In-Semester Exam-IV (Autumn’2017)
IT 214 Database Management Systems

Time: 75 minutes Max Points: 75

IMPORTANT NOTE:
1. Write answers neat and clean. Answers that are difficult to read may simply be discarded.
2. In all questions marks awarding strategy will be discrete (i.e., 0, half, and full marks).

1. Consider Company Schema. Write a stored function (in pseudo code) that computes standard deviation of salary from EMPLOYEE relation and returns.

   Formula for standard deviation is given here for your reference.

   [20]

$$SD = \sqrt{\frac{\sum(x-\bar{x})^2}{n}}$$

```
CREATE OR REPLACE FUNCTION SALSD() RETURNS real AS
$BODY$
DECLARE
	rec RECORD;
	sum real;
	avg real;
	sd real;
	x real;
	n integer;
BEGIN
	sum := 0;
	n := 0;
	FOR rec IN SELECT * FROM employee WHERE salary IS NOT NULL
	Loop
		sum := sum + rec.salary;
		n := n + 1;
	end loop;
	avg := sum / n;
	sum := 0;
	FOR rec IN SELECT * FROM employee WHERE salary IS NOT NULL
	Loop
		x := avg-rec.salary;
		sum := sum + x * x;
	end loop;
	SELECT sqrt(sum/n) INTO sd;
	return sd;
END;
$BODY$ LANGUAGE plpgsql;
```

2. What are the parameters to getConncection message in JDBC?

   [5]

   Three parameters: Database-URL, user-name, and password

3. Consider Indian Railways scenario from one of a Lab. It has been copied here in the box below for your reference. A short name for every attribute is given in parenthesis.

[20]

Train_Number(**TN**) – every train has unique number. For same pair of stations a train has different numbers for to and return.

Train_Run_Day (**DAY**) – like Monday, Tuesday or so; it is day of run from source station [note that train may not run on all days of a week]

Source_Station_Code (**SRC_SCODE**) – like ADI for Ahmedabad and is unique.

Destination_Station_Code (**DST_SCODE**)

Station_Code (**SCODE**) – any other station on train route

Date_of_Run (**DATE**) – a particular date of run

Scheduled_Arrival_Time (**SAT**) – on a station; assume that train arrives at a on same time on all days.

Scheduled_Departure_Time (**SDT**) – from a station

Expected_Arrival_Time (**EAT**) – on the run date on a station

a. List down minimal FD set on all attributes given here.

TN→ {SRC_SCODE, DST_SCODE}

{TN, SCODE} → {SAT, SDT}

{TN, DATE, SCODE} → EAT

b. Beginning from a single schema R, given below, derives BCNF relations using BCNF decomposition algorithm. Make sure that no FD is lost.

R(TN, DAY, SRC_SCODE, DST_SCODE, SCODE, DATE, SAT, SDT, EAT)

Key: {TN, SCODE, DATE, DAY}

FD1 violates BCNF requirement

R1(<u>TN</u>, SRC_SCODE, DST_SCODE)

R2(TN,DAY, SCODE, DATE, SAT, SDT, EAT)

F1: TN→ {SRC_SCODE, DST_SCODE} ; R1 is in BCNF

F2: (FD2) {TN, SCODE} → {SAT, SDT}, (FD3) {TN, DATE, SCODE} → EAT

Key of R2 is {TN, DAY, SCODE, DATE} and is not in BCNF, FD2 violates BCNF requirement of R2, and decomposed into

R21(<u>TN, SCODE</u>, SAT, SDT)

R22(TN, DAY, SCODE, DATE, EAT)

F21 : (FD2) {TN, SCODE} → {SAT, SDT}; R21 is in BCNF

F22 : (FD3) {TN, DATE, SCODE} → EAT

Key of R22 is {TN, DAY, SCODE, DATE}, and not in BCNF, FD3 violates requirement, therefore decomposed as-

R221(<u>TN, DATE, SCODE</u>,EAT)

R222(<u>TN, DATE, SCODE, DAY</u>)

4. Are following FD sets F and G are equivalent (Yes or NO, Give proofs) - [10]

$F = \{A \rightarrow B, AB \rightarrow C, D \rightarrow E\}$ and $G = \{A \rightarrow BC, D \rightarrow AE\}$

Strategy for solving this

> If every FD in F can be inferred from other FD set G; if answer for all FDs come Yes, then we say that F inferred from G.
>
> Reverse of above is also checked, if G is inferred from F.
>
> Inferencing can be checked either by computing closure or by applying inferencing rules

Solution (1)

Bfore proceeding derive minimal FD set

In second FD in F, B is extraneous; you can prove that as following-
$A \rightarrow B \models A \rightarrow AB$;
$A \rightarrow AB$ and $AB \rightarrow C \models A \rightarrow C$
Having FD $A \rightarrow C$ inferred, we say that B is extra
Finally we have following minimal sets for F and G

F(min)	G(min)
$A \rightarrow B$ (F1)	$A \rightarrow B$ (G1)
$A \rightarrow C$ (F2)	$A \rightarrow C$ (G2)
$D \rightarrow E$ (F3)	$D \rightarrow A$ (G3)
	$D \rightarrow E$ (G4)

Let us check for every FD in F, if they are inferred by G

F1: $A \rightarrow B$; FD is directly there in G. Answer is YES

F2: $A \rightarrow C$; FD is directly there in G. Answer is YES

F3: $D \rightarrow E$; FD is directly present in G. Answer is YES

Now let check for all FDs in G (if they are inferred from F)

G1: $A \rightarrow B$; FD is directly there in F. Answer is YES

G2: $A \rightarrow C$; FD is directly there in F. Answer is YES

G3: $D \rightarrow A$ (G3)
Compute D+(in F) = DE; does not have A, therefore answer is NO

G4: $D \rightarrow E$ (G3); Directly there in F; answer is YES

Since G is not fully inferred from F, we can conclude that both sets are not equivalent.

