CS 122A : Introduction to Data Management-Assignment 3

Fall 2019

The questions on Functional Dependencies are **optional**, You are only required to submit answers to relational Algebra Queries.
Deadline : Saturday, November 2nd 23:45 PM

Functional Dependencies

1) Consider the relation R(A,B,C,D,E,F) with FD's:

CDE → B
ACD → F
BEF → C
B → D

a) Show that ACED is a superkey using Armstrong's Axioms.
b) Is ACED a candidate key - why/why not? Explain.

2) Consider the relational schema R(A,B,C,D,E,F,G,H) and the following functional dependencies over R:

A → BCF
AF → E
DEG → H
D → GH

Using Armstrong's Axioms to derive functional dependencies, identify the candidate key for the above relation.

Relational Algebra Queries

For answering the following questions, use the tables provided at the end of this assignment. Express in relational algebra the following queries. If it can not be expressed using relational algebra, then explain why.

Assumptions: Two people cannot be at the same location (X,Y) . Each time a person throws an object assume they are in the same location (X,Y) as the waste bin and whenever a person throws an object LocationObservation,LoadObservation, and ObjectRecognitionObservation are generated with the same timestamp. Notice that locationObservation may be generated at any other timestamps too.

1. Find names and school names of the faculty members whose research area is 'Privacy and Security'

2. Find the names of all buildings with no compost bins

3. Find the location of bins whose load sensor records a weight to be higher than the capacity of the bin after the last collection time which was '2019-10-26 13:00:00'

4. Find ids of users that used a waste bin between '2019-10-26 13:00:00' and '2019-10-26 15:00:00'

5. Find the Outside bins (bins not inside the buildings) that are used by Visitors

6. Find the names of all the students who never used a recycling bin incorrectly (never put any wrong item in the bin). Remember that ObjectRecognitionSensor records a trash_type which can be used to determine if someone incorrectly throws a wrong type of trash in a waste bin.

7. Find the users which have used all the bins

8. Find the users who have never used any recycling bin.

Submission

1. Please submit an appended pdf file with your answers to Relational Algebra Queries. The name of the pdf file should be last names of each team member placed together. For example if

Edgar Codd, Donald Chamberlin and Peter Chen were teammates, they would submit: codd_chamberlin_chen_assignment1.pdf.

2. Upload pdf to Gradescope. Only one member of your team is required to submit the file. Be sure to identify all the team members in pdf file (name and student ID).

Relational Model

General rules:

1. Underlined attributes are primary keys. Primary keys are not NULL.

School(name, budget, dean)
All non key attributes are not NULL.
School(dean) ⊆ Faculty(user_id)

Building(name, lowerLeftX, lowerLeftY,upperRightX,upperRightY)
All non key attributes are not NULL.

Department(school_name, dept_name, building_name, start_date, phone_no, chair, manager)
All non-key attributes are not null.
Department(school_name) ⊆ School(name)
Department(building_name) ⊆ Building(name)
Department(chair) ⊆ Faculty(user_id)
Department(manager) ⊆ Staff(user_id)

User(user_id, name)
All non key attributes are not NULL.

Visitor(user_id, purpose)
All non key attributes are not NULL.
Visitor(user_id) ⊆ User(user_id)

Faculty(user_id, school_name, dept_name, uci_email_address, research_area)
All non key attributes are not NULL.
Faculty(user_id) ⊆ User(user_id)
Faculty(school_name, dept_name) ⊆ Department(school_name, dept_name)

Staff(user_id,school_name, dept_name, uci_email_address, employement_type)
All non key attributes are not NULL.
Staff(user_id) ⊆ User(user_id)
Staff(school_name, dept_name) ⊆ Department(school_name, dept_name)

Student(user_id,school_name, dept_name, uci_email_address, type)
All non key attributes are not NULL.
Type can take values from Enum (“Undergraduate”, “Graduate”).
Student(user_id) ⊆ User(user_id)
Student(school_name, dept_name) ⊆ Department(school_name, dept_name)

WasteBin(waste_bin_id, capacity, X, Y)
All non-key attributes are not null.

CompostBin(waste_bin_id, composting_type)
All non-key attributes are not null.
CompostBin(waste_bin_id) ⊆ WasteBin(waste_bin_id)

RecycleBin(waste_bin_id, contaminated)
All non-key attributes are not null.
RecycleBin(waste_bin_id) ⊆ WasteBin(waste_bin_id)

LandfillBin(waste_bin_id, compression_support)
All non-key attributes are not null.
LandfillBin(waste_bin_id) ⊆ WasteBin(waste_bin_id)

Sensor(sensor_id, ip_address)
All non-key attributes are not null.

LocationSensor(sensor_id, model, max_range, user_id)
All non-key attributes are not null.
LocationSensor(sensor_id) ⊆ Sensor(sensor_id)
LocationSensor(user_id) ⊆ User(user_id)

LoadSensor(sensor_id, waste_bin_id, battery_power, measurement_type)
All non-key attributes are not null.
LoadSensor(sensor_id) ⊆ Sensor(sensor_id)
LoadSensor(waste_bin_id) ⊆ WasteBin(waste_bin_id)

ObjectRecognitionSensor(sensor_id, waste_bin_id, battery_power, quality)
All non-key attributes are not null.
ObjectRecognitionSensor(sensor_id) ⊆ Sensor(sensor_id)
ObjectRecognitionSensor(waste_bin_id) ⊆ WasteBin(waste_bin_id)

LocationObservation(sensor_id, oid, X, Y, timestamp)
All non-key attributes are not null.

LocationObservation(sensor_id) ⊆ Sensor(sensor_id)

LoadObservation(sensor_id, oid, weight, timestamp)
All non key attributes are not NULL.
LoadObservation(sensor_id) ⊆ Sensor(sensor_id)

ObjectRecognitionObservation(sensor_id, oid, trash_type, timestamp)
All non key attributes are not NULL.
ObjectRecognitionObservation(sensor_id) ⊆ Sensor(sensor_id)