WORKSHEET 4

Student Name: Supriya Dutta **UID: 23BCS13291**

Branch: CSE(3rd Year) **Section/Group: Krg-1-A**

Semester: 5th **Date of Performance: 11/09/25**

Subject Name: ADBMS **Subject Code: 23CSP-333**

1. **AIM:**

i) Consider a relation R having attributes as R(ABCD), functional dependencies are given below:
AB->C, C->D, D->A
Identify the set of candidate keys possible in relation R. List all the set of prime and non prime attributes.

ii) Relation R(ABCDE) having functional dependencies as :
A->D, B->A, BC->D, AC->BE
Identify the set of candidate keys possible in relation R. List all the set of prime and non prime attributes.

iii) Consider a relation R having attributes as R(ABCDE), functional dependencies are given below:
B->A, A->C, BC->D, AC->BE
Identify the set of candidate keys possible in relation R. List all the set of prime and non prime attributes.

iv)Consider a relation R having attributes as R(ABCDEF), functional dependencies are given below:
A->BCD, BC->DE, B->D, D->A
Identify the set of candidate keys possible in relation R. List all the set of prime and non prime attributes.

v) Designing a student database involves certain dependencies which are listed below:
X ->Y, WZ ->X, WZ ->Y, Y ->W, Y ->X, Y ->Z
The task here is to remove all the redundant FDs for efficient working of the student database management system.

2. **Tools Used :** SQL Server Management Studio

Solutions:

Q1) Candidate keys: {AB, BC, BD}

Prime attributes: A, B, C, D

Non-prime attributes: none

Highest normal form:

- 1NF: satisfied (atomic attributes).
- 2NF: applies to partial dependencies of non-prime attributes on part of a candidate key. There are no non-prime attributes, so 2NF holds (vacuously).
- 3NF: For every FD X → Y, either X is a superkey or Y is prime.
  - AB → C : AB is a candidate key → OK.
  - C → D : C is not a superkey, but D is prime → allowed in 3NF.
  - D → A : D is not a superkey, but A is prime → allowed in 3NF.

    So 3NF holds.
- BCNF: Requires every FD's left side be a superkey. C → D and D → A have non-superkey left sides, so BCNF is violated.

  Therefore the highest normal form of R is 3NF (but not BCNF).

Q2) Candidate keys = {AC, BC}.

Prime attributes: A, B, C

Non-prime attributes: D, E

Highest normal form:

- 1NF: holds (attributes atomic).
- 2NF: requires no partial dependency of a *non-prime* attribute on a proper subset of a candidate key.

  Here A → D: A is a proper subset of the candidate key AC, and D is non-prime. That is a partial dependency of a non-prime attribute → violates 2NF.
- Since 2NF is violated, higher normal forms (2NF, 3NF, BCNF) cannot hold.

  Therefore the highest normal form of R is 1NF.

Q3) Candidate keys: A and B

Prime attributes: A, B

Non-prime attributes: C, D, E

Highest normal form:

- 1NF: holds (attributes atomic).
- 2NF: holds. Partial-dependency concerns only arise when a *proper part* of a (composite) candidate key determines a non-prime attribute. Here candidate keys are single attributes (A and B), so there are no proper subsets to cause partial dependencies → 2NF holds.

- 3NF: For every FD X→YX\to YX→Y, either XXX is a superkey or each attribute of YYY is prime. All FDs have left sides that are superkeys:
  - B→A: B is a candidate key → OK.
  - A→C:A is a candidate key → OK.
  - BC→D:BC contains key B so is a superkey → OK.
  - AC→BE:AC contains key A so is a superkey → OK.
    So 3NF holds.
- BCNF: Requires every FD left side to be a superkey. As shown above, every FD's left side *is* a superkey. Therefore BCNF also holds.

The relation is in BCNF (hence also in 3NF and 2NF).

Q4) Candidate keys: {AF, BF, DF}

Prime attributes: A, B, D, F

Non-prime attributes: C, E

Highest normal form:

- 1NF: satisfied (attributes atomic).
- 2NF: violated. Reason: 2NF forbids a *partial dependency* of a non-prime attribute on part of a composite key. Keys here are composite (size 2). For example, key AF — the part A (a proper subset) functionally determines C via A→CA\to CA→C, and C is non-prime. That is a partial dependency of a non-prime attribute on part of a candidate key → violates 2NF.
- Since 2NF fails, higher normal forms (3NF, BCNF) do not hold.

Therefore the highest normal form is: 1NF.

Q5) Candidate keys: {X, Y, WZ}

Prime attributes: W, X, Y, Z (all)

Non-prime: none

Highest normal form:

- 1NF: satisfied (attributes atomic).
- 2NF: there is one composite key WZ. Partial-dependency violations require a non-prime attribute depending on part of a composite key. All attributes are prime, so 2NF holds (vacuously).
- 3NF: for every FD X→A, either X is a superkey or A is prime. Here every given FD has a left side that is a superkey (X and Y are keys; WZ is a key), so 3NF holds.
- BCNF: requires every FD left side to be a superkey. That is also true for the given FDs.

Therefore the relation is in BCNF (and hence also in 3NF and 2NF).

Q6) Candidate key(s): AF

Prime attributes: A, F

Non-prime attributes: B, C, D, E

Highest normal form:

- 1NF: holds (attributes are atomic).
- 2NF: requires that *no non-prime attribute* depend on a proper subset of a *composite* candidate key. Here the candidate key is composite {A,F}. We have A→BC and A→D (and thus A→E), so the non-prime attributes B,C,D,E depend on A, which is a proper part of the key {A,F}. That is a partial dependency → violates 2NF.
- Since 2NF is violated, the relation cannot be in 3NF or BCNF.

Therefore the highest normal form is: *1NF*.

3. Learning Outcomes:

- Learned to compute candidate keys using attribute closure.
- Understood how to classify prime and non-prime attributes.
- Identified partial dependencies and their effect on normalization.
- Determined the highest normal form of a relation step by step.
- Gained practical insight into reducing redundancy and anomalies in database design.