Lesson 3: Database Schema Implementation

## INTRODUCTION

## In this lesson you will learn about database schema implementation. You will view information on domains and the importance of data validation. The information explains various constraints and the importance they have in validating data. You will also learn how to identify and describe primary and foreign keys as well how each constraint is used.

## LESSON OBJECTIVES

By the end of this lesson, you will be able to:

1. Define key terms relating to implementing database schemas.

2. Implement subtypes and domains, primary and foreign keys, data validation, relationships, and other geodatabase tools.

## LEARNING SEQUENCE

|  |  |
| --- | --- |
|  | |
| Required Reading | Read the following:  Database Schema Implementation   * Domains * Primary Key * Foreign Key * Relationships |
| Assignments | Complete the following assignments:   * Lab 3: Database Schema Implementation * Quiz 3: Database Schema Implementation |

## INSTRUCTION

## Domains

### What are Domains?

The domain set is a list of possible values for an attribute. A domain can be assigned to multiple attributes in any table. A domain can be an exhaustive list of possible values, or a rule defining the allowed values.

### Examples of Domains

Consider some examples of domains. The list in our first domain contains all the possible salutations of people. In the database column that stores salutations, a value such as Mr., Mrs., Dr., or Miss could be added to that attribute as they are all members of the salutations domain. We cannot use both Mr. and Mrs. in a single field for each person can only have one salutation therefore that would be a restriction of this domain.

Another example would be a rule to define appropriate email addresses. In this rule, all email addresses must use the ‘@’ symbol placed between characters and a dot symbol followed by characters at the end of the email address. For the final example of a domain, consider phone numbers. Many databases require the use of 10 digits to create a phone number.

The reason we use domains for attributes is to ensure data integrity. Ensuring that the data entered into an attribute matches a domain of values allows us to defend against improper entries in the database.

## Data Validation

### The Importance of Data Validation

Information in databases are constantly updated, deleted, queried, and moved. We use data validation to separate the good data from the bad data and to keep the data consistent through all of these database modifications. There are key factors that we will discuss that will help us keep our data valid such as constraints, referential integrity, delete and update. Data validation can be implemented during the design process of the database.

### Check Constraints

The first step in data validation is to check constraints. A check constraint makes certain that a statement about the data is true for all records in the table and enforces domain integrity by limiting the values that are accepted by specific columns in the table. Check constraints can be created using a Boolean expression where the expression will check the value entered into a cell against the list of domain values or a rule.

If this expression returns false, meaning that the entry does not meet the rules of the domain, then the value that is being entered will be rejected. As an example, if our domain rule states that the population entered in the cell must be greater than or equal to 10,000 and it must also be less than 500,000 then if we try to enter a value of 1 million into the cell it will be rejected because it does not meet the domain rule.

#### Unique Constraint

### A unique constraint ensures that there are no duplications for any records in a specific column. If the values of the column should be unique, however and it is not being used as the primary key, the unique constraint is appropriate to use.

#### Not Null Constraint

### The not null constraint can be placed on a column to enforce that the field cannot be left blank upon creation of the row. Additionally, null values will never be accepted in that column as long as the constraint is set to not null.

#### Primary Key Constraints

The primary key constraint is a mixture of the unique and not null constraints. The primary key constraint can be assigned to the column that is to be used as the primary key for the table. When the primary key constraint is in use, no two records can have the same value with the primary key column, and no null values be will be accepted in that column.

### Setting Validation Requirements

### When setting data validation requirements you can set validation rules, data types, and required properties.

#### Validation Rules

A validation rule property allows the designer to set a validation rule which forces data inputted to follow the specified rule. Setting the phone number field requiring the entry of 10 digits is considered a validation rule property. In addition to setting the rule, the designer can use a rule text that displays a message if the user enters invalid information.

#### Data Types

A data type can be set as a validation rule which restricts the type of data that can be entered into a column. If the designer sets a data type to numeric then a number must be entered and all other characters will be denied.

