**TSQL Homework 01**

**Question 1**

How does the book describe the difference between *imperative* and *declarative* languages?

**Answer**

Declarative language provides you with your request in an English-like manner. It’s basically you telling the computer what you want it to do. Where as Imperative language is you telling the computer exactly what to do. Imperative language uses a sequence of statements to determine how to reach a certain goal.

**Question 2**

List three categories of command statements in SQL.

**Answer**

Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL)

**Question 3**

Give an informal definition of *database* as used in the expression ”relational database management system.” Give an informal definition of *database* as used in the expression “Human Resources database.”

**Answer**

Database used in relational database management system gives you access to as concurrent transactions modify and query it. A collection of data on a computer. Database used in human resources database is a collection of data that humans have made. Things like payroll, time cards, benefits, etc.

**Question 4**

The book states that “the goal of the relational model is to enable consistent representation of data with minimal or no redundancy and without sacrificing completeness…” Briefly state your understanding of *minimal* *or no redundancy* and *completeness.* Why do you thin that these values are important?

**Answer**

From understanding they want to be able to present data in its completed form without having to worry about it crashing. They want to build it as easily but as efficiently as possible.

**Question 5**

What is the difference between two-valued logic, three-valued logic, and four-valued logic? How does SQL implement three-valued predicate logic?

**Answer**

Two-valued predicate logic the predicate is either false or true, if it is not false then it is true. Three-valued predicate logic is where the predicate is sometimes missing some values, so yes it can be true or false, however if it is missing some values it can also be unknown. With four-valued predicate logic it can be true, false, missing but applicable, or missing and inapplicable.

**Question 6**

How does SQL enforce *entity integrity*? What is entity integrity?

**Answer**

Entity integrity is the mechanism the system provides to maintain primary keys. Candidate keys provide entity integrity. A candidate key is a key defined on one or more attributes that prevents more than one occurrence of the same tuple (row in SQL) in a relation.

**Question 7**

How doe SQL enforce *referential integrity*? What is referential integrity?

**Answer**

Referential integrity is enforced by foreign keys. It refers to the accuracy and consistency of data within a relationship. Most things are based off this. You have two items, item a and item b. Item b depends on item a. There is no item b unless there is an item a first. If something were to happen to item a you would need to do something with item b.

**Question 8**

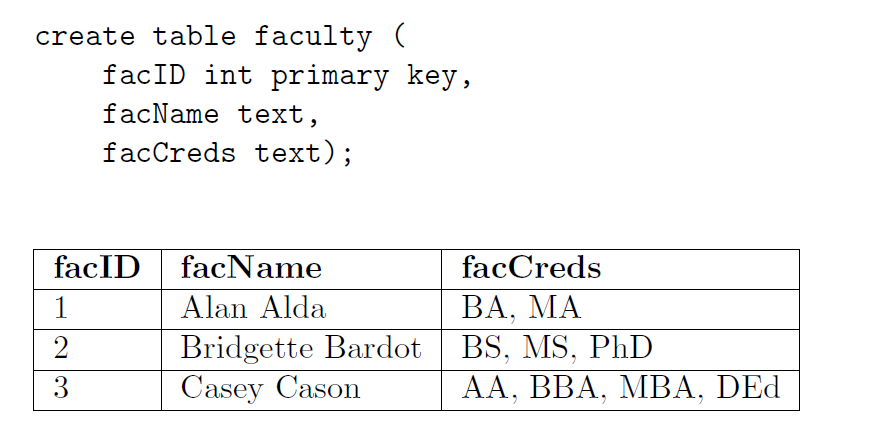
What is a *relation* as defined in the textbook? A one word answer to this question is sufficient.

**Answer**

A set or table.

**Question 9**

Is this table in first normal form? Why or why not? If it is not, how would you change it?

