1. Functional dependencies and normal forms [65 pts]

A. Poster_Incentive(pid, aid, incentive, total) [22 pts]

(a) [10pts] List all of the functional dependencies involving the attributes of the relation in question. (Just give the basic list, not the closure of that list. ☺)

FD1: (pid, aid) → pid, aid, incentive, total  (optional)
FD2: aid → incentive, total

(b) [2 pts] What are the candidate keys for this relation?

(pid, aid)

(c) [10pts] Identify which normal form their relation is in by answering the following questions:

c-1> [2pts] Does their relation satisfy 1NF [Yes/No]? Show your reasoning.
Yes. There are no multi-valued attributes.

c-2> [4pts] Does their relation satisfy 2NF [Yes/No]? Show your reasoning.
No. FD2 violates the 2NF condition since the incentive and total attributes are dependent on a part of the key - aid.

c-3> [4pts] Does their relation satisfy 3NF [Yes/No]? Show your reasoning.
No. The relation is not in 2NF.
B. **User_Activity_Log(uid, op_type, op_time)** [22 pts]

(a) [10 pts] List all of the functional dependencies involving the attributes of the relation in question. (Just give the basic list, not the closure of that list. ☺)

FD1: (uid, op_type) → op_time

(b) [2 pts] What are the candidate keys for this relation?

(uid, op_type)

(c) [10 pts] Identify which normal form their relation is in by answering the following questions:

- c-1> [2 pts] Does their relation satisfy 1NF [Yes/No]? Show your reasoning.

  Yes. There are no multi-valued attributes.

- c-2> [4 pts] Does their relation satisfy 2NF [Yes/No]? Show your reasoning.

  Yes. There is no partial dependency.

- c-3> [4 pts] Does their relation satisfy 3NF [Yes/No]? Show your reasoning.

  Yes. There is no transitive dependency.
C. Expert(email, phonenum, pay_grade, pay_per_hour, topics) [21 pts]

(a) [10pts] List all of the functional dependencies involving the attributes of the relation in question. (Just give the basic list, not the closure of that list. ☺)

FD1: email → phonenum, pay_grade, pay_per_hour, topics
FD2: pay_grade → pay_per_hour

(b) [1 pts] What are the candidate keys for this relation?

email

(c) [10pts] Identify which normal form their relation is in by answering the following questions:

c-1> [2pts] Does their relation satisfy 1NF [Yes/No]? Show your reasoning.
No. The topics attribute is a multi-valued attribute.

c-2> [4pts] Does their relation satisfy 2NF [Yes/No]? Show your reasoning.
No. The relation is not in 1NF.

c-3> [4pts] Does their relation satisfy 3NF [Yes/No]? Show your reasoning.
No. The relation is not in 2NF.

(Additional note: Suppose the topic attribute is not a multi-valued attribute. Still, R is not in 3NF since FD2 violates the 3NF condition. The pay_per_hour attribute is dependent on a non prime attribute pay_grade: email → pay_grade → pay_per_hour.)
2. Normalization [35 pts]
For each question below, normalize the given relation into 3NF (or better) based on the given functional dependencies. Also, describe whether each of the resulting relations is in BCNF or not.

A. [15 pts]
R(A, B, C, D, E)
All attributes contain only atomic values.

FD1: A → E
FD2: (E, C) → B, D
FD3: D → C

(a) [5 pts] What’s the highest normal form that R satisfies and why?
1NF. FD1 violates the 2NF condition since the candidate keys are (A, C) and (A, D). (See below.)

(b) [5 pts] If R is not at least in 3NF, normalize R and show the resulting relation(s) and their candidate keys. Otherwise, just list the candidate keys of R.
Note: Both case 1 and case 2 are acceptable.

Case 1: 3NF

Step 1. Deal with FD1.
R1(A, B, C, D), R2(A, E)
Candidate Keys: R1(A, C), R1(A, D), R2(A)

Step 2. From transitivity and FD1 + FD2, we get
FD2':(A, C) → (B, D), or equivalently: (A, C) → B, (A, C) → D

Step 3. From FD3 and the above, we also get
FD2'':(A, D) → B

Step 4. B is the only non-prime attribute in R1 and it is not partially dependent on R1’s candidate keys. (Which are (A, C) and (A, D).) Therefore, the result of Step 1 (R1 and R2) is in 3NF.
Case 2: BCNF

From the above, we know that R1 is in 3NF and R2 is in BCNF.
R1(A, B, C, D), R2(A, E)
Candidate Keys: R1(A, C), R1(A, D), R2(A)

Since FD3 violates the BCNF condition (D is not a super key), we decompose R1 into two relations R3 and R4.

