Joyce Woznica

jlwoznic@syr.edu

Joyce Woznica  
Database Project

Horse Records

Contents

[Project – Part 1 3](#_Toc532303038)

[Part 1 – Project Overview and Entity Diagrams 3](#_Toc532303039)

[Updated Summary 3](#_Toc532303040)

[Background 3](#_Toc532303041)

[Entities 3](#_Toc532303042)

[Relationships 4](#_Toc532303043)

[Miscellaneous Information 4](#_Toc532303044)

[Example Real Data 4](#_Toc532303045)

[Entity Diagram 5](#_Toc532303046)

[Normalized Model 6](#_Toc532303047)

[Project – Part 2 6](#_Toc532303048)

[Physical Database Design 6](#_Toc532303049)

[Create Tables 6](#_Toc532303050)

[Drop Tables 9](#_Toc532303051)

[Data Creation 9](#_Toc532303052)

[Data Manipulation 14](#_Toc532303053)

[Adding some Data to the Tables 14](#_Toc532303054)

[Functions, Procedures, Views, and More 15](#_Toc532303055)

[Answering Data Questions 18](#_Toc532303056)

[When is a Horse Due for a Specific Vaccine (*WhenDue.sql*) 18](#_Toc532303057)

[Which Vet Performed the Last Vaccination of Vaccine (*WhichVet.sql*) 19](#_Toc532303058)

[When was a Vaccine Performed (*WhenWasVaccine.sql*) 20](#_Toc532303059)

[How much money was spent on vaccines in a given year (*HowMuchForYear.sql*) 21](#_Toc532303060)

[Average Price for a Specific Vaccine (*AveragePrice.sql*) 21](#_Toc532303061)

[How much money was spent on vaccines in a given year for a horse (*TotalPerHorsePerYear.sql*) 22](#_Toc532303062)

[General Maintenance 23](#_Toc532303063)

[Implementation 23](#_Toc532303064)

[Phase 1 – General Screens using eForm Tool 23](#_Toc532303065)

[Phase 2 – Tableau as a User Design Tool 25](#_Toc532303066)

[Reflection 28](#_Toc532303067)

List of Figures

[Figure 1: Horse Record Database Entity Diagram 5](#_Toc532303068)

[Figure 2: Horse Record Database Normalized Entity Diagram 6](#_Toc532303069)

[Figure 3: Create Clinic Table SQL Commands 7](#_Toc532303070)

[Figure 4: Create Vet Table SQL Commands 7](#_Toc532303071)

[Figure 5: Create Barn Table SQL Commands 7](#_Toc532303072)

[Figure 6: Create Horse Table SQL Commands 8](#_Toc532303073)

[Figure 7: Create Vaccine Table SQL Commands 8](#_Toc532303074)

[Figure 3: Create HorseVaccine Table SQL Commands 8](#_Toc532303075)

[Figure 9: Drop Tables in Required Order SQL Commands 9](#_Toc532303076)

[Figure 10: Inserts to Load the Vaccine Table 12](#_Toc532303077)

[Figure 11: Inserts to Load the Barn Table 12](#_Toc532303078)

[Figure 12: Inserts to Load the Horse Table 12](#_Toc532303079)

[Figure 13: Inserts to Load the Clinic Table 13](#_Toc532303080)

[Figure 14: Inserts to Load the Vet Table 13](#_Toc532303081)

[Figure 15: Some Representative Inserts to Load the HorseVaccine Table 13](#_Toc532303082)

[Figure 16: AddVet INSERT statements to Add Data to Vet and Clinic Tables 14](#_Toc532303083)

[Figure 17: AddDentistandChiro INSERT statements to Add Data to Vaccine Table 14](#_Toc532303084)

[Figure 18: MakeBarnNameandVaccineNameUnique ALTER TABLE statements 15](#_Toc532303085)

[Figure 19: RenameScheduleDate Execution of MSSQL Function to Modify the HorseVaccine Table 15](#_Toc532303086)

[Figure 20: NumMonths Function to Determine NextVaccineDate 16](#_Toc532303087)

[Figure 21: VaccineSchedules Stored Procedures to Allow Updates of the Vaccine Table 16](#_Toc532303088)

[Figure 22: RemoveExtraVaccine UPDATE, DELETE to Update Vaccine, HorseVaccine Tables 17](#_Toc532303089)

[Figure 23: UpdateNextVaccineDate Procedure with loop to Update tables 17](#_Toc532303090)

[Figure 24: UpdateMicrochip Stored Procedure to add Microchip data to the Horse Table 18](#_Toc532303091)

[Figure 25: WhenDue Function returns when a horse is due for the noted Vaccine 19](#_Toc532303092)

[Figure 26: WhichVet Function returns which Vet did a Horse’s last Vaccine 20](#_Toc532303093)

[Figure 27: WhenWasVaccine Function returns when a horse is got the last dose of Vaccine 20](#_Toc532303094)

[Figure 28: HowMuchForYear Function returns a currency value 21](#_Toc532303095)

[Figure 29: AveragePrice Function returns a currency value 21](#_Toc532303096)

[Figure 30: TotalPerHorsePerYear Function returns a currency value 22](#_Toc532303097)

[Figure 31: AddHorseVaccine Stored Procedure that Updates the HorseVaccine Table 23](#_Toc532303098)

[Figure 32: Determine when Horse is Due for Vaccine 24](#_Toc532303099)

[Figure 33: After Clicking “When Due” Response displayed 24](#_Toc532303100)

[Figure 34: Determining How Much was Spent on Vaccines for a Particular Horse in a Year 24](#_Toc532303101)

[Figure 35: Form to Add a Vaccine to the Database (Date and Cost are defined as date and float) 25](#_Toc532303102)

[Figure 36: Worksheet to Display Vaccines by Horse by Vaccine Date 25](#_Toc532303103)

[Figure 37: Worksheet to Display Horse and their Barn Location by Barn Name 26](#_Toc532303104)

[Figure 38: Dashboard of Various Worksheets 26](#_Toc532303105)

[Figure 39: Vet, Clinic, Barn and Horse Dashboard 27](#_Toc532303106)

[Figure 40: Vaccine Dashboard 28](#_Toc532303107)

# Project – Part 1

## Part 1 – Project Overview and Entity Diagrams

### Updated Summary

For my project, I elected to create a database that will help me keep track of my horses’ vaccination requirements, dental visits, chiropractor, and other medical records. The information in this database will allow me to answer the following types of inquiries:

* When is <HORSE> due for a <VACCINE>?
* What vet did <HORSE>’s last <VACCINE>?
* When did <HORSE> get the last dose of <VACCINE>?
* How much money did I spent on vaccine(s) in <YEAR>?
* What is the average price for <VACCINE>?
* How much was spent on <HORSE> for vaccines in <YEAR>?

This data is very important and could actually prove to be useful in the industry in general (for other horse owners).

### Background

Certain vaccines are due every 6 months, some once per year, dental visits are usually semi-annual where the chiropractor visits are quarterly. Each horse might get a different batch of a specific vaccine and this can be important if there is a reaction of some kind to the vaccine.

It is very hard to keep track with multiple horses getting vaccines in different locations from different vets. It is important for all horse owners to keep track of when a horse is vaccinated and when he/she is due for his/her next vaccine. It is very important not to miss the proper vaccines as it puts not only that horse in jeopardy, but any horses they come in contact with at their own location and any shows that they might attend.

## Entities

* Horse
  + Horse Microchip
  + Horse Name
  + Show Name
  + Foal Date
  + Breed
* Barn
  + Barn Name
  + Barn Address
  + Barn City
  + Barn State
  + Barn Zip
  + Barn Phone
* Vaccine
  + Name
  + Cycle
* Veterinarian
  + Last Name
  + First Name
  + Cell Phone
  + License Number
  + Clinic Associated
* Vet Clinic
  + Clinic Name
  + Clinic Address
  + City
  + State
  + Zip
  + WebSite
* Horse/Vaccine
  + Horse
  + Vaccine
  + Batch
  + Date Given
  + Vet Used
  + Clinic Used

## Relationships

* One horse can have 0 to many vaccines.
* One veterinarian can provide 0 to many vaccines to 0 to many horses.
* One clinic can have one or more veterinarians that gives vaccines.
* A veterinarian has to provide all vaccines.
* One clinic can have one to many veterinarians.
* One horse can be located at one and only one barn.
* One barn can have 0 to many horses.

One of the most important relationships is where everything comes together mapping the vet, the clinic, the horse, the vaccine and the dates and batches.

