# Ryan Timbrook

## IST 659 Data Admin Concepts &Db Mgmt

## Date: 8/20/18

## Lab Assignment: Lab 6, Querying, Inserting, Updating and Deleting

## Description / Learning Objective

* Demonstrate data manipulation language (DML) proficiency

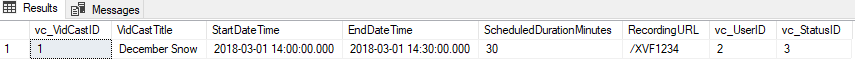
## Responses:

### Part 1, Filling our Tables with Data

*We will have to use each of the CRUD commands to manipulate the data in our tables.*

TODO-1: *Take a screenshot of your results grid and paste it into your answers document labeled ‘First VidCast’*

## First VidCast



TODO-2: *Code and execute the preceding SQL in SSMS. Remember you can highlight lines of code in SSMS and when you click “Execute” it will only execute the highlighted code. Paste a screenshot of your results into your answer document labeled as ‘Saul’s First VidCast’*

## Saul’s First VidCast



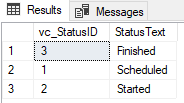
TODO-3: *Code and execute the preceding SQL in SSMS. Remember you can highlight lines of code in SSMS and when you click “Execute” it will only execute the highlighted code. Paste a screenshot of your results into your answer document labeled as ‘Update a User’*

## Update a User



TODO-4: *Code and execute the preceding SQL in SSMS. Remember you can highlight lines of code in SSMS and when you click “Execute” it will only execute the highlighted code. Paste a screenshot of your results into your answer document labeled as ‘No more on time’*

## No more on time

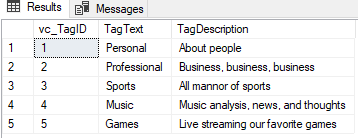


### Part 2, Putting All Together

*In this part, you’ll add some more data to the VidCast tables and write some queries to read the new data.*

TODO-1: *Code and execute the SQL INSERT statement(s) to add the preceding values to the vc\_Tag table. When finished, write a SELECT statement that retrieves all rows from vc\_Tag. Paste a copy of your code and a screenshot of the results to your answer doc labeled ‘Tags’*

## Tags

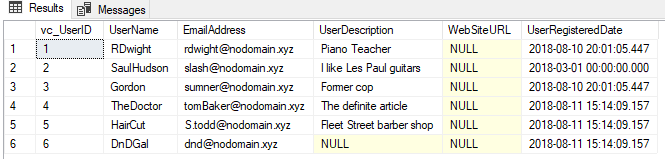


## SQL Query

|  |
| --- |
| -- Adding Tags to the vc\_Tag Table  INSERT INTO vc\_Tag(TagText, TagDescription)  VALUES('Personal','About people'),('Professional','Business, business, business'),('Sports','All mannor of sports'),('Music','Music analysis, news, and thoughts'),('Games','Live streaming our favorite games')  -- Retrieve all rows from vc\_Tag Table  SELECT \* from vc\_Tag |

TODO-2: *Code and execute the SQL INSERT statement(s) to add the preceding values to the vc\_User table. When finished, write a SELECT statement that retrieves all rows from vc\_User. Paste a copy of your code and a screenshot of the results to your answer doc labeled ‘Users’*

## Users

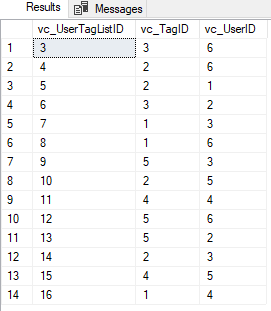


## SQL Query

|  |
| --- |
| -- Adding users to the vc\_User table  INSERT INTO vc\_User(UserName,EmailAddress,UserDescription)  VALUES('TheDoctor','tomBaker@nodomain.xyz','The definite article'),('HairCut','S.todd@nodomain.xyz','Fleet Street barber shop'),('DnDGal','dnd@nodomain.xyz',NULL)  -- Retrieve all rows from vc\_User table  SELECT \* from vc\_User |

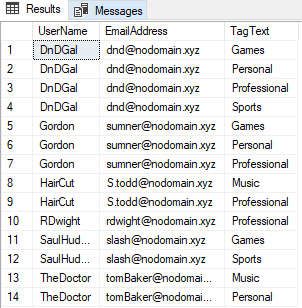
TODO-3: *Copy and paste the complete code from Appendix B into your SQL script file and execute it against your database. After you have inserted those 14 rows, Code and execute a SELECT statement to retrieve all vc\_UserTagList records and paste a screenshot of your results to your answer doc labeled ‘User Tag List’.*

## User Tag List



TODO-4: *Code and execute a SQL SELECT statement that retrieves the vc\_User’s UserName and EmailAddress and the vc\_Tag TagText for all vc\_User records, ordered by user name then tag. Your output should look like the following screenshot. Copy and paste your SQL and a results screenshot to your answers doc labeled ‘User Tags Report’*

## User Tags Report



## SQL Query

|  |
| --- |
| -- Retrieve UserName, EmailAddress, TagText for all User records ordered by UserName, then Tag  SELECT  vc\_User.UserName,  vc\_User.EmailAddress,  vc\_Tag.TagText  FROM vc\_UserTagList  JOIN vc\_User ON vc\_UserTagList.vc\_UserID = vc\_User.vc\_UserID  JOIN vc\_Tag ON vc\_UserTagList.vc\_TagID = vc\_Tag.vc\_TagID  ORDER BY vc\_User.UserName, vc\_Tag.TagText |

