210.02 - Data Normalization Lab (Revised 2/11/04)

Data Normalization Basics
This exercise consists of two exercises that will demonstrate your ability to normalize data using the first three normal forms.

Objectives
After completing this exercise, you will be able to:

- Discuss why a table is not in the First Normal Form
- Convert an unnormalized table into the First Normal Form
- Take a table that is in the First Normal Form and convert it to the Second Normal Form.
- Take a table that is in the Second Normal Form and convert it to the Third Normal Form.
- Identify table keys.

Assignment
Below are a series of two exercises along with a list of requirements for completing each. In the first activity, you will be required to read through each of the steps and convert an unnormalized table into a normalized table as you create an E-R diagram. In the second activity, you will be required to take an existing E-R diagram and create a database schema.

Exercise I. Normalization of the ORDERS Table

The purpose of this exercise is for you to demonstrate your ability to take an unnormalized table and convert it up through the third normal form. Upon completion of this exercise you will have:

- Discussed why the step one table is not normalized
- Converted the table in step one to the First Normal Form
- Converted the table created in step two into the Second Normal Form
- Converted the table created in step three into the Third Normal Form
- Identified all of the names, attributes, and keys associated with each of the tables

Exercise 1 Directions
To complete exercise one, you should do the following:

1. Read and complete each of the four steps identified under exercise one
2. Create a response for each step listed under exercise one
3. Create your response in MS Word
4. When appropriate, use the table feature within Word to create your tables.
5. Save the document as identified in the “Labs” section of the roadmap and upload the file in the course communication space drop-box.

Step 1
Consider the below table with records for three unique ORDER_IDs. Provide three reasons why we normalize tables.

ORDERS

ORDER_ID	DATE	CUST_ID	CUST_NAME	STATE	ITEM_NUM	ITEM_DESCRIP	QUANTITY	PRICE
1001	7/7	A004	George	VA	6531 7890 7956	Tape Table Pens	5 1 1	100.00 400.00 2.00
1002	7/8	A999	Ben	PA	3456	Stove	1	300.00
1003	7/9	A201	Tom	PA	4234 7956	Paper Pens	15 10	234.00 20.00

Step 2
First Normal Form prohibits repeating groups, such as, ITEM_NUM, ITEM_DESCRIP, and PRICE. To convert the table to First Normal Form:

1. Remove the repeating groups from the base table.
2. Create a new table with the Primary Key of the base table and the repeating group.

Assuming that a situation might arise where an order could have multiple customers (i.e., multiple people receiving the items), the attributes of {ORDER_ID, CUST_ID} taken together are the primary key for the relation. The functional dependencies are:

ORDER_ID → DATE, ITEM_NUM, QUANITY
CUST_ID → CUST_NAME, STATE
ITEM_NUM → ITEM_DESCIP, PRICE

Convert the table shown in Step 1 to the First Normal Form using the below table.

Table Name:

Column Name									
Key Type									
Sample Data									

Step 3

Second Normal Form removes non-key columns that are not dependent upon the table's entire primary key. The steps to accomplish this are:

1. Determine which non-key columns are not dependent upon the table's entire primary key.
2. Remove those columns from the base table.
3. Create a second table with those columns and the column(s) from the Primary Key that they are dependent upon.

To be in Second Normal Form we remove *partial key dependencies,* or, in other words, each column must be dependent upon the entire primary key. Also, any table with a single column primary key is automatically in Second Normal Form.

Convert the table created in Step 2 to the Second Normal Form. Provide the Table name, column names, and appropriate key types (Note: you may not use all of the below empty tables).

Table Name:

Column Name									
Key Type									

Table Name:

Column Name									
Key Type									

Table Name:

Column Name									
Key Type									

Step 4
Third Normal Form removes *transitive dependencies,* or, in other words, any columns that are dependent upon another non-key column. The steps to accomplish this are:

1. Determine which columns are dependent upon another non-key column.
2. Remove those columns from the base table.
3. Create a second table with those columns and the non-key column that they are dependent upon.

Convert the table created in Step 3 to the Third Normal Form. Provide the Table name, column names, and appropriate key type.

Table Name:

Column Name									
Key Type									

Table Name:

Column Name									
Key Type									

Table Name:

Column Name									
Key Type									

Table Name:

Column Name									
Key Type									

Exercise II - Normalization of the CAR_SALE Table.

The purpose of this exercise is for you to demonstrate your ability to take a database schema and convert it up through the Third Normal Form. Upon completion of this exercise you will have:

- Listed the functional dependencies for a database schema
- Explained why a specific schema is not in Second or Third Normal Form
- Normalized a given schema into the Third Normal Form

The below scenario has been created to help you determine the table structures required for each of the subsequent normalized tables.

Scenario

You are given the database schema for a car sales database as follows:

Table Name: CAR_SALE

Column Name	Car _num	Date _sold	Salesman	Commission _percent	Discount _amt
Key Type	Primary		Primary		

Assuming that a car can be sold by multiple salesmen and, therefore, the attributes of Car_num and Salesman {Car_num, Salesman} taken together are the primary key for the relation. In addition, you are told that the date the car sells determines the discount amount and that each salesman has a unique commission rate.

Exercise 2 Directions

To complete exercise one, you should do the following:

1. Read and complete each of the three steps identified under exercise two
2. Create a response for each step listed under exercise two
3. Create your response using MS Word
4. When appropriate, use the table feature within Word to create your tables.
5. Save the document as identified in the "Labs" section of the roadmap and upload the file in the course communication space drop-box.

Step 1

List the functional dependencies in the relation CAR_SALE. Based on the given primary key, decide if the dependency is (1) completely dependent on the primary key (primary key dependency), (2) partially dependent on the primary key (partial key dependency), or (3) dependent on a non-key column (transitive dependency) for each of the Functional Dependencies you list.

Step 2

Explain why the relation CAR_SALE is not in 2NF or 3NF. (be specific).

Step 3
Normalize the relation CAR_SALE into 3NF. Show your results by providing the resulting table schemas.

Resources
To complete the above two activities, you may need to reference the IST 210 course content.

Grading Rubric

Checklist	**Possible Points**	**Points Scored**
Exercise 1		
- Provided three reasons why we normalize tables	3 pts	
- Converted the unnormalized table shown in Step 1 to the First Normal Form	3 pts	
- Converted the table created in Step 2 to the Second Normal Form and provided the Table name, column names, and appropriate key types.	3 pts	
- Converted the table created in Step 3 to the Third Normal Form and provided the Table name, column names, and appropriate key type.	4 pts	
Exercise 2		
- Listed the functional dependencies in the relation CAR_SALE	4 pts	
- Explained why CAR_SALE is not in 2NF or 3NF	4 pts	
- Normalized the relation CAR_SALE into 3NF	4 pts	
Total	**25 pts**	