

Q1-1: As an example of insertion anomaly, we can mention that if we want to insert a new Airline, we will be forced to insert number of flights for that airline (unintentionally), which is possible that airline hasn't have any departures till now. Or as another example, we want to add a new terminal, but then we also need to add a new AirlineID and AirlineName, which is unintentionally, and we might don't want to have new Airline in the list of our airlines in the table.

Q1-2: As an example of deletion anomaly, we can mention that, if we delete airline Southwest, TerminalID 'C' also will be deleted unintentionally and this is while this airline is the only airline that is assigned to terminalID 'C', so this will change our data about terminalIDs in the table, we will only have terminal A and B.

Q1\_3: As an example of modification anomaly we can mention that, if we want to modify TerminalID 'A' to for example 'M', may cause this update to be made on all AirlineNames that are assigned to TerminalID 'A', so one modification causes some other unintentional modifications on other things like AirlineNames.

Q1\_4-a: full functional dependencies:

$\{Date, AirlineID\} \rightarrow NumberOfDepartingFlights$

$\{Date, AirlineID\} \rightarrow AirlineName$

$\{Date, AirlineID\} \rightarrow TerminalID$

Q1\_4-b: partial functional dependencies

$AirlineID \rightarrow AirlineName$

$AirlineID \rightarrow TerminalID$

$AirlineID \rightarrow NumberOfGates$

Q1\_4-c: Transitive functional dependencies

$AirlineID \rightarrow TerminalID$

$TerminalID \rightarrow NumberOfGates$

## Normalization of Airport KLX relation

(Date, AirlineID, AirlineName, TerminalID, NumberOfGates, NumberOfDepartingFlights)

**Step1:** Check First Normal Form : the relation (Date, AirlineID, AirlineName, TerminalID, NumberOfGates, NumberOfDepartingFlights) is in the 1NF, because all attributes are single valued attributes (atomic), there is no nested relations and there is no composite attribute

**Step2:** Second Normal Form: (Date, AirlineID, AirlineName, TerminalID, NumberOfGates, NumberOfDepartingFlights) is not in the 2<sup>nd</sup> normal form. To be in the 2NF, the relation should be in the 1NF and it should not have partial dependencies between non-prime attributes with prime keys, in other words every non-prime attribute A in R is fully functionally dependent on the primary key. Partial dependency:

AirlineID -> AirlineName, TerminalID, NumberOfGates

Show partial dependency, because non-prime attributes are dependent on a part of candidate key, which is AirlineID. To make it 2NF, we need to do decomposition:

(AirlineID, AirlineName)

(AirlineID, TerminalID, NumberOfGates)

(Date, AirlineID, NumberOfDepartingFlights)

**Step 3:** We need to check if relations are in the third normal form, 3NF. We need to be sure that the relations are in the 1NF and 2NF and there should not be any transitive dependencies. The relation isn't in 3NF, because it has transitive dependency.

We need to decompose (AirlineID, TerminalID, NumberOfGates) into:

(AirlineID, TerminalID)

(TerminalID, NumberOfGates)

**Step4:** check for BCNF:

List all functional dependencies:

AirlineID -> AirlineName

AirlineID -> TerminalID

TerminalID -> NumberOfGates

{Date, AirlineID} -> NumberOfDepartingFlights

All the left-hand side attributes are a key, there is no other FD than these 4 FD, so all 4 relations are in BCNF

Q2-1: All candidate keys:

EmplID

ProjID

(EmplID, ProjID, Specialization)

(EmplID, Specialization)

Q2-2

Relations are in 1NF, 2NF and 3NF, but not in BCNF

Relations:

ProjID  $\rightarrow$  ProjTitle, SupervisorName

SupervisorName  $\rightarrow$  SupervisorLocation

ManagerID  $\rightarrow$  Specialization

Violates BCNF, because ProjID, SupervisorName and ManagerID are not candidate keys

We decompose to:

(ProjID, ProjTitle, SupervisorName )

(SupervisorName, SupervisorLocation)

(ManagerID, Specialization)

(EmpID, EmpName, Bonus)

(EmpID, ProjID, Specialization, ManagerID)