## Miscellaneous Information

* The date the vaccine is given and the “batch” is very important as all the scheduling and future dates stem from this information.
* Each vaccine should have an “cycle” with it to note how often it should be given. Possibly cycles used for this project are “Annual,” “Semi-Annual,” “Quarterly”.
* It is important to know the due date of the next vaccine.
* Costs can also be captured to determine if there are vets that might charge a lesser price.

## Example Real Data

I have moved all my data to the spreadsheet included with this submission. I used the techniques with sorting data in Excel to help normalize my data.

### Entity Diagram

Note: Changed to reflect changes to database column names and primary keys

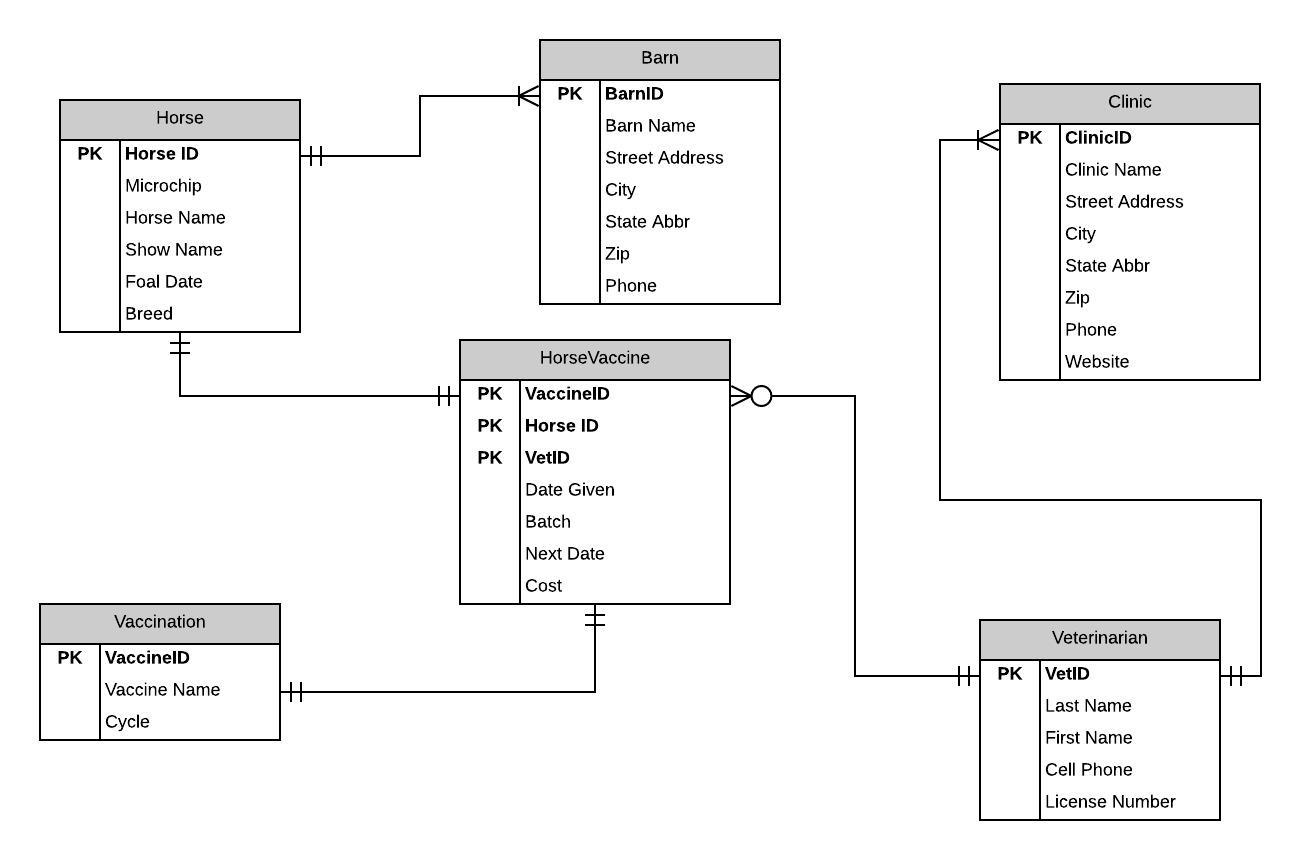


Figure 1: Horse Record Database Entity Diagram

## Normalized Model

Note: Changed to reflect changes to database column names and primary keys

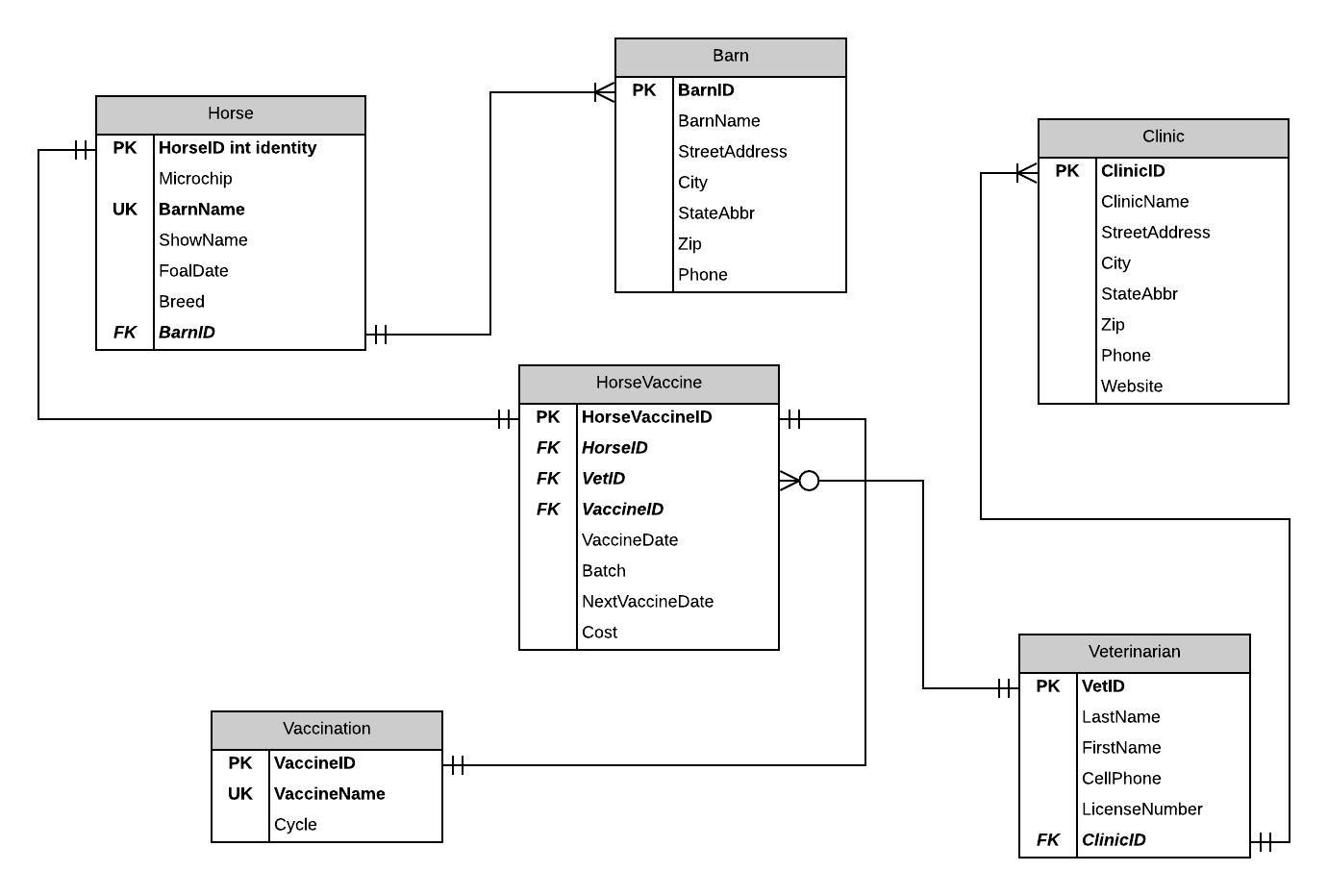


Figure 2: Horse Record Database Normalized Entity Diagram

# Project – Part 2

## Physical Database Design

### Create Tables

The physical database design is shown in the Create Table SQL code that appears in the following figures. However, please note that one table was changed (***HorseVaccine***) to provide some clarity around the column *VaccineScheduledDate.*

