Brandon Phipps Section 1

CSCI 466 Assignment 3 (Spring 2022)

Normalization (50 pts)

The Task

We discussed in class that a relational database designed in a poor way will allow for *anomalies* to occur. This is undesirable, so we use normalization to prevent them. Several relations are provided below, along with their functional dependencies. Answer the questions provided and fix what is broken. Perform only the current step for each question, i.e. when fixing First Normal Form (1NF), fix only 1NF, leaving the 2NF and 3NF violations untouched until the question that asks about them.

The Questions

For each of the below, part b refers to the results of part a , and part c refers to the results of part b – any changes made during the previous steps should be considered in the steps that follow. Each of these is worth 10 points for a total of 50.

1. 1  **R**(A, B, C, D, E, F, G) **Functional Dependencies:** 
   * ▶  A⟶D,E,F
   * ▶  C⟶G
   * ▶  A,C⟶H

a  Is this relation in 1NF? If not, explain why not, then make the necessary changes to fix it.

**This relation is not 1NF, as it is atomic but doesn’t have any primary keys. You would need to make A,C the primary key. R**(A,B,C,D,E,F,G)

b  Is this relation in 2NF? If not, explain why not, then make the necessary changes to fix it.

* + **No, the relation is not in 2NF. Functional dependencies use only parts of the primary key.** A⟶D,E,F  C⟶G

**RA**(A,D,E,F)

**RC**(C,G)

c  Is this relation in 3NF? If not, explain why not, then make the necessary changes to fix it.

Brandon Phipps Section 1

**Yes, this relation is in 3NF. No non-prime attributes, that so called determine non dependencies.**

1. **StockExchange**(Company, Symbol, HQ, Date, ClosePrice) **Functional Dependencies:** 
   * ▶  Symbol, Date ⟶ Company, HQ, ClosePrice
   * ▶  Symbol ⟶ Company, HQ
   * ▶  Symbol ⟶ HQ

a  Is this relation in 1NF? If not, explain why not, then make the necessary changes to fix it.

**No, this relation isn’t 1NF, as no primary keys are listed, although it is atomic. The primary key should be** Symbol, Date.

**StockExchange**(Company, Symbol, HQ, Date, ClosePrice)

b  Is this relation in 2NF? If not, explain why not, then make the necessary changes to fix it.

**No, it is not in 2NF, as** Symbol ⟶ (Company, HQ), (Symbol ⟶ HQ) **are parts of the primary key, and we need to decompose StockExchange into two relations.**

**StockExchange1**(Symbol, Date, ClosePrice)

Symbol, Date ⟶ ClosePrice

**StockExchange2**(Symbol, Company, HQ)

Symbol ⟶ Company, HQ

c  Is this relation in 3NF? If not, explain why not, then make the necessary changes to fix it.

**Yes, because none of the rhs nonprimes determine another non-prime / non dependence.**

Brandon Phipps Section 1

1. **Company**(EmpID, EmpName, EmpAddr, (ProjID, ProjName, MgrID, MgrName, HoursWorked)) **Functional Dependencies:** 
   * ▶  EmpID ⟶ EmpName, EmpAddr
   * ▶  ProjID ⟶ ProjName, MgrID, MgrName
   * ▶  EmpID, ProjID ⟶ HoursWorked
   * ▶  MgrID ⟶ MgrName

a  Is this relation in 1NF? If not, explain why not, then make the necessary changes to fix it.

**This is not 1NF, as the duplicate repeating group** (ProjID, ProjName, MgrID, MgrName, HoursWorked) and would be fixed by making a attribute of that group a primary key. (EmpID, ProjID).

**Company** (EmpID, EmpName, EmpAddr, ProjID, ProjName, MgrID, MgrName, HoursWorked)

b  Is this relation in 2NF? If not, explain why not, then make the necessary changes to fix it.

No, this is not 2NF, as the primary key: (EmpID, ProjID). It is used as partial keys in the first and second functional dependencies. (EmpID, ProjID) must be foreign keys in the new relations.

