CSCI 2141
W2013
Assignment 3, Due 9:35 am, February 20, 2013
Groups of up to 3 students permitted.

Submission instructions: Submit an electronic copy of your assignment (SCAN your work as a PDF if needs be, scanner located by tech support on 1st floor). Email your assignment to the TA, Savneet Arora (sarora@cs.dal.ca). In the subject line, use the following template: 2141 Assignment-3 (Banner ID(s))

In the following questions, the primary key fields are underlined. {attribute, attribute} indicates a repeating group of attributes. If you need to make assumptions about business rules, please note the assumptions made.

1. A table design is shown which violates 1NF. Give a design which will store the same information but which is in 1NF.

Person(personid, personname, {carid, carname})

A: Person(personid, personname), Car(carid, carname, personid fk)

2. Not necessarily in 1NF. Project(projid, other proj. fields, {empid, other emp. Fields}). Give two answers to the problem, one assuming that the relationship between employees and projects is many-to-one, and the other assuming the relationship is many-to-many.

A: project(projid, other project fields…)
employee(empid, other employee fields…, projid fk)

or:

project(projid, other project fields…)
employee-project(empid fk, projid fk, …)
employee(empid, other employee fields…)

3. A table design is shown which violates 2NF. Give a design which will store the same information but which is in 2NF.

Transaction(personid, timestamp, personname)

A: Person(personid, personname), Transaction(personid fk, timestamp)

4. A table design is shown which violates 2NFGive a design which will store the same information but which is in 2NF.

Consignment(<u>consignmentid</u>, <u>itemid</u>, itemcolor, itemprice)

A: Consignment(<u>consignerid</u>, <u>itemid</u> fk), Item(<u>itemid</u>, itemcolor, itemprice)

5. Not necessarily in 2NF. Appointment(<u>service provider, client</u>, client diagnosis, date, time). Observe that this may also have problems with the key fields, depending on what assumptions you make. If you detect problems, make and state your assumptions and define the keys accordingly.

A: This problem contains multiple assumptions to worry about. What if a client has multiple diagnoses and separate appointments for different problems? The date and time field may have to be in the key in order to distinguish between multiple appointments on the same day. Note that it may end up being convenient to assign an arbitrary ID number to appointments rather than dealing with multiple concatenated keys.

appointment(<u>service provider, client fk</u>, date, time)
client(<u>client</u>, client diagnosis)

or:

service provider(<u>service provider</u>, other service provider fields…)
appointment(<u>service provider fk, client fk</u>, date, time [date and time optionally in key] …)
client(<u>client</u>, client diagnosis)

6. A table design is shown which violates 3NF. Assume that department determines boss. Give a design which will store the same information but which is in 3NF.

Employee(<u>SSN</u>, empname, deptid, bossid)

A: Employee(<u>SSN</u>, empname, deptid fk), Department(<u>deptid</u>, bossid)

7. A table design is shown which violates 3NF. Assume that ownerid determines ownername and owneraddress. Give a design which will store the same information but which is in 3NF.

Vehicle(<u>VIN</u>, ownerid, ownername, owneraddress)

A: Vehicle(<u>VIN</u>, ownerid fk), Owner(<u>ownerid</u>, ownername, owneraddress)

8. Not necessarily in 3NF. Tour(<u>tourpackage</u>, destination, departure date, return date, guide name, guide nationality).

A: tour(<u>tourpackage</u>, destination, departure date, return date, guide name fk)
guide(<u>guide name</u>, guide nationality)

You might want a many-to-many relationship, but the original design did imply a one-to-many relationship.

9. A table design is shown which violates BCNF. Assume that each player can play many sports and vice-versa. Also assume that each coach coaches only one sport. Give a design which will store the same information but which is in BCNF.

Player-Sport(<u>playerid</u>, <u>sportid</u>, coachid)

A: Player-Coach(<u>playerid</u>, <u>coachid</u>), Coach-Sport(<u>coachid</u>, sportid)

8. A table design is shown which violates BCNF. Assume that a project may take place at more than one location at a time, but at any one location there is only one project going on at a time. Give a design which will store the same information but which is in BCNF.