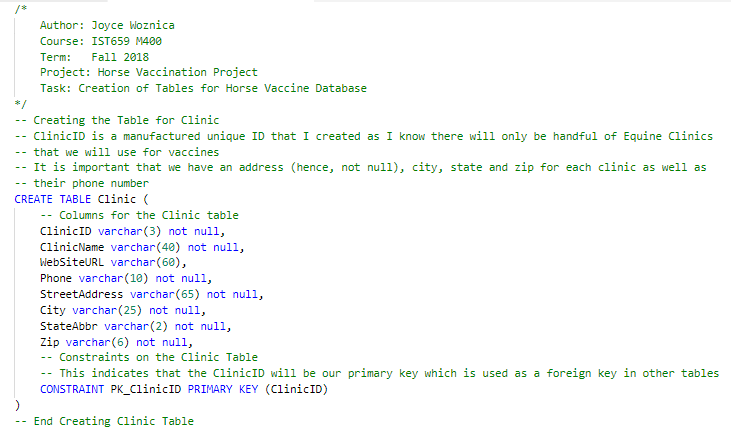


Figure 3: Create Clinic Table SQL Commands

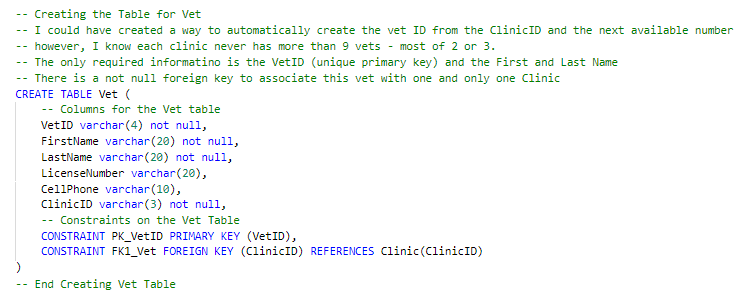


Figure 4: Create Vet Table SQL Commands

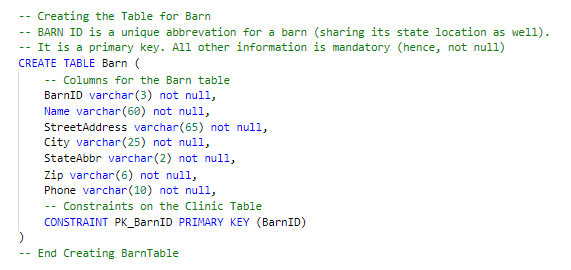


Figure 5: Create Barn Table SQL Commands

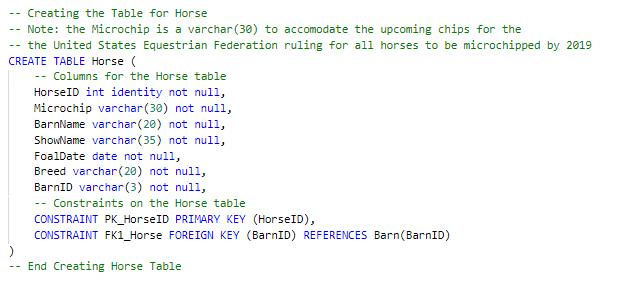


Figure 6: Create Horse Table SQL Commands

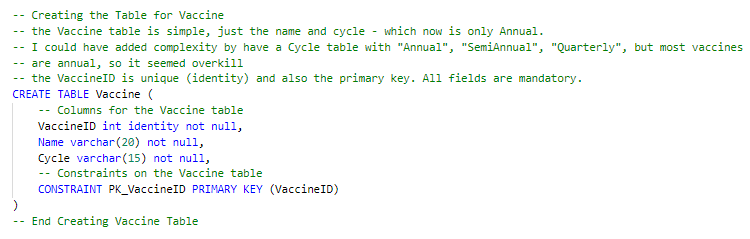


Figure 7: Create Vaccine Table SQL Commands

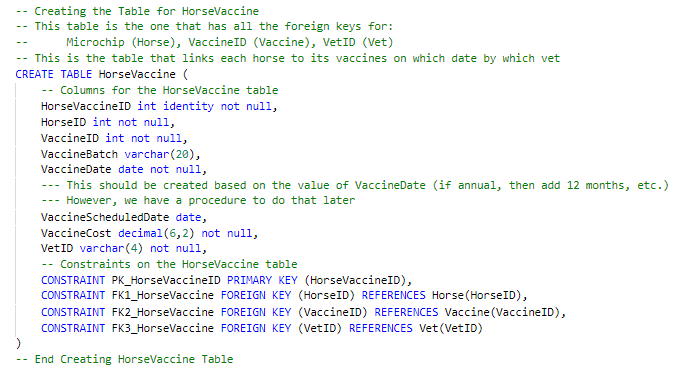


Figure 8: Create HorseVaccine Table SQL Commands

### Drop Tables

For removing the data, I created the following SQL file, *DropTables4Project.sql* which is shown below. This file represents the order that tables must be dropped due to dependencies. I tested these drop statements multiple times by creating the tables and then dropping them and then recreating.

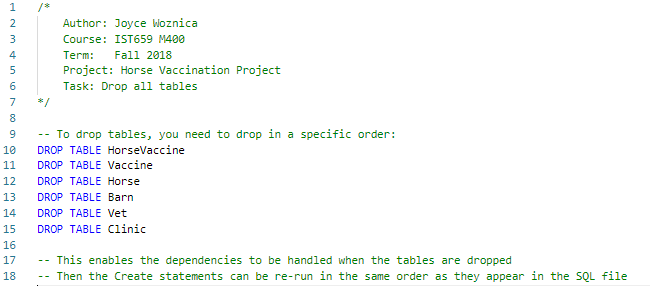


Figure 9: Drop Tables in Required Order SQL Commands

## Data Creation

I created insert files for each individual table so it was easier for me to open and close files when needed. I did these in a specific order to verify my dependencies were correct and the data was correct.

* *01-InsertintoVaccine.sql*
* *02-InsertintoBarn.sql*
* *03-InsertintoHorse.sql*
* *04-InsertintoClinic.sql*
* *05-InsertintoVet.sql*
* *06-InsertintoHorseVaccine.sql*

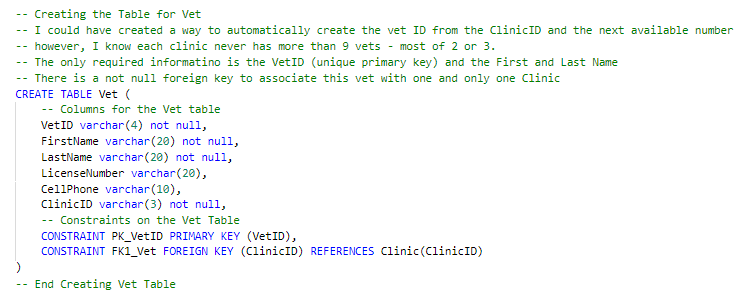
In generating these INSERT statements, I made some mistakes and had to drop tables and recreate them (see above in the section entitled “