R3(A,B,D), R4(C,D), R2(A,E)
Candidate keys: R3(A,D), R4(D), R2(A)

(c) [5 pts] Is your answer to (b) in BCNF? Why or why not?
Note: Both case 1 and case 2 are acceptable as the answer for this question based on the answer to the (b).

Case 1: No. FD3 violates the BCNF condition in R1 since D is not a super key.
Case 2: Yes. For each functional dependency X → Y in each relation, X is a super key for that relation. Thus, each relation is in BCNF.
B. [20 pts]
R(A, B, C, D, E, F, G)
All attributes contain only atomic values.
FD1: (A, E) → B, C, D, F, G
FD2: E → F
FD3: A → B, C, D
FD4: C → D

(a) [5 pts] What’s the highest normal form that R satisfies and why?
1NF. FD2 violates the 2NF condition.

(b) [10 pts] If R is not at least in 3NF, normalize R and show the resulting relation(s) and their candidate keys. Otherwise, just list the candidate keys of R.
R(A, E, G), R1(E, F), R2(A, B, C), R3(C, D)
Candidate Keys: R(A, E), R1(E), R2(A), R3(C)

(c) [5 pts] Is your answer to (b) in BCNF? Why or why not?
Yes. For each functional dependency X → Y in each relation, X is a super key for that relation. Thus, each relation is in BCNF.
CREATE DATABASE IF NOT EXISTS cs122ahw2;
USE cs122ahw2;

-- Clean-up statements to allow for repeated script testing
DROP TABLE IF EXISTS Poster_Incentive;
DROP TABLE IF EXISTS User_Activity_Log;
DROP TABLE IF EXISTS Expert;

-- New tables:

-- Foreign Key constraints to the original tables are not enforced to simplify the table layout.

-- New table for recording incentive information for posters.
-- When a poster posts their first article, that poster will get 10 cents for that article.
-- Once a poster posts 5 articles, that poster will get an additional 10 cents for that article.
-- Once a poster reaches 10 articles, that poster will get an additional 150 cents.
-- The total attribute keeps the total_amount that a poster has received so far.
CREATE TABLE Poster_Incentive(
    pid INTEGER NOT NULL,
    aid INTEGER NOT NULL,
    incentive INTEGER NOT NULL,
    total INTEGER NOT NULL
);

INSERT INTO Poster_Incentive(pid, aid, incentive, total)
VALUES
    (1, 1, 10, 10),
    (1, 2, 0, 10),
    (1, 3, 0, 10),
    (1, 4, 0, 10),
    (1, 5, 10, 20),
    (1, 6, 0, 20),
    (1, 7, 0, 20),
    (1, 8, 0, 20),
    (1, 9, 0, 20),
    (1, 10, 150, 170),
    (2, 1, 10, 10);

-- New table for recording the most recent operation of each user.
-- The possible values of the op_type attribute are:
-- 1 - login, 2 - read, 3 - recommend an article, 4 - logout
CREATE TABLE User_Activity_Log (
    uid INTEGER NOT NULL,
    op_type INTEGER NOT NULL,
    op_time DATETIME
);
INSERT INTO User_Activity_Log(uid, op_type, op_time)
VALUES
    (1, 1, '2017-04-21 01:05:10'),
    (1, 2, '2017-01-26 01:08:35');

-- New table for recording information about Experts.  
-- Each expert has a unique e-mail address.  
-- The phonenum attribute is a phone number of each expert.  
-- A phone number can be shared by many experts. (e.g., a corporate phone number)  
-- The pay_grade attribute is set by the company and based on the level of each expert.  
-- A higher pay_grade means that the expert has greater knowledge of certain topics.  
-- Possible pay_grade values: A (most expensive), B, C, D, and F (least expensive).  
-- The pay_per_hour attribute is the hourly pay that the company pays for each expert  
-- based on their pay_grade. Currently, for grade A, the company pays $500 per hour;  
-- for B, it pays $250, and so on (C: $200, D: $100, F: $50).  
-- The topics attribute contains one or more topics that each expert has knowledge of.  
-- If there are multiple topics for an expert, then each topic is separated by a comma.  
CREATE TABLE Expert(
    email VARCHAR(30),
    phonenum VARCHAR(20),
    pay_grade CHAR(1),
    pay_per_hour INTEGER,
    topics VARCHAR(200)
);

INSERT INTO Expert(email, phonenum, pay_grade, pay_per_hour, topics)
VALUES
    ('a@expertgroup.com', '(321) 123-4567', 'A', 500, 'business, politics'),
    ('b@iamthebest.com', '(231) 987-2345', 'F', 50, 'sports, movies, music, tv'),
    ('c@bestonly.com', '(222) 111-3333', 'C', 200, 'technology'),
    ('d@bestonly.com', '(222) 111-3333', 'D', 100, 'technology, science');