Employee-Project(<u>empid</u>, <u>projid</u>, locid)

A: Emp-Loc(<u>empid</u>, <u>locid</u> fk), Loc-Proj(<u>locid</u>, projid)

9. Not necessarily in BCNF. Sale(<u>item class code, salesrep id, date</u>, price, custid). The first task here is dreaming up a business assumption that would cause this to violate BCNF. State the assumption and then fix the table.

A: This question is full of otherwise unsupportable assumptions. For example, only providing for one sale per salesrep per date is unrealistic. However, it is still useful in order to pose the question at hand. A possible violation of BCNF might be that the item class code is somehow dependent on price. For example, item class code might consist of a set of different ranges of prices, so knowing the actual price determines the item class. Another possible assumption would be that any given customer is assigned to one single salesrep. If so, then customer determines salesrep. Here is a solution based on that scenario.

Sale(<u>item, custid fk, date</u>, price)
Customer(<u>custid</u>, salesrep id fk)

This design assumes that item prices are negotiated for a particular sale. A more full-scale solution, incidentally assuming that prices are not negotiated and are an attribute of items, might be:

Item(<u>itemid</u>, price)
Sale(<u>itemid fk, custid fk, date</u>)
Customer(<u>custid</u>, salesrep id fk)

Salesrep(salesrep id, other salesrep fields)

10. Not necessarily in 4NF. Publications(pubid, readers, advertisers, other pub attributes). State explicitly the assumption that would make this violate 4NF and fix the table.

A: While it's true that a publication may have many readers and many advertisers, these multi-valued dependencies are not related to each other. A correct design might consist of the following set of tables:

reader(readerid, …)
reader-publication(readerid fk, pubid fk, …)
publication(pubid, …)
publication-advertiser(pubid fk, adid fk, …)
advertiser(adid, …)

11. Consider the following table. Following the steps of normalization (beginning with determining the functional dependencies), do what is necessary to take it to 1NF to 2NF and to 3NF. Once in 3NF, discuss whether or not it is also in BCNF, 4NF, and 5NF. If it is not, make the necessary changes. Assume that Dept# and Cust# construct the primary key in our initial table.

Unnormalized Table

Dept #	Dept Name	Location	Mgr Name	Mgr ID No.	Tel Extn.	Cust #	Cust Name	Date of Complaint	Nature of Complaint
11232	Soap Division	Cincinnati	Mary Samuel	S11	7711	P10451	Robert Drumtree	1998-01-12	Poor Service
						P10480	Steven Parks	1998-01-14	Discourteous Attendant

BCNF: A relation is in BCNF, if and only if, every determinant is a candidate key. A 3NF relation is NOT in BCNF if:

- Candidate keys in the relation are composite keys (they are not single attributes), and
- There is more than one candidate key in the relation, and
- The keys are not disjoint, that is, some attributes in the keys are common

The last three tables only have single attribute candidate keys, so cannot be in violation. The first table has a composite primary key (Dept#,Cust#) but no other candidate keys, so can't be in violation

4NF: A Table is in 4NF if and only if, for every one of its non-trivial multivalued dependencies X -->>Y, X is a superkey—that is, X is either a candidate key or a superset thereof. There are multivalued dependences iff X->Y and X->Z and Y and Z are independent of each other. Need at least 3 attributes in a table to have a 4NF multivalued dependences, so last 2 tables are fine. In first table, Date of complaint and nature of complaint are not independent; similarly for second table, Dept Name, location, MGR ID #, and Tel ext are not independent. So tables are in 4NF.