Figure 4: Create Vet Table SQL Commands

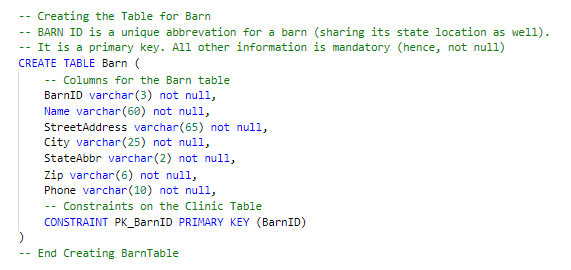


Figure 5: Create Barn Table SQL Commands

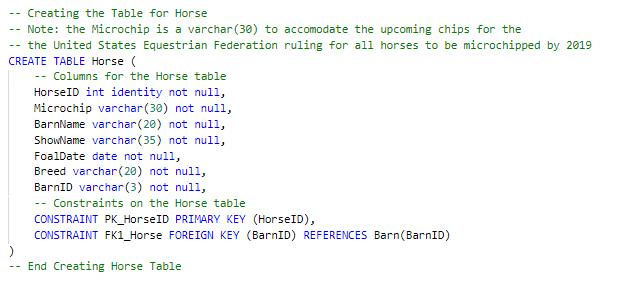


Figure 6: Create Horse Table SQL Commands

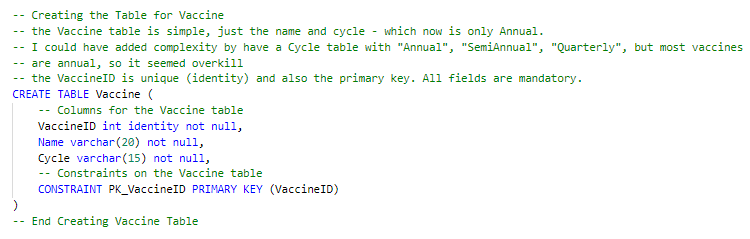


Figure 7: Create Vaccine Table SQL Commands

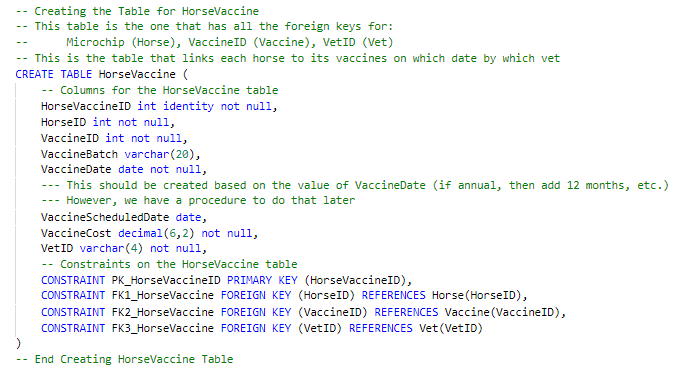


Figure 8: Create HorseVaccine Table SQL Commands

Drop Tables on page 7). Here are some excerpts from these files.

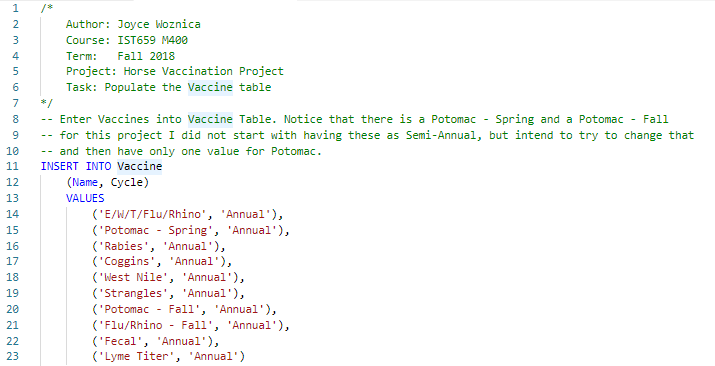


Figure 10: Inserts to Load the Vaccine Table

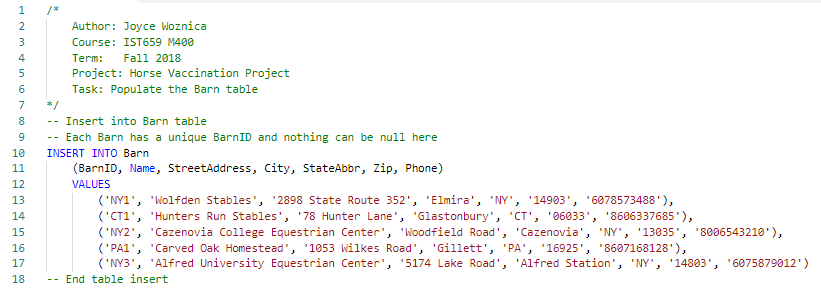


Figure 11: Inserts to Load the Barn Table

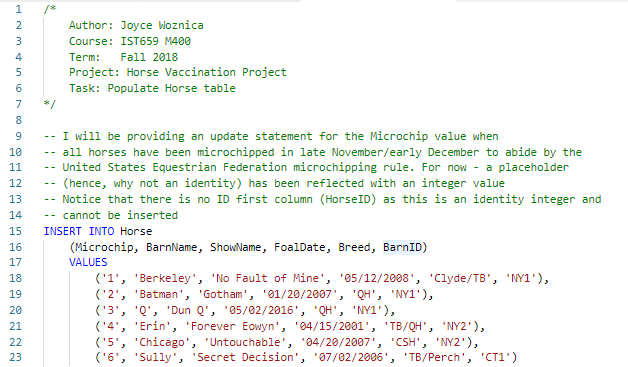


Figure 12: Inserts to Load the Horse Table

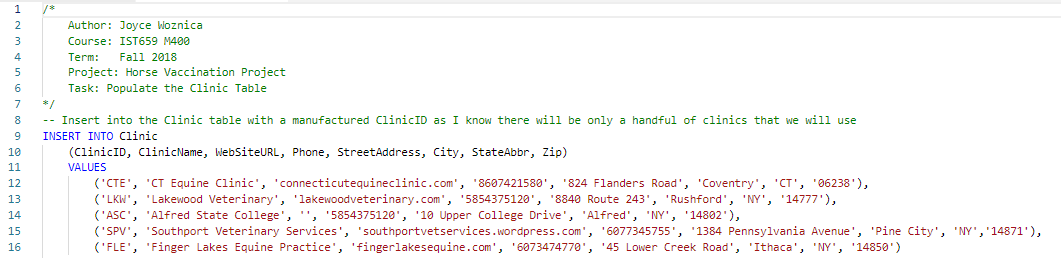


Figure 13: Inserts to Load the Clinic Table

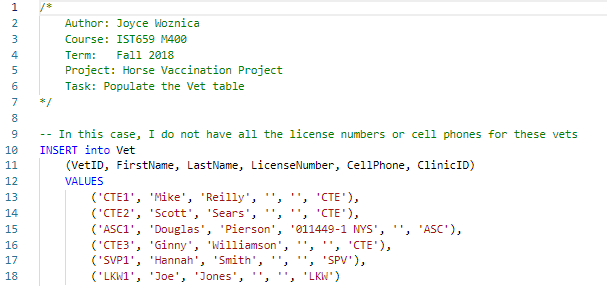


Figure 14: Inserts to Load the Vet Table

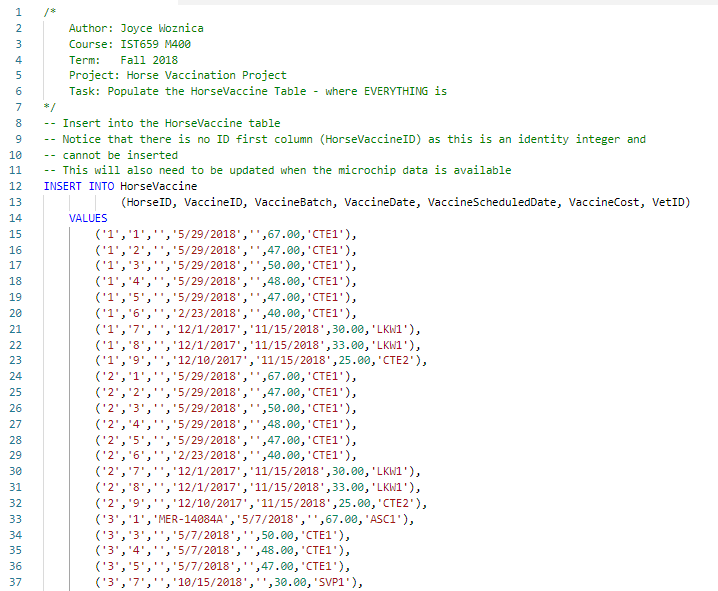


Figure 15: Some Representative Inserts to Load the HorseVaccine Table

## Data Manipulation

I have provided some updates, deletions, alters, etc. to the data based on what I had to do in the short term to provide changes to the table data.

### Adding some Data to the Tables

I decided to provide some additional information to the Clinic and Vet tables as well as the Vaccine table to support vaccines recently done in the last two weeks. In order to do that, I created a couple of general SQL scripts with INSERT statements to accommodate those changes.

#### AddVet.sql

This adds 3 vets and 2 clinics to associated tables.

