DIGITAL ASSIGNMENT 1

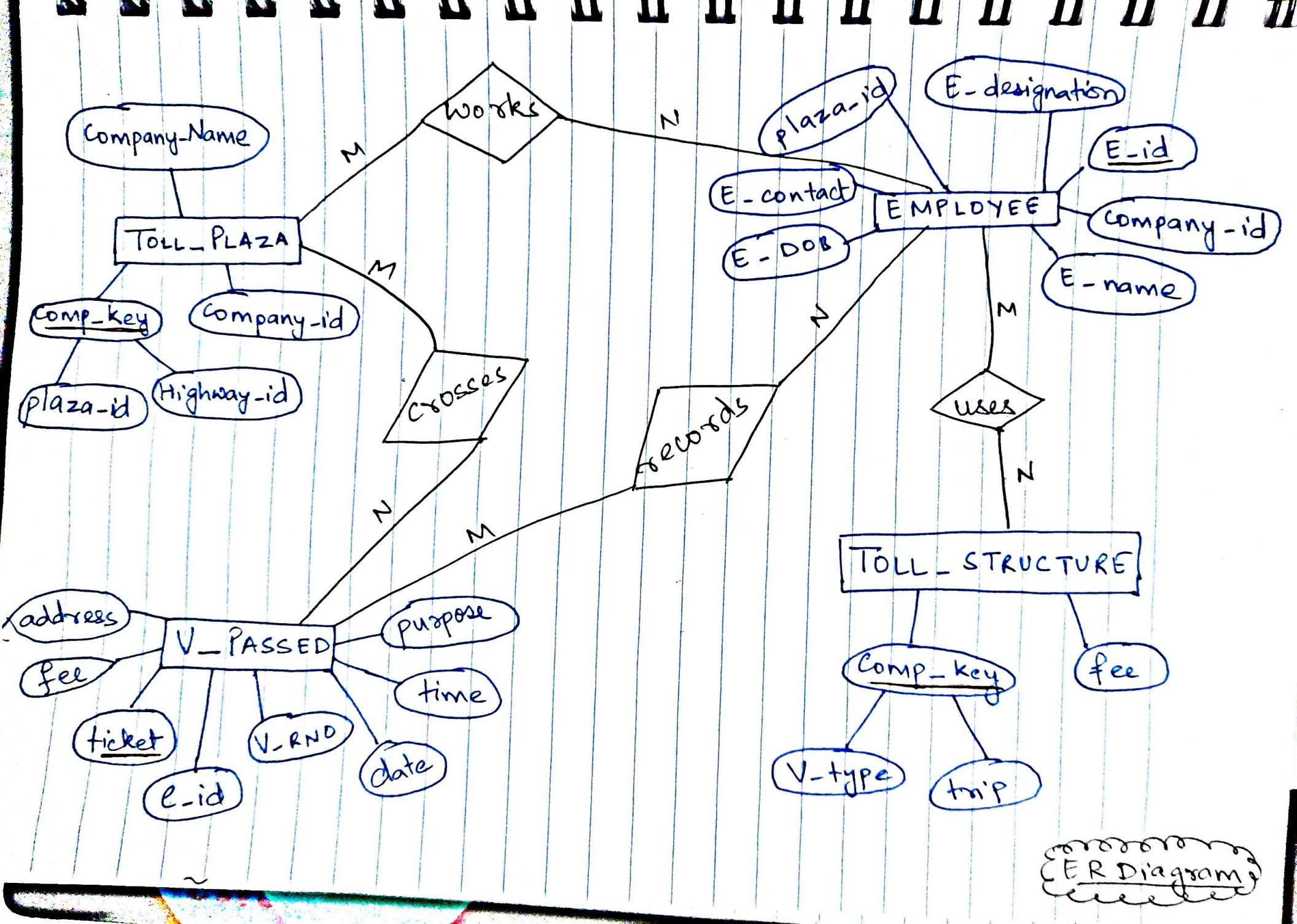
REGISTER NUMBER : 19BCE2074

QUESTION NUMBER : 2

Answer :

Entity sets :

1. **TOLL_PLAZA**
   To store highway number/id, plazas present and company working on plaza
2. **EMPLOYEE**
   Details of employee, the company he/she is working for and plaza duty alloted
3. **V_PASSED**
   Details of vehicle passed during a particular day/time and fee paid.
4. **TOLL_STRUCTURE**
   Basis of fee to be deducted from various types of vehicles.


Company-Name
Works
M
N
plaza-id
E-designation
E-id
E-contact
EMPLOYEE
Company-id
TOLL_PLAZA
E-DOB
E-name
Comp_key
Company-id
M
M
N
plaza-id
Highway-id
Crosses
Records
N
uses
N
TOLL_STRUCTURE
address
V_PASSED
purpose
fee
time
ticket
V_RND
date
e_id
Comp_key
fee
V-type
trip

ER Diagram

Reduction of ER Diagram. [COMPOSITE KEYS PRESENT]

TOLL-PLAZA(plaza_id, Highway_id, company_name, company_id)

EMPLOYEE(E_id, company_id, E_name, E_DOB, E_designation, E_contact, plaza_id)

Works(plaza_id, highway_id, E_id)

V_PASSED(ticket, e_id, V_RNO, date, time, purpose, address, fee)

crosses(ticket, plaza_id, highway_id)

TOLL_STRUCTURE(V_type, trip, fee)

uses(E_id, V_type, trip)

records(ticket, e_id)

Normalisation.

