# Ryan Timbrook

## IST 659 Data Admin Concepts &Db Mgmt

## Date: 8/13/18

## Lab Assignment: Lab 5, Physical Design and DDL

## Description / Learning Objective

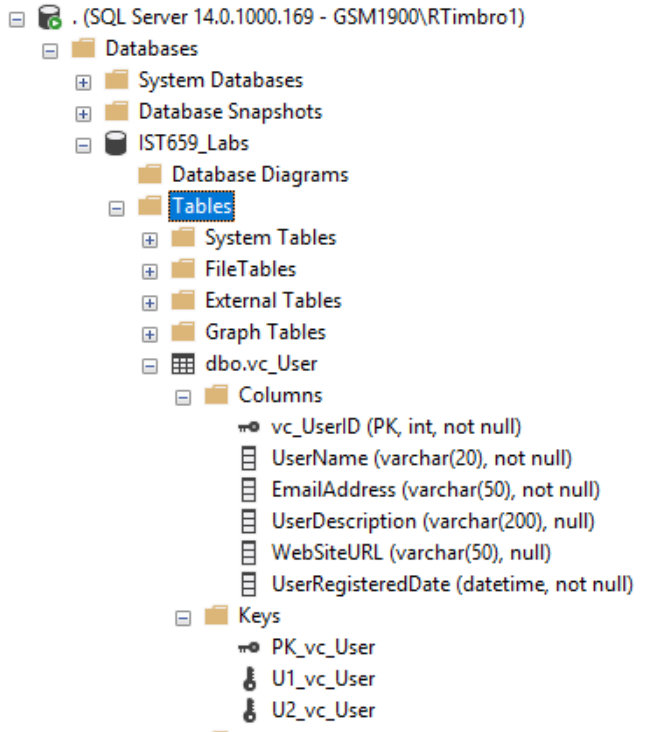
* Demonstrate data definition language (DDL) proficiency
* Demonstrate ability to convert from diagrams to SQL code

## Responses:

### Part 1, Creating Tables

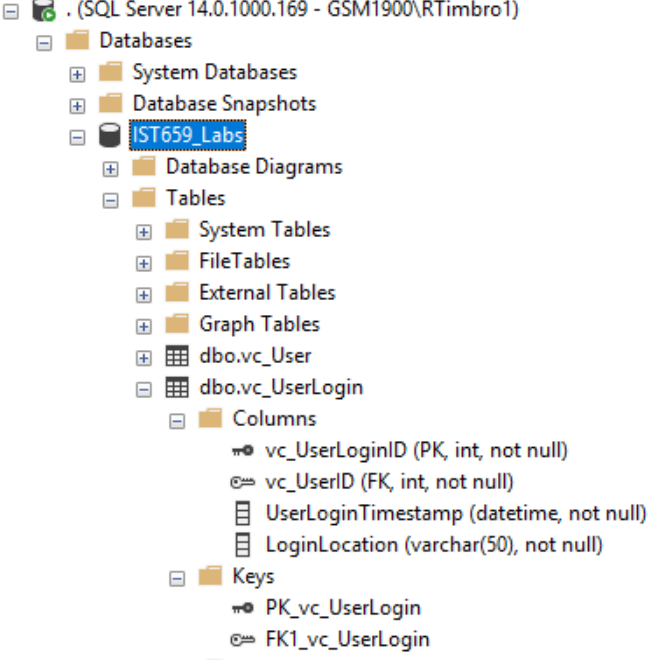
TODO-1: *Take a screenshot of your Object Explorer with your table and its columns and keys expanded. Paste this screenshot into your answer doc labeled as “User Table”*

## User Table



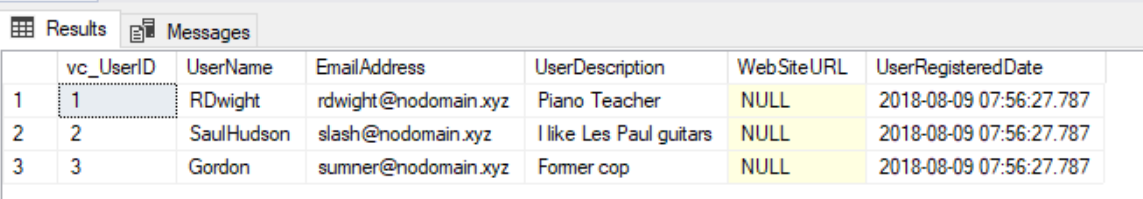
TODO-2: *Take a screenshot of your Object Explorer with your vc\_UserLogin table and its columns and keys expanded. Paste this screenshot into your answer doc labeled as “User Login Table”*