Solution (2)

F	G
$A \rightarrow B$ (F1) $AB \rightarrow C$ (F2) $D \rightarrow E$ (F3)	$A \rightarrow B$ (G1) $A \rightarrow C$ (G2) $D \rightarrow A$ (G3) $D \rightarrow E$ (G4)

Let us check for every FD in F, if they are inferred by G

F1: $A \rightarrow B$; this FD is directly there in G. Answer is YES
A+ (in G) = ABC, and has B

F2: $AB \rightarrow C$;
Compute AB+ in G = ABC; answer is YES
Also: augment G2 by B (augmentation rule) = $AB \rightarrow CB$; implies $AB \rightarrow C$ (split rule)

F3: $D \rightarrow E$; directly present in G; answer is YES

Now let check for all FDs in G (if they are inferred from F)

G1: $A \rightarrow B$; this FD is directly there in F. Answer is YES

G2: $A \rightarrow C$;
By F1 and F2, we can prove that B is extraneous in F2,
and we get FD $A \rightarrow C$; and hence Answer is YES
Proof: $A \rightarrow B \models A \rightarrow AB$;
$A \rightarrow AB$ and $AB \rightarrow C \models A \rightarrow C$
Therefore B is redundant

OR: Compute A+(in F) = ABC; has C therefore G2 is inferred from F; YES

G3: $D \rightarrow A$ (G3)
Compute D+(in F) = DE; does not have A, therefore answer is NO

G4: $D \rightarrow E$ (G3); Directly there in F; answer is YES

<u>Since G is not fully inferred from F, we can conclude that both sets are not equivalent.</u>

Solution (3); you may even do it without rewriting the FDs; consider original F and G only as following-

F	G
A → B (F1) AB → C (F2) D → E (F3)	A → BC (G1) D → A (G2) D → E (G3)

Let us check for every FD in F, if they are inferred by G

F1: A → B; inferred from G1 by split rule. Answer is YES
A+ (in G) = ABC, and has B

F2: AB → C;
Compute AB+ in G = ABC; answer is YES
Also: augment G1 by B (augmentation rule) = AB → BC; implies AB → C (split rule)

F3: D → E; directly present in G; answer is YES

Now let check for all FDs in G (if they are inferred from F)

G1: A → BC;
Compute A+(in F) = ABC; implies A → BC; Answer YES
Other way:
Augment F1 with A; ⊨ A → AB (T1)
A → AB (T1) and AB → C (F2) ⊨ A → C (T3)
A → B (F1) and A → C (T3) ⊨ A → BC; Answer YES

G2: D → A (G3)
Compute D+(in F) = DE; does not have A, therefore answer is NO

G4: D → E (G3); Directly there in F; answer is YES

<u>Since G is not fully inferred from F, we can conclude that both sets are not equivalent.</u>

5. Give a relation R(ABCDEF), and following FD set F

[10]

$$A \rightarrow B;\ A \rightarrow C;\ CD \rightarrow E;\ CD \rightarrow F;\ B \rightarrow E$$

Do following FDs are inferred from F? Yes/No, Give Proof.

$A \rightarrow E$
$CD \rightarrow EF$
$AD \rightarrow F$
$B \rightarrow CD$

Given FDs
(1) $A \rightarrow B$
(2) $A \rightarrow C$
(3) $CD \rightarrow E$
(4) $CD \rightarrow F$
(5) $B \rightarrow E$

Can be again solved either by computing closure of left side for all FDs in question or by applying inference rules.

$A \rightarrow E$

Apply transitive rule on fds (1) and (5); we can infer A → E; answer is YES.

Also A+=ABCE; confirms YES

$CD \rightarrow EF$

Apply union rules on fds (3) and (4); we can infer CD → EF; answer is YES.

Also CD+=CDEF; confirms YES

$AD \rightarrow F$

Apply augmentation rule on fd (2); we get AD → CD (t1)
Applying transitive rule on (t1) and (4); AD → F; answer is YES.

Also AD+=ABCDEF; confirms YES

$B \rightarrow CD$

Also B+=BE; answer is NO

6. Given a relation R(ABCDEF), and following FDs

[10]

$$ABC \rightarrow E$$
$$ABCD \rightarrow F$$

What is the key? What normal form it is in? Name FD that violates requirement of next immediate higher normal form.

Can you loss-lessly decompose R into BCNF? If yes, give decomposed relations.

Key: ABCD

Normal Form: 1NF

FD1 violates 2NF requirement

Decomposition based on FD1 is

R1(<u>ABC</u>E)
R2(<u>ABCD</u>F)