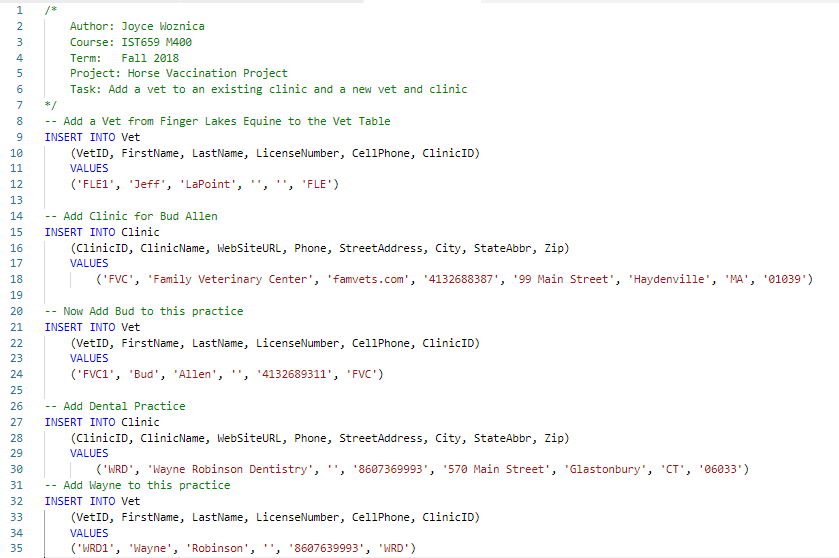


Figure 16: AddVet INSERT statements to Add Data to Vet and Clinic Tables

#### AddDentistandChiro.sql

This function adds entries into the ***Vaccine*** table for the *Chiropractor* on a *Quarterly* basis and the Dentist on a Semi-Annual basis.

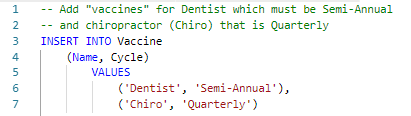


Figure 17: AddDentistandChiro INSERT statements to Add Data to Vaccine Table

#### MakeBarnNameandVaccineNameUnique.sql

This file updates the ***Horse*** table construct and the ***Vaccine*** table construct to make *BarnName* and *Name* UNIQUE constraints, respectively.

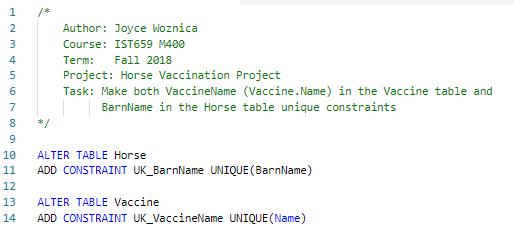


Figure 18: MakeBarnNameandVaccineNameUnique ALTER TABLE statements

### Functions, Procedures, Views, and More

In addition to just update the tables a bit with various information, I also created several functions, procedures, views and other items to manipulate the data and to make certain procedures and functions available for what will become the UI that I plan to create.

#### RenameScheduleDate.sql

This alters a table (using the MSSQL “sp\_rename” function) to change a column name in the ***HorseVaccine*** table.

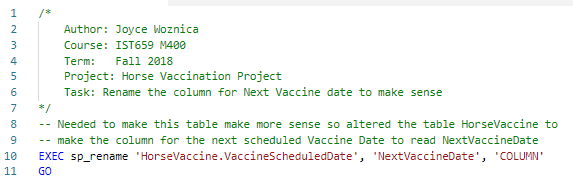


Figure 19: RenameScheduleDate Execution of MSSQL Function to Modify the HorseVaccine Table

#### NumMonthsFunction.sql

This is a function that will determine the number of months to add to the *VaccineDate* to derive the *ScheduledVaccineDate* based on the *Cycle* in the ***Vaccine*** table. It accepts the *VaccineID*. Included is how to execute this function for a specific vaccine.

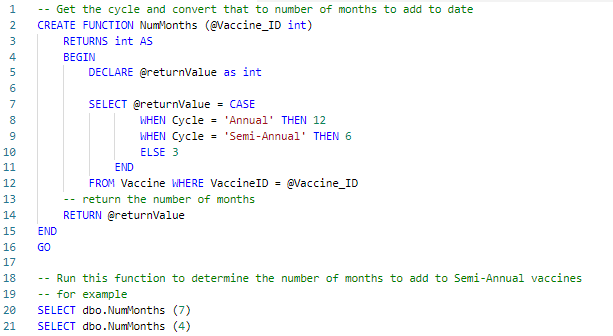


Figure 20: NumMonths Function to Determine NextVaccineDate

#### VaccineSchedules.sql

This code provides two stored procedures that provide updates to the ***Vaccine*** table. One for changing the name of a vaccine and one for modifying the cycle for a vaccine.

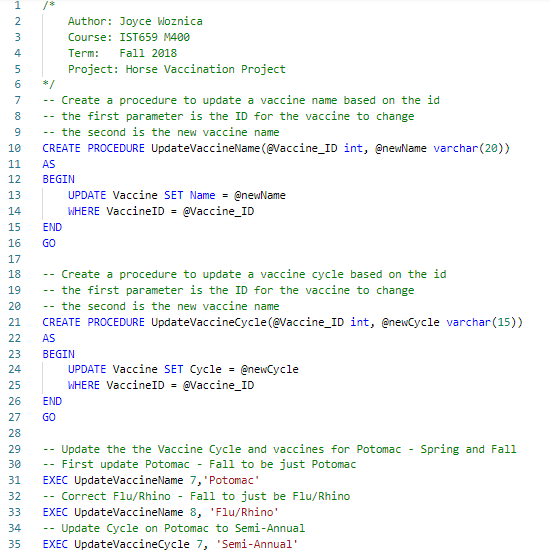


Figure 21: VaccineSchedules Stored Procedures to Allow Updates of the Vaccine Table

#### RemoveExtraVaccine.sql

This code is provided in this section because of the order I needed to do things based on my original design (which I elected not to change as I learned so much by taking that approach). This renames all the ‘Potomac – Spring’ (VaccineID #2) to 7 (Just ‘Potomac’) in the ***HorseVaccine*** table, then I was able to delete the extra ‘Potomac’ entry and use the code shown in this document to make ‘Potomac’ a Semi-Annual vaccine instead of having two similar entries once per year.

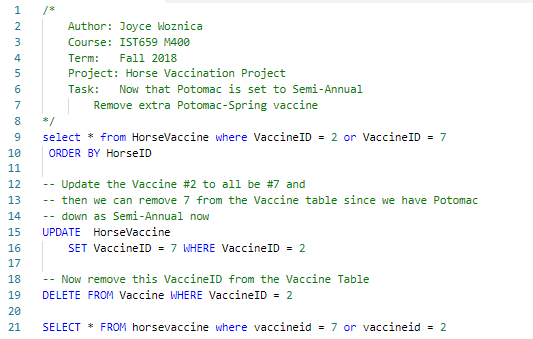


Figure 22: RemoveExtraVaccine UPDATE, DELETE to Update Vaccine, HorseVaccine Tables

#### UpdateNextVaccineDate.sql

Updates the *NextVaccineDate* on all rows to be the proper date based on the *Cycle* for the specific *Vaccine* in that row.   
This was one of the most difficult for me to write because of the need to make a massive change to the entire table (which will not be required again because of the *AddHorseVaccine* procedure that appears in the section Answering Data Questions page 18.   
The WHILE loop provides an update looping through all the vaccines in the ***Vaccine*** table.

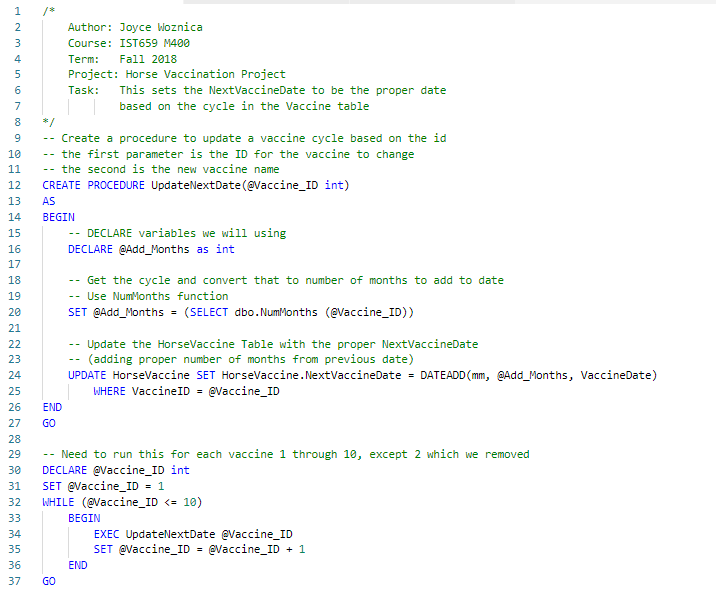


Figure 23: UpdateNextVaccineDate Procedure with loop to Update tables

#### UpdateMicrochip.sql

This file updates based on the current microchip number, the new one reported by the vet when inserted.

