Assignment 2

1. Consider a relation R with attributes ABCDEF. You are given the following dependencies A-> B, D -> E, AC -> E.

   1. Attribute closure of A?
      $\{A\}^+ = \{AB\}$
   2. Attribute closure of C?
      $\{C\}^+ = \{C\}$
   3. Attribute closure of D?
      $\{D\}^+ = \{DE\}$
   4. Attribute closure of AC?
      $\{AC\}^+ = \{ABCE\}$
   5. Attribute closure of ABCDEF?
      $\{ABCDEF\}^+ = \{ABCDEF\}$
   6. Identify which, if any, of the above are candidate keys.
      None of the above are candidate keys.

2. Consider a relation R with attributes ABCDE. You are given the following dependencies AC-> B, AC-> D, AC-> E, E -> C.

   1. List all (candidate) keys for R.
      {AC, AE}
   2. Is R in 3NF? Why or why not?
      R is in 3NF because every FDs RHS attribute is a prime attribute.
   3. Is R in BCNF? Why or why not?
      R is NOT in BCNF because of the FD: E → C, in which LHS attribute E is not a super key nor a candidate key.

3. Consider a relation R with attributes ABCDE. You are given the following dependencies AB -> C, AB -> D. AB -> E.

   1. List all (candidate) keys for R.
      {AB}
   2. Is R in 3NF? Why or why not?
      R is in 3NF because every FDs RHS attribute is a prime attribute.
   3. Is R in BCNF? Why or why not?
      R is in BCNF because every LHS attribute in every FD is a super key or candidate key.

For the next set of questions consider the relational schema R=(P, Q, R, S, T, U, V, W) and the set of functional dependencies FD:

$$
\begin{aligned}
Q &\rightarrow U && (1)\\
U &\rightarrow V && (2)\\
PQ &\rightarrow WST && (3)\\
SU &\rightarrow TR && (4)\\
VT &\rightarrow RW && (5)\\
R &\rightarrow W && (6)
\end{aligned}
$$

(a) Which of the following is a minimum cover of the FD? Mark all that qualify;if none, mark accordingly, and give your *own.* answer. SHOW THE WORK.

i. The given FDs (Eq 1-6), is a minimum cover already.

ii. $\{Q \rightarrow U, U \rightarrow V, PQ \rightarrow S, SU \rightarrow T, SU \rightarrow R, VT \rightarrow R, VT \rightarrow W, R \rightarrow W\}$

iii. $\{Q \rightarrow U, U \rightarrow V, PQ \rightarrow S, SU \rightarrow T, PQ \rightarrow W, VT \rightarrow R, PQ \rightarrow T, R \rightarrow W\}$

iv. $\{Q \rightarrow U, U \rightarrow V, PQ \rightarrow S, SU \rightarrow T, VT \rightarrow R, R \rightarrow W\}$

v. $\{Q \to U, U \to V, PQ \to S, SU \to T, SU \to R, VT \to R, PQ \to T, R \to W\}$

vi. *None of the above.*

iv. {Q → U, U → V, PQ → S, SU → T, V T → R, R → W } is the minimal cover.

Showing work:

First, convert the set of functional dependencies to canonical form:

Q→U

U→ V

PQ → W

PQ → S

PQ → T

SU → T

SU → R

VT → R

VT → W

R → W

Next, check for any extraneous attributes on the LHS.

There are no extraneous attributes on the LHS of any of these FDs. This is because the possible FDs such as P→W or Q → W can't be obtained from the other FDs through the inference rules.

Next, check for any redundant FDs.

The redundant FDs are:

PQ → T

PQ → W

SU → R

VT → W

Therefore, the minimal cover is: {Q→U,U→V, PQ→S, SU→T, VT→R, R→W}, which corresponds to answer iv.

Q. 14.26, 14.27 14.30

14.26. Consider the following relation:

A	B	C	TUPLE#
10	b1	c1	1
10	b2	c2	2
11	b4	c1	3
12	b3	c4	4
13	b1	c1	5
14	b3	c4	6

a. Given the previous extension (state), which of the following dependencies *may hold* in the above relation? If the dependency cannot hold, explain why *by specifying the tuples that cause the violation*.

i. $A \to B$, ii. $B \to C$, iii. $C \to B$, iv. $B \to A$, v. $C \to A$

b. Does the above relation have a potential candidate key? If it does, what is it? If it does not, why not?

a. i. $A \to B$ may not hold due to tuples 1 and 2
   ii. $B \to C$ may hold
   iii. $C \to B$ may not hold due to tuple 3
   iv. $B \to A$ may not hold due to tuples 1 and 5
   v. $C \to A$ may not hold due to tuples 1 and 3

b. Yes, the above relation has a potential candidate key. AB is a potential candidate key for the above relation.

14.27. Consider a relation $R(A, B, C, D, E)$ with the following dependencies:

$$AB \to C,\ CD \to E,\ DE \to B$$

Is AB a candidate key of this relation? If not, is ABD? Explain your answer.

No, AB is not a candidate key of this relation because the candidate key needs to contain AD since AD is not present on any of the FD's RHS.

ABD is a candidate key because: $\{ABD\}^+ = \{ABCDE\}$

1. ABD determines itself
2. AB determines C
3. CD determines E

4. Therefore $\{ABD\}^+ = \{ABCDE\}$
5. Thus, ABD is a candidate key

14.30. Consider the following relation:

CAR_SALE(Car#, Date_sold, Salesperson#, Commission%, Discount_amt)

Assume that a car may be sold by multiple salespeople, and hence {Car#, Salesperson#} is the primary key. Additional dependencies are

Date_sold → Discount_amt and
Salesperson# → Commission%

Based on the given primary key, is this relation in 1NF, 2NF, or 3NF? Why or why not? How would you successively normalize it completely?

This relation is in 1NF because:

- Each column is unique.
- Each attribute is unique.
- No row has duplicate/composite values.

The relation is not in 2NF because the nonprime attribute ***Commission%*** is partially dependent on the primary key***. Commission%*** is partially dependent on ***Salesperson#*** as: ***Salesperson# → Commission%.***
It is not in 3NF because it is not in 2NF.

To normalize completely normalize the relation:

1. Make the relation satisfy the requirements of 2NF. Hence, we need to get rid of the partial dependency. To do this we must make a new relation, remove the dependent nonprime attribute of the first relation and put it in the new relation.

   We should now have the following relations:

   CAR_SALE(Car#, Date_sold, Salesperson#, Discount_amt)
   NEW_RELATION(Salesperson#, Commission%)

2. We must get rid of the transitive dependency in the relation CAR_SALE. The transitive dependency is ***: {Car#, Salesperson#} → Date_sold → Discount_amt***. To get rid of this transitivity, we must do the same thing as the previous step.

   We should now have the following relations:

   CAR_SALE(Car#, Date_sold, Salesperson#)
   NEW_RELATION1(Date_sold, Discount_amt)
   NEW_RELATION2(Salesperson#, Commission%)

3. The relation is now normalized completely