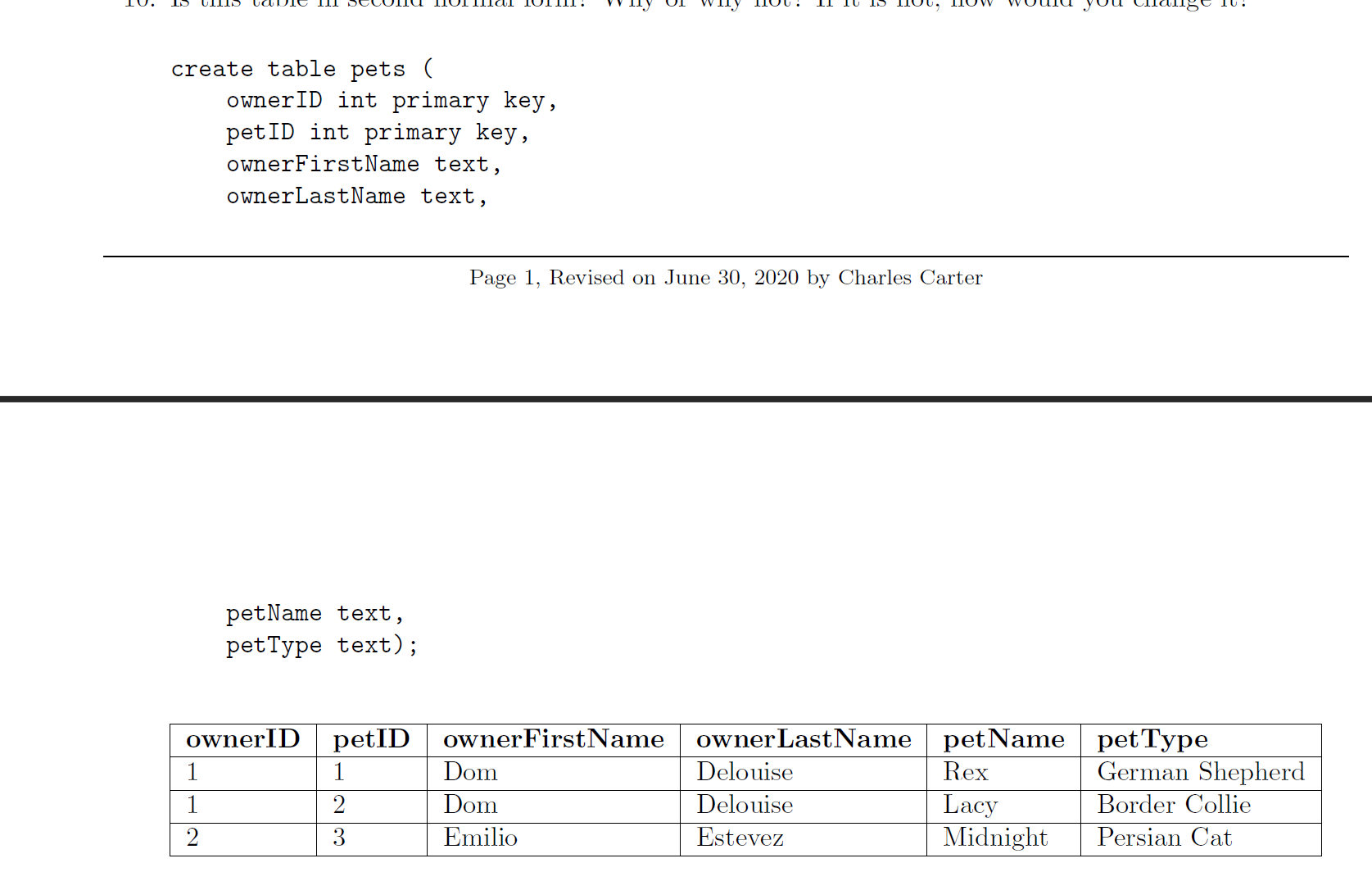


**Answer**

It is not. In the first normal form each cell has to be atomic. The first normal form says that the tuples (rows) in the relation (table) must be unique and attributes should be atomic. This is a redundant definition of a relation; in other words, if a table truly represents a relation, it is already in first normal form.