#### Required Properties

The required property helps to avoid null values and unwanted errors. If a user attempts to leave a value blank, they can be prompted with an error message asking them to enter a value. For example, primary keys are required; therefore a value must be entered to complete the record.

## Primary Key

### Creating a Primary Key

There are constraints to consider when creating a primary key. A table can only contain one primary key. The primary key column cannot accept null values and also requires that the data values be unique for that table and column. A primary key is frequently defined on some sort of column containing identity information such as an employee number, product identification number, and so on.

When determining which column is defined as the primary key you should always keep in mind that the main function of a primary key is to identify a row within a table. It is not necessarily meant to describe the record it is only meant to identify. However, it is possible that the primary key cannot be some sort of descriptor of the record so long as the descriptor is always unique. Besides uniquely identifying a record in a table, the primary key also works with foreign keys to allow for relationships between tables.

### Good and Bad Primary Keys

Some good primary key choices would be an employee identification number, product identification number, sales receipt number, or ticket number. These are all good choices because they logically make sense to be unique, they do not contain any private information, and do not require a significant amount of memory to store.

Examples of bad primary key choices are ZIP Codes if the purpose of the ZIP Code was to uniquely identify cities. This is a bad idea as many cities may have the same ZIP Code. A Social Security number is potentially a good primary key although it is considered private information and people may not be willing to use their Social Security number as a lookup mechanism. Some people may not have a Social Security number, and Social Security numbers are often reused after death which would allow for one Social Security number to link to multiple people.

## Foreign Key

### Why use a Foreign Key?

The purpose of creating a foreign key is to create a link between two tables when the columns of both tables hold the primary key value for one table and is referenced by the foreign key column of another table. A foreign key constraint forces the value entered as a foreign key to be equal to that of the primary key value used in another table.

### Characteristics

When creating a foreign key keep in mind that the foreign key for one table is the primary key for another table. This means that the foreign key value must be the same as the value of a primary key in another table which you wish to have it related to. All guidelines for naming conventions, characteristics, etc., for the primary key will all apply to the foreign key.

## Relationships

### What are Relationships?

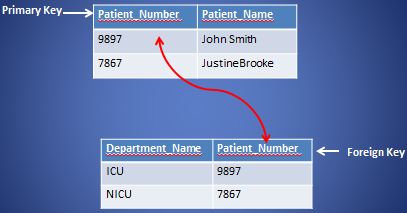
A relationship is an association between entities or records. To establish a relationship between entities, the entities need to have some sort of relatable primary key and foreign key. Participants of a relationship are those entities that are in a relationship with others whether in the same table or in a separate table.

### Implementing 1:1 Relationships

To implement a 1:1 relationship, one table should have a key designated as the primary key, and the relating table should have a foreign key. Using the foreign key, a table can relate back to the table that has the associated primary key and vice versa.

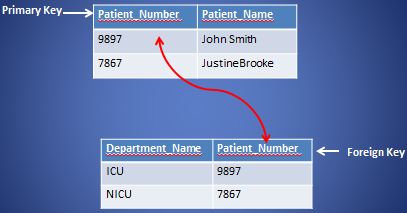
As an example, the table on the top of the image is the patient table which contains the patient number primary key column in the patient name column. In the bottom table, the department name is the primary key for the table, and the patient number is the foreign key which relates back to the primary key in the patient table.

This allows the patient table to be a sort of lookup table. In other words, looking at the department name, we can get a patient number, and then reference that patient number back to the patient table to determine the patient’s name. This keeps us from having to enter the patient’s name inside the department table thereby reducing the amount duplication, and maintaining data integrity.



### Implementing 1: Many Relationships

Many foreign key entries are dependent upon a single primary key. To implement a 1: many relationship, one table has a single primary key that can map to other tables that have multiple entries of its related foreign key.



## SUMMARY

## This lesson taught you about Database Schema Implementation. You learned information on domains and the importance of data validation. The information explained various constraints and the importance they have in validating data. You also learned how to identify and described primary and foreign keys as well how each constraint is used.

## ASSIGNMENTS

1. Lab 3: Database Schema Implementation
2. Quiz3: Database Schema Implementation