

1.

normalization process:

(1NF=has primary key+no repeating groups)

It is 1NF because the primary key is customerNumber, date, time, and partNumber and there is no repeating group.

all data in one table:

customerName

customerNumber PK

customerType

date PK

time PK

employee

partNumber PK

name

type

cageCode

quantityOrdered

unitPrice

(2NF=1NF+one field that makes up the primary key or all non-PK fields are dependent on all the PK fields)

It is not 2NF because it has a composite PK and there are attributes dependent upon only part of the PK:

name, type, cageCode, and unitPrice are dependent on partNumber

customerName and customerType are dependent on customerNumber

employee are dependent on the date

Moving these fields out of the table results in the following 2NF version:

order_detail

partNumber PK FK

customerNumber PK FK

date PK FK

time PK

quantityOrdered

customer_detail

customerNumber PK

customerName

customerType

part_detail

partNumber PK

name
type
cageCode
unitPrice

employee_detail
date PK
employee

(3NF=2NF+no transitive dependencies)

It is not in 3NF as there is a transitive dependency:

cageCode is dependent on type

Moving this data out of the tables results in the following 3NF versions:

Tables:

order_detail
partNumber PK FK
customerNumber PK FK
date PK FK
time PK
quantityOrdered

customer_detail
customerNumber PK
customerName
customerType

part_detail
partNumber PK
name
type FK
unitPrice

employee_detail
date PK
employee

inventory_detail
type PK
cageCode

Assumption:

- assume each type corresponds to a cage, and all inventory of the same type is placed in a cage. Therefore, inventory of the same type has the same cage code.
- Only one customer is being helped at a time
- each day has only one employee

normalization process:

(1NF=has primary key+no repeating groups)

It is a 1NF since it has the primary key and there is no repeating groups.

The primary key is staffNo, patNo, and appointmentDateTime.

all data in one table:

staffNo PK

therapistName

patNo PK

patName

appointmentDateTime PK

branchNo

(2NF=1NF+one field that makes up the primary key or all non-PK fields are dependent on all the PK fields)

It is not 2NF because it has a composite PK and there are attributes dependent upon only part of the PK:

therapistName are dependent on staffNo

patName and branchNo are dependent on patNo

Moving these fields out of the table results in the following 2NF version:

therapist_order

staffNo PK FK

patNo PK FK

appointmentDateTime PK

branchNo FK

patient_detail

patNo PK

patName

branchNo

therapist_detail

staffNo PK

therapistName

(3NF=2NF+no transitive dependencies)

It is 3NF because there is no transitive dependency.

therapist_order

staffNo PK FK

patNo PK FK

appointmentDateTime PK

branchNo FK

patient_detail

patNo PK

patName

branchNo

therapist_detail

staffNo PK

therapistName

Assumption:

-Patients always only go to one branch, so one patients only appears in one branches

normalization process:

(1NF=has primary key+no repeating groups)

It is a 1NF since it has the primary key and there is no repeating groups.

The primary key is eNo and contractNo.

all data in the table:

eNo PK

contractNo PK

hours

eName

eventNo

eventLoc

(2NF=1NF+one field that makes up the primary key or all non-PK fields are dependent on all the PK fields)

It is not 2NF since it has a composite PK and there are attributes dependent upon only part of the PK:

eName is dependent on eNo

eventNo and eventLoc is dependent on contractNo

Moving these fields out of the table results in the following 2NF version:

employee_contract

contractNo PK FK

eNo PK FK

hours

contract_detail

contractNo PK

eventNo

eventLoc

employee_detail

eNo PK

eName

(3NF=2NF+no transitive dependencies)

Using the tables created in 2NF, it is not 3NF because eventLoc is dependent on eventNo. It has transitive dependency.

In order to resolve this I would move eventLoc into a new table.

employee_contract

contractNo PK FK

eNo PK FK

hours

contract_detail
contractNo PK
eventNo FK

employee_detail
eNo PK
eName

event_detail
eventNo PK
eventLoc

Assumption:
-the event can happen in only one location