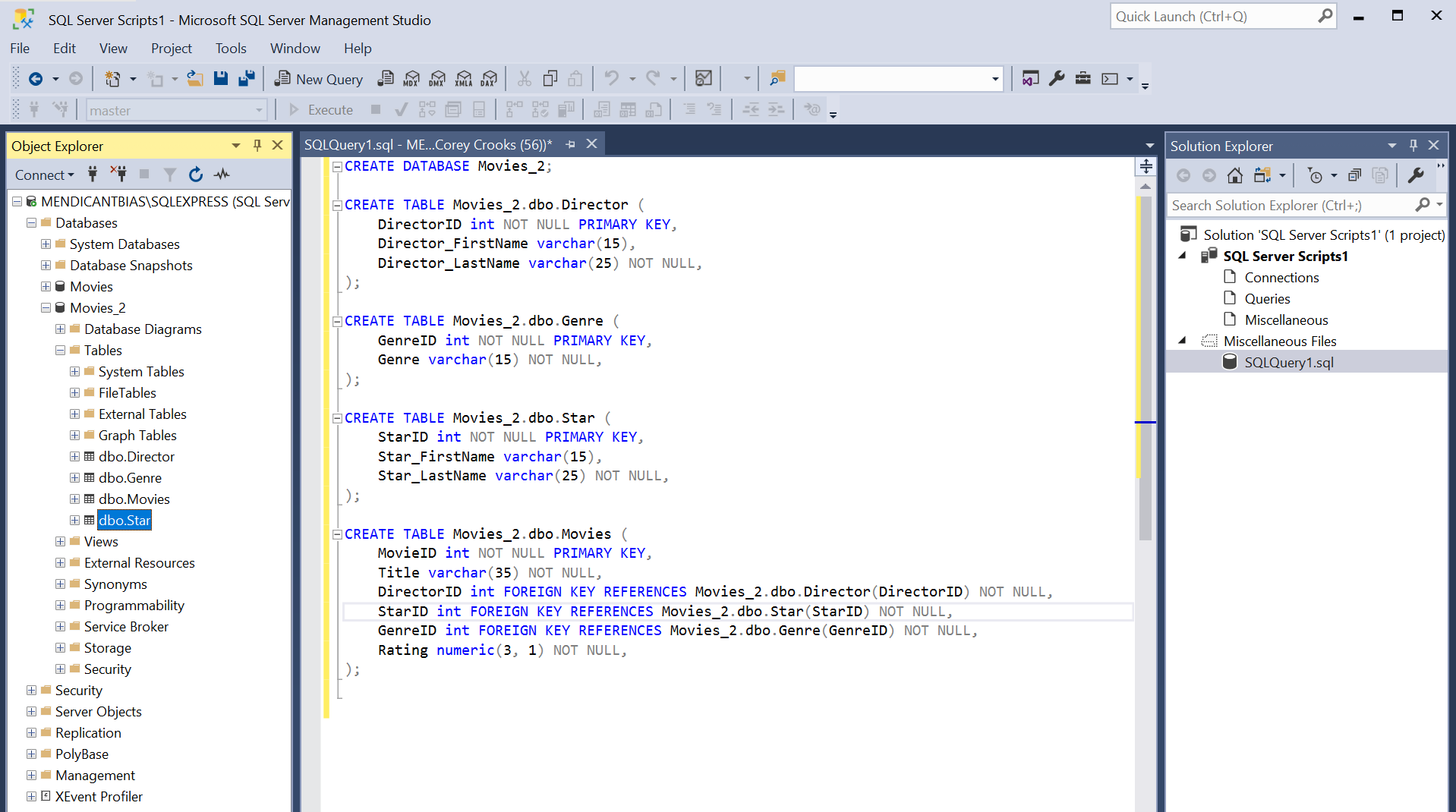
**Unit 3 Movie Database Design with SQL**

Corey Crooks

Purdue University Global

IT234 – Leon King

May 2, 2021



Above is a screenshot of my SQL statements written to create a database identical to that of the Movies database. I will say, I was unable to run my script with the Producer table not integrated, and since the assignment Rubric only calls for four tables (Director, Genre, Movies, and Star) I opted to remove the problem statement. This could be easily rectified by utilizing the same CREATE TABLE statements found to create the other tables should it be necessary in the future.

Starting off, I use a CREATE DATABASE statement to create the Movies\_2 database object. Below that, I use a CREATE TABLE statement to create the Director table within the new Movies\_2 database using a breadcrumb solution utilizing the [database].dbo.[table] syntax. I specifiy that the attributes of this table shall be DirectorID (an integer), Director\_FirstName (an up to 15 letter string), and Director\_LastName (an up to 25 letter string). The Primary key of this table is DirectorID. Only the first name is allowed to be left blank.

Moving on, I create the Genre table utilizing the same structure found above. This includes the attributes GenreID (an integer as well as the primary key), and Genre (an up to 15 letter string). No value in this table is allowed to be left blank.

After that, I create the Star table with the attributes StarID (an integer), Star\_FirstName (an up to 15 letter string), and Star\_LastName (and up to 25 letter string). StarID is the primary key of this table. Only Star\_FirstName is allowed to be blank.

Finally, we have the bread and butter of this database. The Movies table is created to tie everything together. This includes the attributes MovieID (an integer and the primary key), Title (an up to 35 letter string), DirectorID (an integer and a foreign key), StarID (an integer and a foreign key), GenreID (an integer and a foreign key), ProducerID (an integer and a foreign key), and Rating (a decimal with up to three numbers to the left of the decimal and one number to the right). No values in this table are allowed to be left blank.

This method was very intriguing to work with after designing the database with software and buttons. Having this raw utility focused language without all those GUI elements to help and provide context was definitely challenging. But overall, as a Software Programming major, I really enjoyed working with SQL to design a database from scratch. I feel that committing to learning SQL to create your database may really help out in the long run, as this will provide an avenue for you to be a much more flexible developer. After all, database management software that doesn’t have support for SQL of any kind probably isn’t software that you should be using for your business.