5NF: To be in 5NF, we need to isolate semantically related multiple relationships (separating many to many relationships). Designed to reduce redundancy in relational databases recording multi-valued facts by isolating semantically related multiple relationships. A table is said to be in the 5NF if and only if every join dependency in it is implied by the candidate keys. A join dependency implies there is a lossless non-additive decomposition into smaller relations.
   • lossless: no loss of tuples when relations are joined
   • non-additive: no spurious tuples generated when relations are joined
Fifth normal form is based on the concept of join dependence. If a relation has a join dependency then it can be decomposed into smaller relations such that one can rejoin these relations to reproduce the original relation. A 5NF relation does not have any join dependencies. Our tables do not contain semantically related many to many relationshiop and cannot be decomposed into smaller relations, so are in 5NF.

12. See Sales Order form below. Design the database to support it and bring the tables to 3NF. At each step of the way, describe any repetition of data, delete anomalies, insert anomalies, and update anomalies that remain.

Sales Order

Fiction Company
202 N. Main
Mahattan, KS 66502

CustomerNumber:	*1001*	**Sales Order Number:**	405
Customer Name:	*ABC Company*	**Sales Order Date:**	2/1/2000
Customer Address:	*100 Points*	**Clerk Number:**	*210*
	Manhattan, KS 66502	**Clerk Name:**	*Martin Lawrence*

Item Ordered	Description	Quantity	Unit Price	Total
800	*widgit small*	*40*	*60.00*	*2,400.00*
801	*tingimajigger*	*20*	*20.00*	*400.00*
805	*thingibob*	*10*	*100.00*	*1,000.00*
Order Total				*3,800.00*

Fields in the original data table will be as follows:

SalesOrderNo, Date, CustomerNo, CustomerName, CustomerAdd, ClerkNo, ClerkName, ItemNo, Description, Qty, UnitPrice

Normalization: First Normal Form

The new table is as follows:

SalesOrderNo, ItemNo, Description, Qty, UnitPrice

The repeating fields will be removed from the original data table, leaving the following.

SalesOrderNo, Date, CustomerNo, CustomerName, CustomerAdd, ClerkNo, ClerkName

These two tables are a database in first normal form

What if we did not Normalize the Database to First Normal Form?

Repetition of Data – SO Header data repeated for every line in sales order.

Normalization: Second Normal Form

The new table will contain the following fields:

ItemNo, Description

All of these fields except the primary key will be removed from the original table. The primary key will be left in the original table to allow linking of data:

SalesOrderNo, ItemNo, Qty, UnitPrice

Note: It is dangerous to treat price as dependent on item. Price may be different for different sales orders (discounts, special customers, etc.)

Along with the unchanged table below, these tables make up a database in second normal form:

SalesOrderNo, Date, CustomerNo, CustomerName, CustomerAdd, ClerkNo, ClerkName

What if we did not Normalize the Database to Second Normal Form?

- Repetition of Data – Description would appear every time we had an order for the item
- Delete Anomalies – All information about inventory items is stored in the SalesOrderDetail table. Delete a sales order, delete the item.
- Insert Anomalies – To insert an inventory item, must insert sales order.
- Update Anomalies – To change the description, must change it on every SO.

Normalization: Third Normal Form

The new tables would be:

CustomerNo, CustomerName, CustomerAdd

ClerkNo, ClerkName

All of these fields except the primary key will be removed from the original table. The primary key will be left in the original table to allow linking of data as follows:

SalesOrderNo, Date, CustomerNo, ClerkNo

Together with the unchanged tables below, these tables make up the database in third normal form.

ItemNo, Description

SalesOrderNo, ItemNo, Qty, UnitPrice

What if we did not Normalize the Database to Third Normal Form?

- Repetition of Data – Detail for Cust/Clerk would appear on every SO
- Delete Anomalies – Delete a sales order, delete the customer/clerk
- Insert Anomalies – To insert a customer/clerk, must insert sales order.
- Update Anomalies – To change the name/address, etc, must change it on every SO.

Completed Tables in Third Normal Form

Customers: CustomerNo, CustomerName, CustomerAdd
Clerks: ClerkNo, ClerkName
Inventory Items: *ItemNo*, Description
Sales Orders: SalesOrderNo, Date, CustomerNo, ClerkNo
SalesOrderDetail: SalesOrderNo, ItemNo, Qty, UnitPrice