**Question 10**

Is this table in second normal form? Why or why not? If it is not, how would you change it?

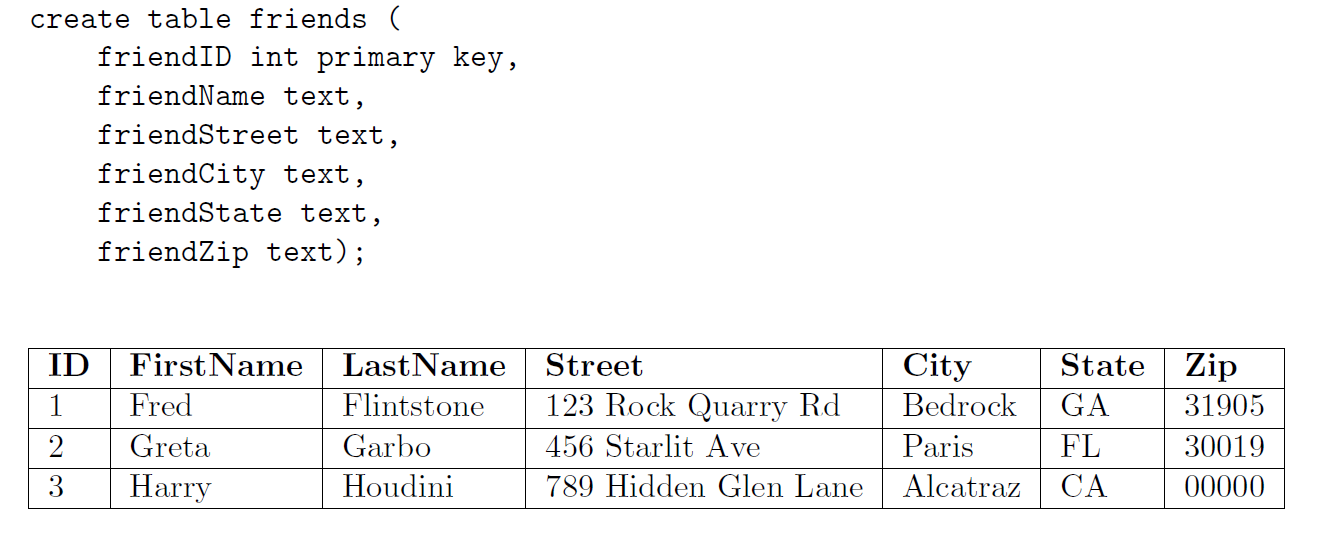


**Answer**

It is not. Second Normal Form has two rules. The first rule is it must meet the standards of the first normal form. The second rule is that it addresses the relationship between nonkey and candidate-key attributes. For every candidate key. In other words, a nonkey attribute cannot be fully functionally dependent on part of a candidate key. If you need to obtain any nonkey attribute value, you need to provide the values of all attributes of a candidate key from the same tuple. You can find any value of any attribute of any tuple if you know all the attribute values of a candidate key.

**Question 11**

Is this table in third normal form? Why or why not? If it not, how would you change it?



**Answer**

It is not. Third normal form has two rules also. The first being it must meet the standards of the second normal form, which means it must also meet the standards of the first normal form. With the third normal form all nonkey attributes must be dependent on candidate keys nontransitvely. This rules means that all nonkey attributes must be mutually independent. One nonkey attribute cannot be dependent on another nonkey attribute.

**Question 12**

List the components of a *four-part object name?*

**Answer**

The components are the instance name, database name, object name, and the table name.

**Question 13**

What is the difference between *declarative data integrity*  and *procedural data integrity?*

**Answer**

Declarative data integrity is enforced as part of the model-namely, as part of the table definitions. You define it when you define your table. Procedural data integrity is enforced with code, such as with stored procedures or triggers.