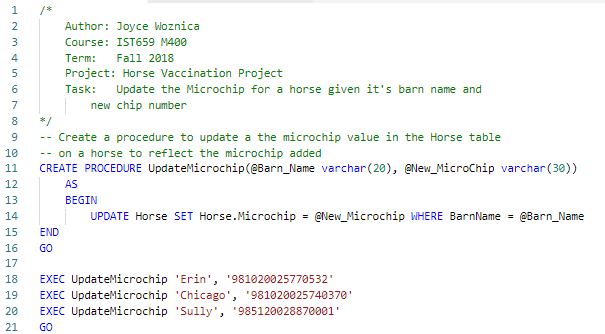


Figure 24: UpdateMicrochip Stored Procedure to add Microchip data to the Horse Table

## Answering Data Questions

There are several questions that I would like for this data to answer. Here is some of the queries that I put together to answer some of these questions. Most of my question provide single line answers as that was my goal. These are not reports, but in Implementation section starting on page 23, I have provided some reports on the data as well.

### When is a Horse Due for a Specific Vaccine (*WhenDue.sql*)

This function uses others created (like NumMonths for deriving the schedule date for a vaccine) to provide the answer to the question “When is <HORSE> due for <VACCINE>?” The code below also shows how the function is called. As shown in my Normalized Model on page 6, the *BarnName* (in the ***Horse*** table) and *Name* (in the ***Vaccine*** table) were made unique with ALTER TABLE commands provided in MakeBarnNameandVaccineNameUnique.sql section that is shown on page 15.

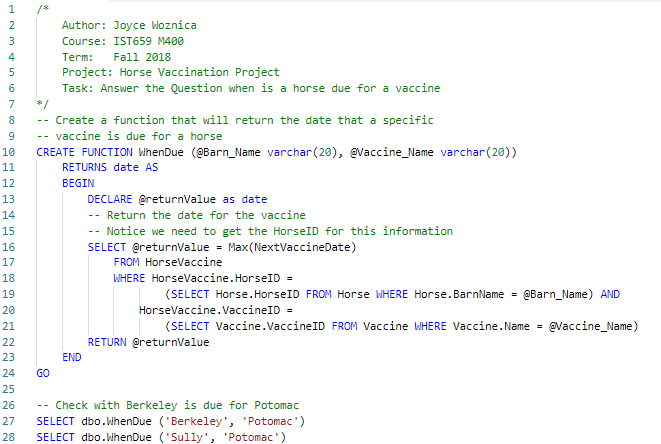


Figure 25: WhenDue Function returns when a horse is due for the noted Vaccine

### Which Vet Performed the Last Vaccination of Vaccine (*WhichVet.sql*)

This function uses others created (like *WhenWasVaccine* for deriving the last date for a vaccine) to provide the answer to the question “What vet did <HORSE>’s last <VACCINE>?” This includes a view that joins the ***Vet*** and ***Clinic*** tables to provide a list of the VetID, Vet First Name, Vet Last Name and Clinic where he/she works as a response to this question.

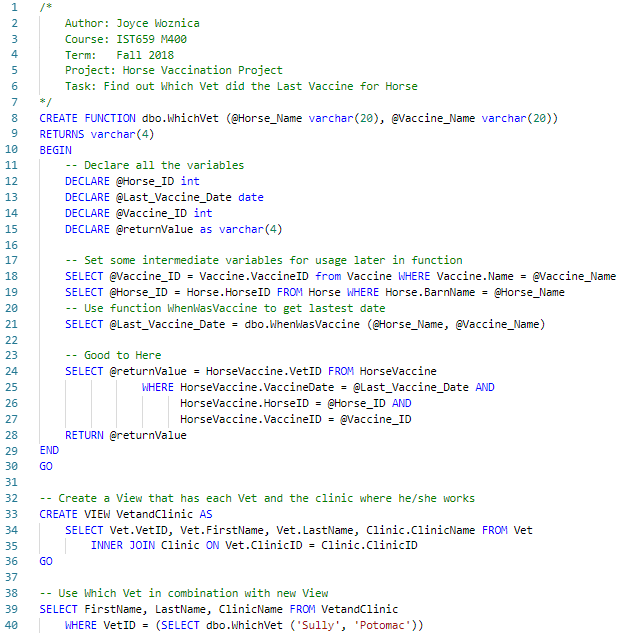


Figure 26: WhichVet Function returns which Vet did a Horse’s last Vaccine

### When was a Vaccine Performed (*WhenWasVaccine.sql*)

This function answers the question “When did <HORSE> get the last dose of <VACCINE>?” given a specific horse name and vaccine name.

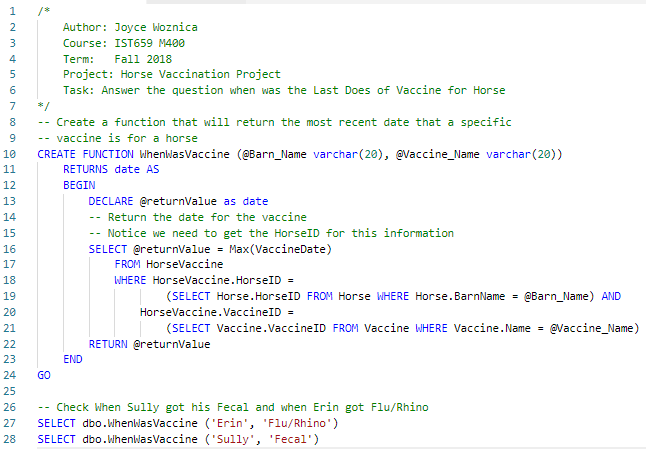


Figure 27: WhenWasVaccine Function returns when a horse is got the last dose of Vaccine

### How much money was spent on vaccines in a given year (*HowMuchForYear.sql*)

This function answers the question “How much money was spent in <Year> on vaccines?” given a specific year.

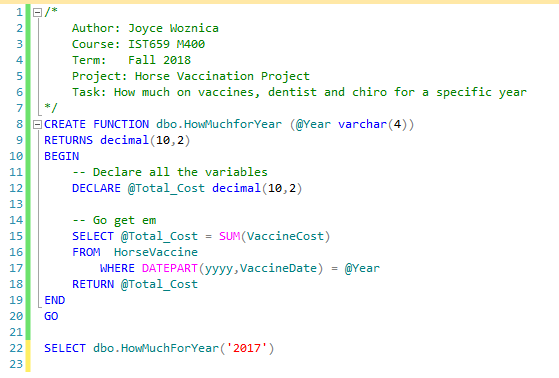


Figure 28: HowMuchForYear Function returns a currency value

### Average Price for a Specific Vaccine (*AveragePrice.sql*)

This function answers the question “What is the Average Price for <Vaccine>?”

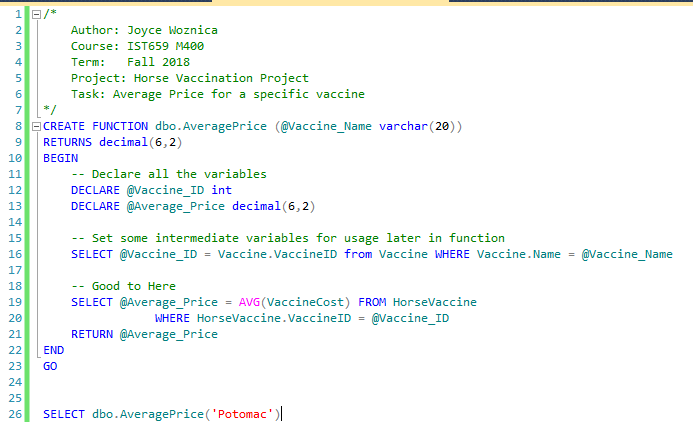


Figure 29: AveragePrice Function returns a currency value

### How much money was spent on vaccines in a given year for a horse (*TotalPerHorsePerYear.sql*)

This function answers the question “How much money was spent in <Year> on vaccines for <Horse>?”