## User Login Table



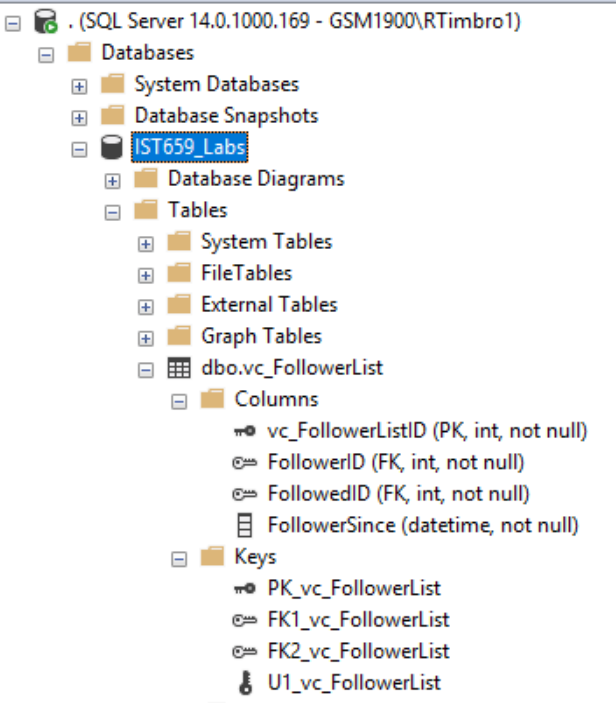
TODO-3: *Take a screenshot of your results grid and paste it into your answer document. Label it “User records”*

## User Records



TODO-4: *In the Object Explorer, refresh your Tables folder and take a screenshot of this portion of the screen. Paste it into your answers document and label it “Follower List”*

## Follower List

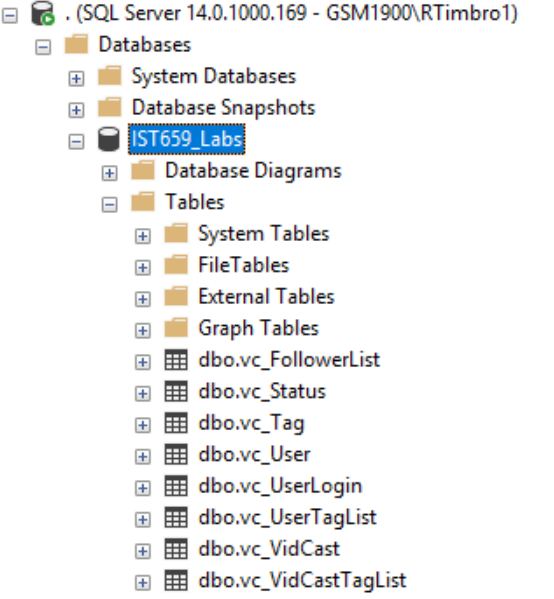


### Part 2, The Rest of the Tables

*Code and execute the SQL CREATE TABLE statements to create the remaining tables.*

TODO-1: *Refresh your Tables folder in the Object Explorer and take a screenshot of the listing as before. Paste this screenshot into your answers document labeled “VidCast tables”*

## VidCast Tables



TODO-2: *After completing Part 2, copy and paste the text of your SQL query file at the end of your answers document. Save this document and submit it to the appropriate section on the LMS.*