**Company** (EmpID, ProjID, HoursWorked)

**Company** (EmpID, EmpName, EmpAddr)

**Company** (ProjID, ProjName, MgrID, MgrName)

c  Is this relation in 3NF? If not, explain why not, then make the necessary changes to fix it.

**No, it is not in 3NF, as the FD is made up of non-prime attributes. To fix this you would take out MgrID and make it a foreign key to the ProjID FD.**

**Company** (EmpID, ProjID, HoursWorked)

**Company** (EmpID, EmpName, EmpAddr)

**Company** (ProjID, ProjName, MgrID)

**Company (**MgrID, MgrName)

Brandon Phipps Section 1

1. Property(id, county, lotNum, lotArea, price, taxRate, (datePaid, amount)) **Functional Dependencies:** 
   * ▶  id ⟶ county, lotNum, lotArea, price, taxRate
   * ▶  lotArea ⟶ price
   * ▶  county ⟶ taxRate
   * ▶  id, datePaid ⟶ amount

a Is this relation in 1NF? If not, explain why not, then make the necessary changes to fix it.

**No this is not 1NF as it isn’t atomic (has repeating data** (datePaid, amount)**) and doesn’t have any primary keys listed. You would make datePaid and id primary composite keys. (Id, datePaid).**

Property (id, county, lotNum, lotArea, price, taxRate, datePaid, amount)

b Is this relation in 2NF? If not, explain why not, then make the necessary changes to fix it.

**This relation is not 2NF, as there are partial relations in the 1st FD. (**id ⟶ county, lotNum, lotArea, price, taxRate). You would need to decompose it and make id a foreign key to the primary keys of id, and datePaid.

Property (id, datePaid, amount)

Property (id, county, lotNum, lotArea, price, taxRate)

c  Is this relation in 3NF? If not, explain why not, then make the necessary changes to fix it.

**This is not 3NF as the FDs (**lotArea ⟶ price), (county ⟶ taxRate) are non-prime keys, and need to be included in foreign keys from the relation. Property (id, county, lotNum, lotArea, price, taxRate)

Property (id, datePaid, amount)

Property (id, county, lotNum, lotArea)

Property (lotArea, price)

Property (county, taxRate)

Brandon Phipps Section 1

5 **Pharmacy** (patient\_id, patient\_name, address, (Rx\_num, trademark\_name, generic\_name, (filldate, num\_refills\_left), num\_refills))  
**Functional Dependencies:**

* ▶ patient\_id ⟶ patient\_name, address
* ▶ patient\_id, Rx\_num ⟶ trademark\_name, generic\_name
* ▶ Rx\_num ⟶ num\_refills
* ▶ Rx\_num, filldate ⟶ num\_refills\_left

a Is this relation in 1NF? If not, explain why not, then make the necessary changes to fix it.

**This relation isn’t 1NF as it has duplicate data (non atomic) and has no primary keys:**

(Rx\_num, trademark\_name, generic\_name, (filldate, num\_refills\_left), num\_refills)).

**To fix this we need to elect primary keys which are patient\_id, Rx\_num filldate**

**Pharmacy** (patient\_id, patient\_name, address, Rx\_num, trademark\_name, generic\_name, filldate, num\_refills\_left, num\_refills)

b Is this relation in 2NF? If not, explain why not, then make the necessary changes to fix it.

**This relation is not 2NF, each of the FDs from the original example have partial keys, and need to be decomposed to fix them, assigning foreign keys where needed.**

**Pharmacy** (patient\_id, patient\_name, address)

**Pharmacy** (Rx\_num, filldate, num\_refills\_left)

**Pharmacy** (Rx\_num, num\_refills)

**Pharmacy** (patient\_id, Rx\_num, trademark\_name, generic\_name)

c Is this relation in 3NF? If not, explain why not, then make the necessary changes to fix it.

**This relation is 3NF as It has no non-prime keys, and carries over no dependency on one another.**