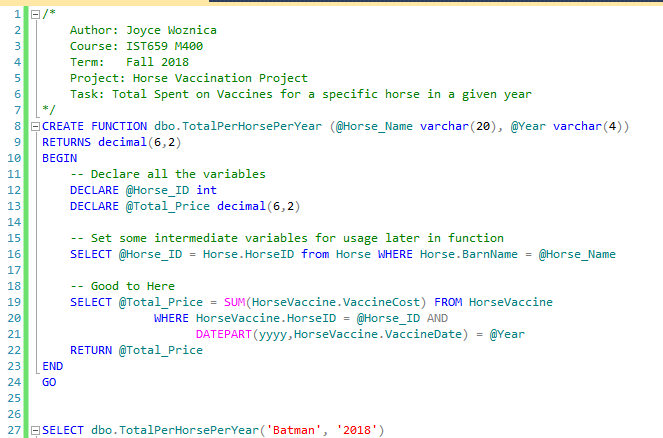


Figure 30: TotalPerHorsePerYear Function returns a currency value

### General Maintenance

#### AddHorseVaccine.sql

This file updates the **HorseVaccine** table with a recent vaccine passing it the required parameters to update the row. It includes running the *NumMonths* function to calculate the *NextVaccineDate* using the proper *Cycle*. This is key to the application as we will need to update the database with this function every time a horse gets his/her next vaccine.

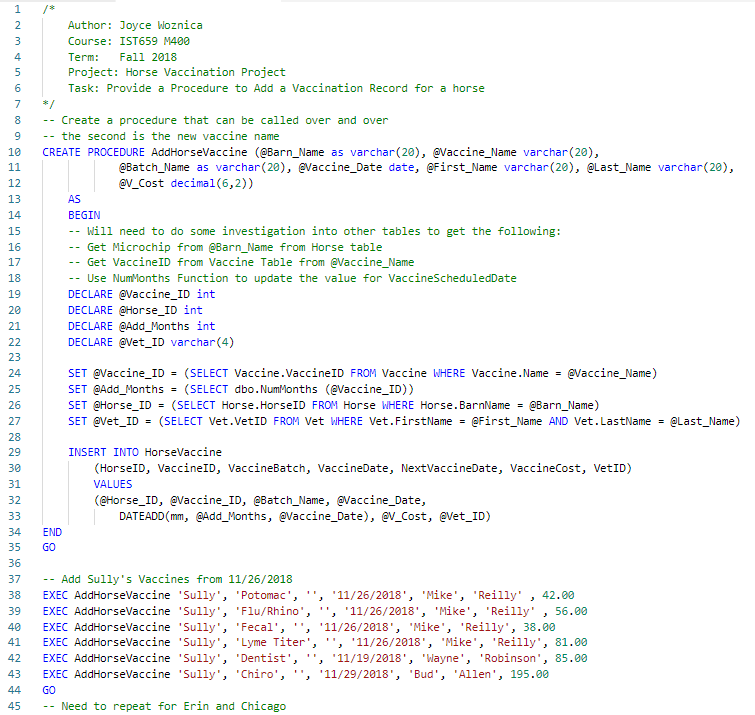


Figure 31: AddHorseVaccine Stored Procedure that Updates the HorseVaccine Table

## Implementation

I used two approaches for implementing a user interface to my database. I wanted something easy to use for the end user (including myself) to allow for accessing the data and obtaining the information desired.

### Phase 1 – General Screens using eForm Tool

Originally, I started out attempting to use an eForm product to which I am family as it allows database lookups and connection to Microsoft SQL Server through JDBC connections which I have used before; however, this proved to be quite cumbersome. I was able to at least draw what the forms I wanted to create would look like and display the data as I felt it might appear. The following screenshots show some of these ideas.

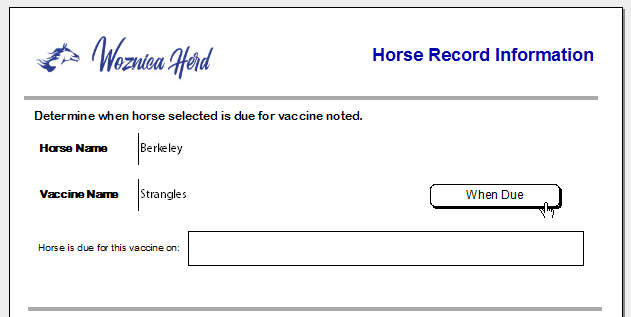


Figure 32: Determine when Horse is Due for Vaccine

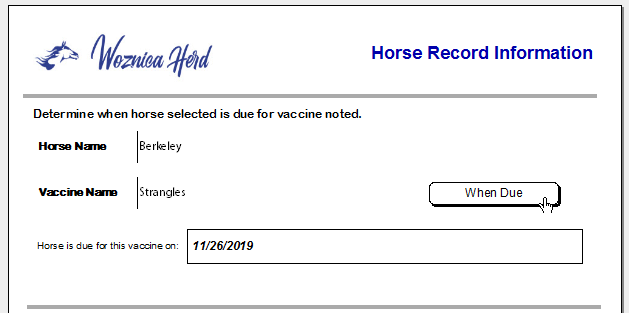


Figure 33: After Clicking “When Due” Response displayed

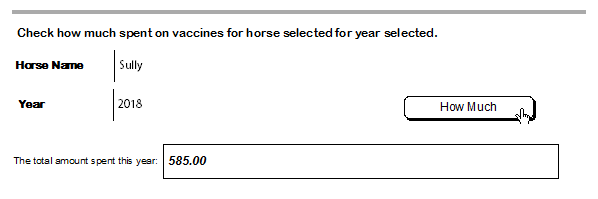


Figure 34: Determining How Much was Spent on Vaccines for a Particular Horse in a Year

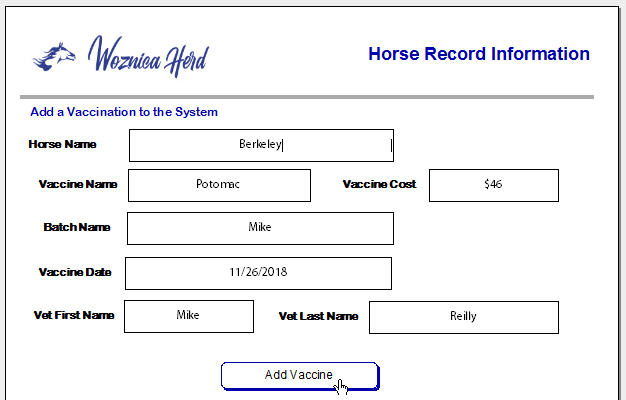


Figure 35: Form to Add a Vaccine to the Database (Date and Cost are defined as date and float)

### Phase 2 – Tableau as a User Design Tool

My second approach was using Tableau to design some ways to display the data in informative ways for the end user. I have not used this tool before, so there was a learning curve associated with this approach.