*SQL Query*

|  |
| --- |
| /\*  Author: Ryan Timbrook  Course: IST 659 Data Admin Concepts & Db Mgmt  Term: Summer, 2018  \*/  -- Creating the User Table  Create Table vc\_User(  -- Columns for the User Table  vc\_UserID int identity,  UserName varchar(20) not null,  EmailAddress varchar(50) not null,  UserDescription varchar(200),  WebSiteURL varchar(50),  UserRegisteredDate datetime not null default GetDate(),  -- Constraints on the User Table  CONSTRAINT PK\_vc\_User PRIMARY KEY(vc\_UserID),  CONSTRAINT U1\_vc\_User UNIQUE(UserName),  CONSTRAINT U2\_vc\_User UNIQUE(EmailAddress)  )  -- End Creating the User Table  -- Creating the UserLogin Table  Create Table vc\_UserLogin(  -- Columns for the UserLogin table  vc\_UserLoginID int identity,  vc\_UserID int not null,  UserLoginTimestamp datetime not null default GetDate(),  LoginLocation varchar(50) not null,  -- Constraints for the UserLogin table  CONSTRAINT PK\_vc\_UserLogin PRIMARY KEY(vc\_UserLoginID),  CONSTRAINT FK1\_vc\_UserLogin FOREIGN KEY(vc\_UserID) REFERENCES vc\_User(vc\_UserID)  )  -- End Creating the UserLogin Table  -- Adding Data to the User Table  INSERT INTO vc\_User(UserName, EmailAddress, UserDescription)  VALUES  ('RDwight','rdwight@nodomain.xyz','Piano Teacher'),  ('SaulHudson','slash@nodomain.xyz','I like Les Paul guitars'),  ('Gordon','sumner@nodomain.xyz','Former cop')  SELECT \* from vc\_User  -- End adding data to the User Table  -- Creating the follower List Table  CREATE TABLE vc\_FollowerList(  -- Columns for the follower List Table  vc\_FollowerListID int identity,  FollowerID int not null,  FollowedID int not null,  FollowerSince datetime not null,  -- Constraints on the Follower List Table  CONSTRAINT PK\_vc\_FollowerList PRIMARY KEY(vc\_FollowerListID),  CONSTRAINT U1\_vc\_FollowerList UNIQUE(FollowerID, FollowedID),  CONSTRAINT FK1\_vc\_FollowerList FOREIGN KEY(FollowerID) REFERENCES vc\_User(vc\_UserID),  CONSTRAINT FK2\_vc\_FollowerList FOREIGN KEY(FollowedID) REFERENCES vc\_User(vc\_UserID)  )  -- End Creating the follower List Table  -- Creating the following Tag Table  -- Order: 1  CREATE TABLE vc\_Tag(  -- Columns for the follower List Table  vc\_TagID int identity,  TagText varchar(20),  TagDescription varchar(100),  -- Constraints on the Follower List Table  CONSTRAINT PK\_vc\_TagID PRIMARY KEY(vc\_TagID),  CONSTRAINT U1\_TagText UNIQUE(TagText)  )  -- End Creating the Tag Table  -- Creating the following Status Table  -- Order: 2  CREATE TABLE vc\_Status(  -- Columns for the follower List Table  vc\_StatusID int identity,  StatusText varchar(20),  -- Constraints on the Follower List Table  CONSTRAINT PK\_vc\_StatusID PRIMARY KEY(vc\_StatusID),  CONSTRAINT U1\_StatusText UNIQUE(StatusText)  )  -- End Creating the Status Table  -- Creating the following VidCast Table  -- Order: 3  CREATE TABLE vc\_VidCast(  -- Columns for the follower List Table  vc\_VidCastID int identity,  VidCastTitle varchar(50),  StartDateTime datetime,  EndDateTime datetime,  ScheduledDurationMinutes int,  RecordingURL varchar(50),  vc\_UserID int,  vc\_StatusID int,  -- Constraints on the Follower List Table  CONSTRAINT PK\_vc\_VidCastID PRIMARY KEY(vc\_VidCastID),  CONSTRAINT FK1\_vc\_UserID FOREIGN KEY(vc\_UserID) REFERENCES vc\_User(vc\_UserID),  CONSTRAINT FK2\_vc\_StatusID FOREIGN KEY(vc\_StatusID) REFERENCES vc\_Status(vc\_StatusID)  )  -- End Creating the VidCast Table  -- Creating the following VidCastTagList  -- Order: 4  CREATE TABLE vc\_VidCastTagList(  -- Columns for the follower List Table  vc\_VidCastTagList int identity,  vc\_TagID int,  vc\_VidCastID int  -- Constraints on the Follower List Table  CONSTRAINT PK\_vc\_VidCastTagList PRIMARY KEY(vc\_VidCastTagList),  CONSTRAINT FK1\_vc\_TagID FOREIGN KEY(vc\_TagID) REFERENCES vc\_Tag(vc\_TagID),  CONSTRAINT U1\_vc\_TagID UNIQUE(vc\_TagID), --ARE THESE UNIQUE DECLARATIONS NEEDED? FKs should enforce this?  CONSTRAINT FK2\_vc\_VidCastID FOREIGN KEY(vc\_VidCastID) REFERENCES vc\_VidCast(vc\_VidCastID),  CONSTRAINT U2\_vc\_VidCastID UNIQUE(vc\_VidCastID) --ARE THESE UNIQUE DECLARATIONS NEEDED? FKs should enforce this?  )  -- End Creating the VidCastTagList  -- Creating the following UserTagList Table  -- Order: 5  CREATE TABLE vc\_UserTagList(  -- Columns for the follower List Table  vc\_UserTagListID int identity,  vc\_TagID int,  vc\_UserID int,  -- Constraints on the Follower List Table  CONSTRAINT PK\_vc\_UserTagListID PRIMARY KEY(vc\_UserTagListID),  CONSTRAINT FK1\_vc\_UserTagList\_TagID FOREIGN KEY(vc\_TagID) REFERENCES vc\_Tag(vc\_TagID),  CONSTRAINT U1\_vc\_UserTagList\_TagID UNIQUE(vc\_TagID),  CONSTRAINT FK2\_vc\_UserTagList\_UserID FOREIGN KEY(vc\_UserID) REFERENCES vc\_User(vc\_UserID),  CONSTRAINT U2\_vc\_UserTagList\_UserID UNIQUE(vc\_UserID)  )  -- End Creating the UserTagList Table |