*
b. For each film, give the number of views and the percentage of best ratings received. *Sample output is as follows:*

fid	*TotalViews*	*%bestRatings*
f1	*3*	*33*
f3	*4*	*50*
f5	*5*	*20*

ANSWER:
a)
SQL:

```
SELECT *
FROM films
WHERE yrrelease = (SELECT   MAX(yrrelease)    FROM films);
```

RA:

R1 (newestYear) ← $\mathscr{F}_{MAX(yerelease)}$ (FILMS)

Result ← (FILMS $\bowtie_{yrrelease\ newestYear}$ R1)

b)
SQL:

```
SELECT T1.fid, T1.totalViews, T2.bestRatings/T1.totalViews*10  AS %bestRatings
FROM (SELECT fid, COUNT(*) AS totalViews   FROM ratings   GROUP BY fid) T1
JOIN (SELECT fid, COUNT(*) AS bestRatings   FROM ratings   WHERE rating=5   GROUP BY fid) T2
ON T1.fid = T2.fid
```

RA:

R1 (fid, totalViews) ← fid $\mathscr{F}_{COUNT(*)}$ (RATINGS)

R2 (fid, bestRatings) ← fid $\mathscr{F}_{COUNT(*)}$ ($\sigma_{rating=5}$ RATINGS)

R3 ← (R1 $\bowtie_{R1.fid = R2.fid}$ R2)

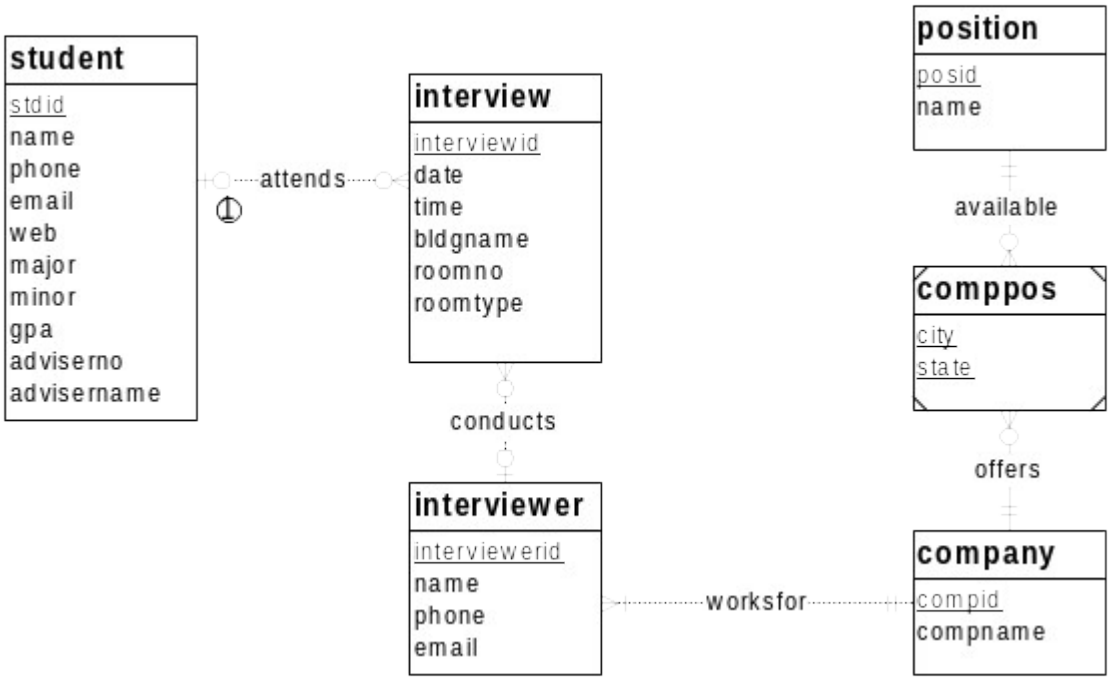
Result (fid, totalViews, %bestRatings) ← $\pi_{fid,\ totalViews,\ bestRatings/totalViews*100}$ (R3)

Q7. *(10 points)* Draw an ER diagram for the following case study. Your diagram should indicate the keys of all entities, as well as the cardinality and participation constraints of all relationships. Note any unspecified requirements, and make appropriate assumptions to make the specification complete but clearly state your assumptions along the diagram.

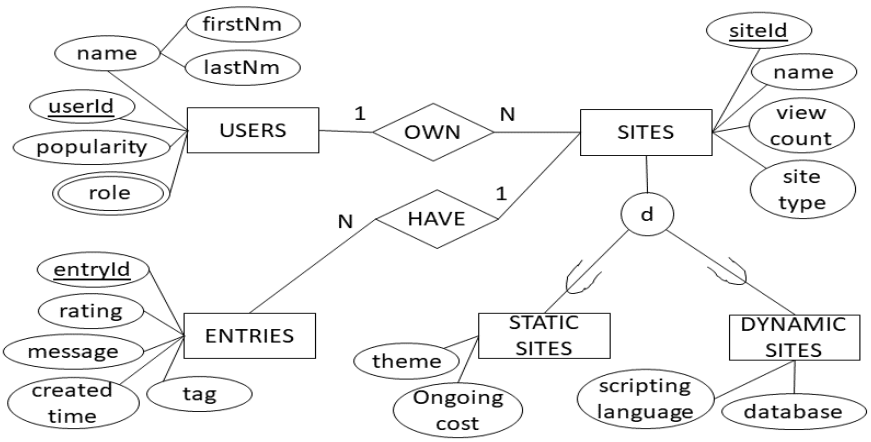
The database supports the placement office of a leading graduate school of computing. The primary purpose of the database is to schedule interviews and facilitate searches by students and companies. Consider the following requirements:

- Student data include a unique student identifier, a name, a phone number, an email address, a major, a minor, and a GPA.
- The placement office maintains a standard list of positions based on the Labor Department's list of occupations. Position data include a unique position identifier and a position description.
- Company data include a unique company identifier, a company name, and a list of positions and interviewers. Each company must map its positions into the position list maintained by the placement office. For each available position, the company lists the cities in which positions are available.
- Interviewer data include a unique interviewer identifier, a name, a phone, an email address, and a web address. Each interviewer works for one company.
- An interview includes a unique interview identifier, a date, a time, a location (building and room), an interviewer, and a student.

ANSWER:



Q8. *(5 points)* Map following EER diagram into a relational model.



ANSWER:

Users (userId, popularity, name) UsersRoles (*userId*, role) Sites (siteId, siteName, *userId*, viewCount, siteType)

StaticSites (*siteId*, theme, ongoingCost) DynamicSites (*siteId*, scriptingLanguage, database)

Entries (entryId, *siteId*, rating, message, createdTime, tag)

Q9. *(10 points)*

a. Discuss the ACID properties of a database transactions. What is the system log used for?

b. How does a category differ from a regular shared subclass? What is a category used for? Illustrate your answer with example.

ANSWER:

a) ACID properties of transactions are:

Atomicity: Transaction performed in its entirety or not at all

Consistency preservation: Takes database from one consistent state to another

Isolation: Not interfered with by other transactions

Durability or permanency: Changes must persist in the database

b) see textbook - EER chapter.