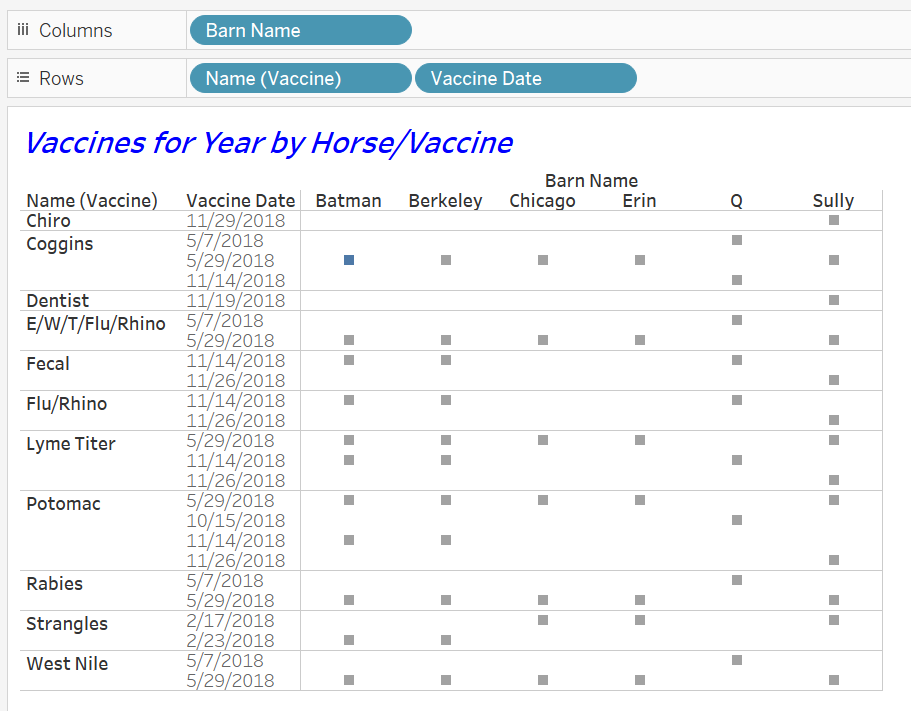


Figure 36: Worksheet to Display Vaccines by Horse by Vaccine Date

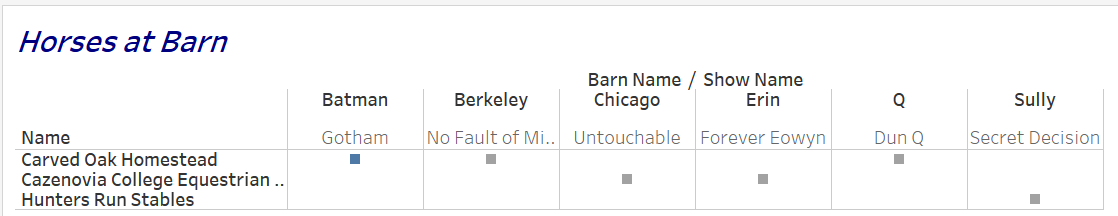


Figure 37: Worksheet to Display Horse and their Barn Location by Barn Name

After working with Tableau, I was able to take a few worksheets that I created and put them all together in a dashboard for the end user. The dashboard presented below is a combination of the previously displayed worksheets and an additional one for costs for vaccines by year by horse. This provides the end user a nice view into the information in the database.

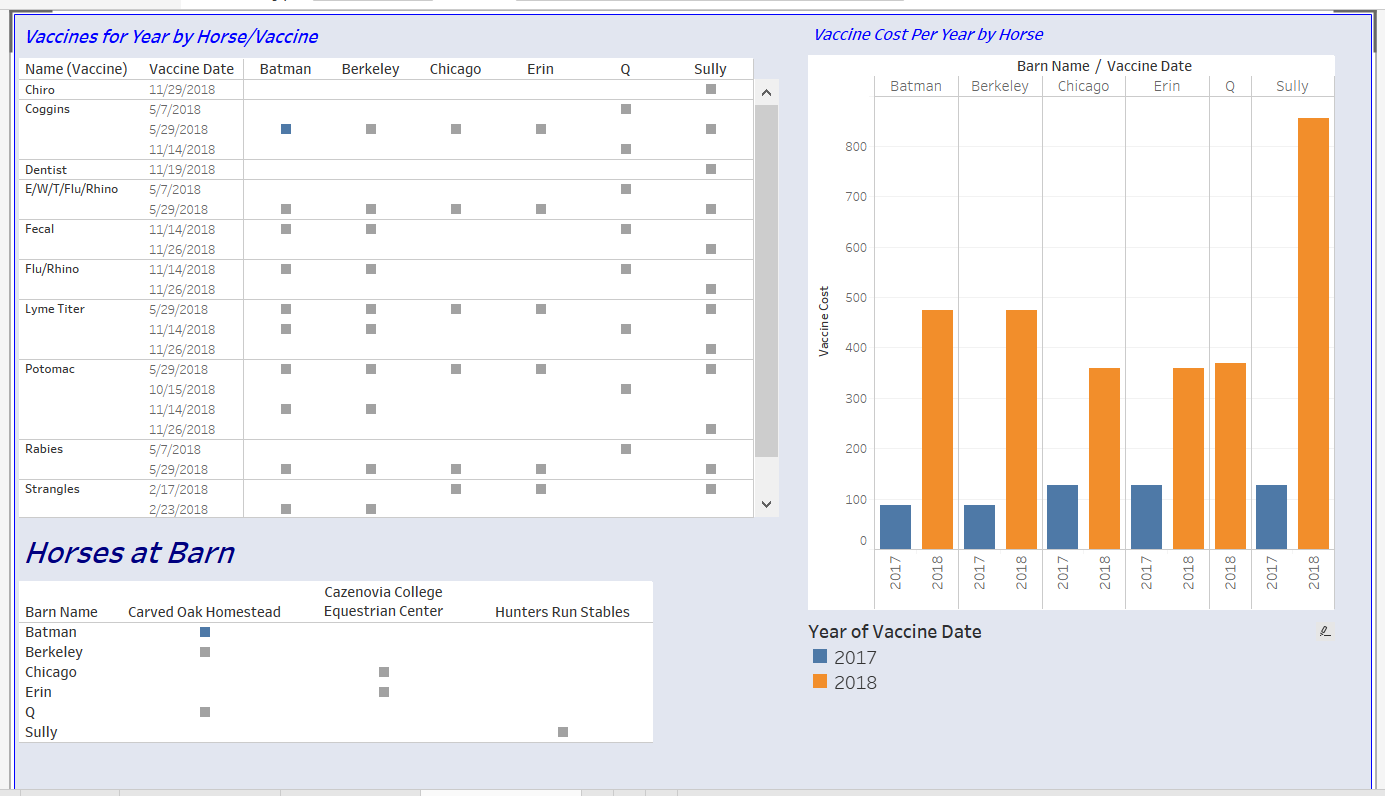


Figure 38: Dashboard of Various Worksheets

In the following two dashboards, I made more appropriate dashboards around my data questions. The first display more about the Vets, Clinics, Barns and Horses linking data by state to provide further insight – like which vet I can use for a particular horse based on the barn location.



Figure 39: Vet, Clinic, Barn and Horse Dashboard

The next dashboard provides a little more information about average prices per vaccines, costs per horse per year and other vaccine related data that would be of interest.

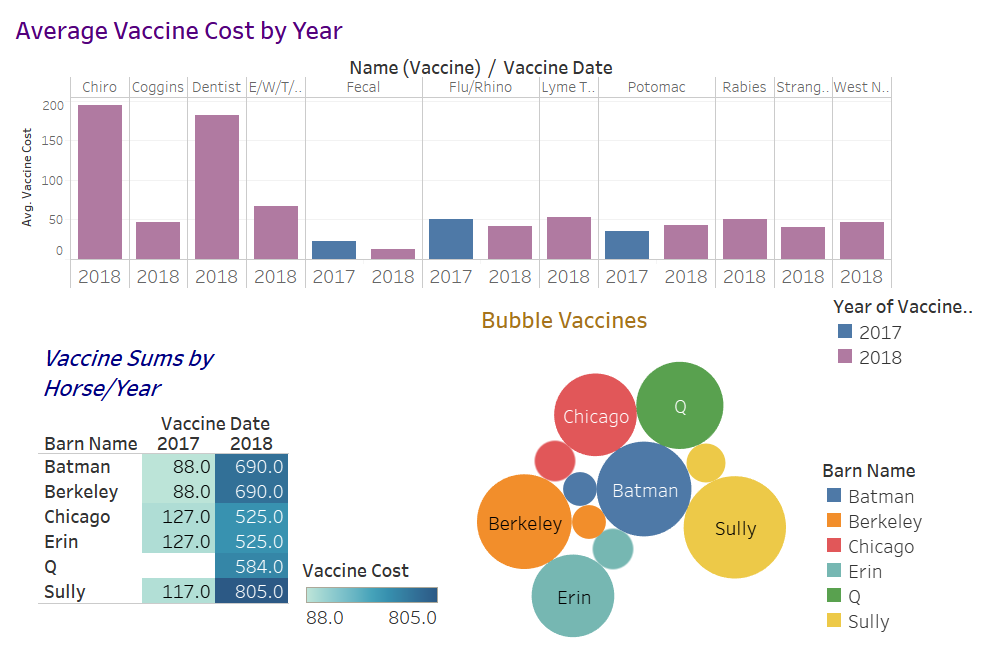


Figure 40: Vaccine Dashboard

# Reflection

I found the overall project very informative and it also provided me with additional insight into this data. I am very familiar with this data and how I need to use the information and found that I had a bit of “scope creep” wanting to add more to the data as it is a real-world problem for me.

At the last minute, I decided to add in a few “procedures,” but to lump them into the *Vaccine* and *HorseVaccine* tables. That works fine, but I think I would have used different table and column names so that my model so that I could delineate the information more clearly. For example, dentistry and chiropractic work skew the cost information as they are high ticket items and not traditional vaccines. not listed as a specific vaccine as it takes away from the information.

In working with data like this, I found that knowing what questions you need to ask help you define how the model must look. It is like working backwards. Understanding how the information and data needs to be used also helps to shape the models in a way (like I mentioned the chiropractor and dentist being lumped into vaccines, which is not completely appropriate – which would have changed my approach in the long run had I thought to include this data up front).