Assignment 4

Student Name: HARSHVARDHAN **UID: 23BCS10363**
Branch: CSE **Section/Group: KRG 1-A**
Semester: 5th **Date of Performance:11/09/2025**
Subject Name: ADBMS **Subject Code: 23CSP-333**

1. **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.

Closure:

- (AB)+: AB → C ⇒ ABC, C → D ⇒ ABCD → covers all.
- (B)+: only B. No further FDs.
- (C)+: C → D ⇒ CD, D → A ⇒ ACD (missing B).
- (BC)+: BC → D ⇒ BCD, D → A ⇒ ABCD → covers all.
- (BD)+: BD → A ⇒ ABD, AB → C ⇒ ABCD → covers all.

Candidate Keys:
Candidate keys are {AB, BC, BD}.

Prime & Non-prime:

- Prime attributes = A, B, C, D (all appear in at least one candidate key).
- Non-prime attributes = none.

---

2. 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.

Candidate Keys:

- (B)+ = B → A ⇒ AB → D ⇒ ABD (C,E missing).
- (C)+ = C only.
- (BC)+ = BC → D ⇒ BCD, B→A ⇒ ABCD (E missing).
- (AC)+ = AC → BE ⇒ ACBE, A→D ⇒ ABCDE → covers all.
  Minimal key = AC (since neither A+ nor C+ covers all).

Candidate Key: AC.

Prime & Non-prime:

- Prime attributes: A, C.
- Non-prime attributes: B, D, E.

---

3. 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.

Candidate Keys:

- (B)+ = B→A ⇒ AB → C ⇒ ABC (missing D, E).
- (BC)+ = BC→D ⇒ BCD, B→A ⇒ ABCD, A→C ⇒ ABCD, AC→BE ⇒ ABCDE → full closure.
  So BC is a candidate key (minimal).

Candidate Key: BC.

Prime & Non-prime:

- Prime: B, C.
- Non-prime: A, D, E.

4. 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.

Candidate Key:

- (A)+ = ABCD, BC→DE ⇒ ABCDE (missing F).
- (B)+ = B→D ⇒ BD, D→A ⇒ ABD, A→BCD ⇒ ABCD (missing E,F).
- (D)+ = D→A ⇒ AD, A→BCD ⇒ ABCD (missing E,F).

- Now test with F:
- (AF)+ = (A)+ + F = ABCDEF → full.
- (BF)+ = B→D ⇒ BD, D→A ⇒ ABD, A→BCD ⇒ ABCD, add F ⇒ ABCDF, BC→DE ⇒ ABCDEF → full.
- (DF)+ = D→A ⇒ AD, A→BCD ⇒ ABCD, add F ⇒ ABCDF, BC→DE ⇒ ABCDEF → full.

Candidate Keys: AF, BF, DF.

Prime & Non-prime:

- Prime: A, B, D, F.
- Non-prime: C, E

5. Designing a student database involves certain dependencies which are listed below:

- **X ->Y**
- **WZ ->X**
- **WZ ->Y**
- **Y ->W**
- **Y ->X**
- **Y ->Z**

I. **Closure &** II. **Candidate Key(s)**

Let's find a minimal superkey.

- (Y)+:
  - Start: Y
  - Using Y->W: WY
  - Using Y->X: WXY
  - Using Y->Z: WXYZ
  - (Y)+ = WXYZ

Is Y minimal? Yes, because no smaller set can work.

- (WZ)+:
  - Start: WZ
  - Using WZ->X: WXZ
  - Using WZ->Y: WXYZ
  - (WZ)+ = WXYZ
    WZ is also a candidate key.
    Candidate Keys: Y and WZ

III. **Prime and Non-Prime Attributes**

- **Prime Attributes:** Parts of the candidate keys (Y, W, Z).
- **Non-Prime Attributes:** There is only one other attribute: X. So, **X** is non-prime.

IV. **Normal Form (NF) and Why?**

- **1NF:** Yes (assumed).
- **2NF: Yes.** There is only one non-prime attribute (X). For the key Y, it's a single attribute key, so no partial dependency is possible. For the key WZ, the non-prime attribute X is dependent on the full key WZ (from WZ->X), so no partial dependency.
- **3NF: Yes.** The only non-prime attribute is X, and it is directly dependent on the candidate keys (Y->X and WZ->X). There is no transitive dependency.
- **BCNF: No.** Check all FDs. The FD X -> Y is a problem. X is a non-prime attribute, and it determines Y, which is a prime attribute. But for BCNF, the left side of *every* FD must be a superkey. X is not a superkey. This violates BCNF.
- **Conclusion:** The highest normal form is **3NF**.

6. Debix Pvt Ltd needs to maintain database having dependent attributes ABCDEF. These attributes are functionally dependent on each other for which functionally dependency set F given as:

{A -> BC, D -> E, BC -> D, A -> D} Consider a universal relation R1(A, B, C, D, E, F) with functional dependency set F, also all attributes are simple and take atomic values only. Find the highest normal form along with the candidate keys with prime and non-prime attribute.

I. **Closure &** II. **Candidate Key(s)**

Let's find a minimal superkey. Notice F is not on the right side of any FD, so it must be part of every candidate key.

- (A)+:
  - Start: A
  - Using A->BC: ABC
  - Using A->D: ABCD
  - Using BC->D: ABCD (same)
  - Using D->E: ABCDE
  - (A)+ = ABCDE (Missing F)

Therefore, A alone is not a key. Let's try (AF)+:

- (AF)+ = (A)+ + F = ABCDEF
  Is AF minimal? Check if A is necessary: (A)+ = ABCDE (missing F). Check if F is necessary: (F)+ = F. So both are needed.
  Is there a smaller key? Could A be replaced? No. So AF is a candidate key.
  We could also have other keys like (ABF)+, (ACF)+, etc., but they are not minimal since AF is sufficient.
  Candidate Key: AF

III. **Prime and Non-Prime Attributes**

- **Prime Attributes:** Parts of the candidate key (A, F).
- **Non-Prime Attributes:** The remaining attributes (B, C, D, E).

IV. **Normal Form (NF) and Why?**

- **1NF:** Yes (as stated in the problem).
- **2NF: No.** There are partial dependencies. The candidate key is AF. Look at the FD A->BC. A is a proper subset of the key, and it determines the non-prime attributes B and C. This is a partial dependency, which violates 2NF. Similarly, A->D is a partial dependency.
- **Conclusion:** The highest normal form is **1NF**.