① TOLL_PLAZA (plaza_id, highway_id, company_id, company_name)

Let us assume the schema to be in the form R (A, B, C, D)

A = plaza_id
B = highway_id
C = company_id
D = company_name.

∴ Functional dependencies identified are :-

AB → CD
BC → AD
C → D.

Normalisation :-

1NF

All entries hold only ATOMIC values in all attributes, ie, no multivalued attributes present.

All F.D(s) are in 1NF

2NF

Candidate Keys : AB and BC

as $(AB)^+ = ABCD$

$(BC)^+ = ABCD$

∴ Prime Attributes : A, B, C

Non-prime Attributes : D.

∴ PARTIAL DEPENDENCY present :

$C \longrightarrow D$.

Removing partial dependency.

we get :

$R_1(CD)$

$R_2(BCAD)$

The resultant R_1 and R_2 are in BCNF form as super key is present in all FD(s)

$R_1 \cup R_2 \Rightarrow \{ABCD\} = R$

$R_1 \cap R_2 \Rightarrow \{CD\}$

$\{CD\}^+ = \{CD\} = R_2$.

∴ Lossless decomposition.

From R_1 we get $C \rightarrow D$, R_2 we get $BC \rightarrow AD$

∴ Dependency Preserving

(2). EMPLOYEE(E_id, company-id, E-name, E-DOB, E-designation, E-contact, plaza-id)

Let us assume the schema to be in the form.

R(ABCDEFG)

A = E_id, B = company-id, C = E-name, D = E-DOB, E = E-designation, F = E-contact, G = plaza-id.

∴ The functional dependencies identified are :—

AF ⟶ BCDEG

A ⟶ BCDEFG

F ⟶ ABCDEFG.

Candidate Key. Calculation :

From the dependencies, candidate keys are A and F

as $(A)^+$ = ABCDEFG

$(F)^+$ = ABCDEFG.

Normalisation :-

All dependencies are already in BCNF as all dependencies are containing the SUPER KEY. (AF) ⟶ BCDEG, (A) ⟶ BCDEFG, (F) ⟶ ABCDEG

Hence no further normalisation required

(3) V_PASSED (ticket, e_id, V-RNO, date, time, purpose, address, fee)

Let us assume the schema to be in the form

R (A B C D E F G H)

A = ticket, B = e_id, C = V_RNO, D = date, E = time, F = purpose
G = address, H = fee.

∴ The functional dependencies identified are :-

$A \longrightarrow BCDEFGH$

$CF \longrightarrow G$

$CF \longrightarrow H$.

Normalisation :

1 NF

All entries hold ATOMIC values in all attributes, ie, no multivalued attribute present.

All FD(S) are in 1NF

Candidate Key.

(A) is the candidate key as

$(A)^+ = ABCDEFGH$

2NF

Prime Attributes : A

Non-Prime Attributes : BCDEFGH

$A \longrightarrow BCDEFGH$

(Not Partially dependent as A is SUPER KEY)

$CF \longrightarrow G$

$CF \longrightarrow H$

(Both dependencies of the form

Non-Prime $\longrightarrow$ Non-Prime

$\therefore$ No partial dependency.)

$\therefore$ None of the dependencies are partially dependent

$\therefore$ R is already in 2NF

3NF

$A \longrightarrow BCDEFGH$

(Not transitively dependent as A is SUPER KEY)

∴ TRANSITIVE DEPENDENCIES present are :-

$CF \longrightarrow G$ (Non-prime → Non-prime)

$CF \longrightarrow H$ (Non-prime → Non-prime)

R_1 (CFG)

R_2 (CFH)

R_3 (ABCDEFGH)

The resultant R_1, R_2, R_3 are in BCNF as SUPER KEY is present in all FD(s) of R_1, R_2, R_3.

$R_1 \cup R_2 \cup R_3 = \{ABCDEFGH\} = R$

$R_1 \cap R_2 \cap R_3 \Rightarrow \{CF\}$

$(CF)^+ = CFG$ and CFH

$= R_2$ and R_3.

Hence Lossless decomposition.

From R_1 we get $CF \longrightarrow G$, R_2 we get $CF \longrightarrow H$, R_3 we get $A \longrightarrow BCDEFGH$.

∴ Dependency is preserved.

(4) TOLL-STRUCTURE (V-Type, Trip, fee)

Let the schema be in the form

R(A B C)

A = V-Type, B = Trip, C = fee.

∴ The functional dependencies identified are

$AB \longrightarrow C$

$B \longrightarrow C$

Candidate Key Calculation:

$(AB)^+ = ABC.$

∴ Candidate Key = AB.

Normalisation:

1NF

All entries hold ATOMIC values in all attributes, ie, no multivalued attributes present.

2NF

Candidate Key = AB

Prime Attributes = A B

Non-Prime Attributes = C

↳ $AB \longrightarrow C$

No partial dependency as AB is SUPER KEY.

↳ $B \longrightarrow C$

Partial Dependency as Prime $\longrightarrow$ Non-Prime

Removing partial dependency we get.

R_1 (B C)

R_2 (A B)

The resultant R_1, R_2 are in BCNF as SUPER KEY is present in all FD(C) of R1, R2.

$R1 \cup R2 = \{ABC\}$

$R1 \cap R2 = \{B\}$

$(B)^+ = \{BC\} \Rightarrow R_2$

$\therefore$ Lossless decomposition.

From R1 we get $B \rightarrow C$, R2 we get $AB \rightarrow AB$ (trivial)

$\therefore$ dependency is preserved.