## SQL Query File

|  |
| --- |
| /\*  Author: Ryan Timbrook  Course: IST 659 Data Admin Concepts & Db Mgmt  Term: Summer, 2018  Lab: 6, Querying Inserting Updating and Deleting  \*/  -- Adding a row to the vc\_Status table  INSERT INTO vc\_Status(StatusText)  VALUES('Scheduled')  -- The following line shows all of the rows in vc\_Status  SELECT \* FROM vc\_Status  -- Adding three more rows to the vc\_Status table  INSERT INTO vc\_Status(StatusText)  VALUES('Started'),('Finished'),('On Time')  -- The following line shows all of the rows in vc\_Status  SELECT \* FROM vc\_Status  -- Adding a vidcast record to the VidCast Table  SELECT \* FROM vc\_User WHERE UserName = 'SaulHudson'  SELECT \* FROM vc\_Status WHERE StatusText = 'Finished'  INSERT INTO vc\_VidCast(VidCastTitle, StartDateTime, EndDateTime, ScheduledDurationMinutes, RecordingURL, vc\_UserID, vc\_StatusID)  VALUES ('December Snow', '3/1/2018 14:00', '3/1/2018 14:30', 30, '/XVF1234', 2, 3)  -- Read all rows from vc\_VidCast  SELECT \* FROM vc\_VidCast  -- Saul's First VidCast  SELECT  vc\_User.UserName,  vc\_User.EmailAddress,  vc\_VidCast.VidCastTitle,  vc\_VidCast.StartDateTime,  vc\_VidCast.EndDateTime,  vc\_VidCast.ScheduledDurationMinutes / 60.0 as ScheduledHours,  vc\_Status.StatusText  FROM vc\_VidCast  JOIN vc\_User ON vc\_VidCast.vc\_UserID = vc\_User.vc\_UserID  JOIN vc\_Status ON vc\_VidCast.vc\_StatusID = vc\_Status.vc\_StatusID  WHERE vc\_User.UserName = 'SaulHudson'  ORDER BY vc\_VidCast.StartDateTime  -- End Saul's First  -- Correcting a User's UserRegisteredDate  UPDATE vc\_User SET UserRegisteredDate = '3/1/2018' WHERE UserName = 'SaulHudson'  SELECT \* FROM vc\_User Where UserName = 'SaulHudson'  -- End Update  -- Deleting a record from the Status Table  -- See what rows we have in status  SELECT \* FROM vc\_Status  -- Delete the On time status  DELETE vc\_Status WHERE StatusText = 'On time'  -- See the effect  SELECT \* FROM vc\_Status  -- End Deleting a record from the Status Table  /\*  Part 2 - Putting All Together  In this part, you'll add some more data to the VidCast tables and write some queries to read the new data  \*/  -- Adding Tags to the vc\_Tag Table  INSERT INTO vc\_Tag(TagText, TagDescription)  VALUES('Personal','About people'),('Professional','Business, business, business'),('Sports','All mannor of sports'),('Music','Music analysis, news, and thoughts'),('Games','Live streaming our favorite games')  -- Retrieve all rows from vc\_Tag Table  SELECT \* from vc\_Tag  -- Adding users to the vc\_User table  INSERT INTO vc\_User(UserName,EmailAddress,UserDescription)  VALUES('TheDoctor','tomBaker@nodomain.xyz','The definite article'),('HairCut','S.todd@nodomain.xyz','Fleet Street barber shop'),('DnDGal','dnd@nodomain.xyz',NULL)  -- Retrieve all rows from vc\_User table  SELECT \* from vc\_User  -- Inserting 14 rows into the vc\_UserTagList  -- Run Select Statements to view UserID values before executing INSERT  SELECT vc\_UserID FROM vc\_User WHERE UserName = 'DnDGal'  SELECT vc\_UserID FROM vc\_User WHERE UserName = 'RDwight'  SELECT vc\_UserID FROM vc\_User WHERE UserName = 'SaulHudson'  SELECT vc\_UserID FROM vc\_User WHERE UserName = 'Gordon'  SELECT vc\_UserID FROM vc\_User WHERE UserName = 'HairCut'  SELECT vc\_UserID FROM vc\_User WHERE UserName = 'TheDoctor'  -- Run Select Statements to view TagID values before executing INSERT  SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Sports'  SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Professional'  SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Personal'  SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Games'  SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Music'  INSERT INTO vc\_UserTagList(vc\_UserID,vc\_TagID)  VALUES  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'DnDGal'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Sports')), --DnDGal/Sports  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'DnDGal'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Professional')), --DnDGal/Professional  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'RDwight'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Professional')), --RDwight/Professional  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'SaulHudson'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Sports')), --SaulHudson/Sports  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'Gordon'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Personal')), --Gordon/Personal  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'DnDGal'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Personal')), --DnDGal/Personal  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'Gordon'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Games')), --Gordon/Games  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'HairCut'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Professional')), --HairCut/Professional  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'TheDoctor'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Music')), --TheDoctor/Music  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'DnDGal'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Games')), --DnDGal/Games  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'SaulHudson'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Games')), --SaulHudson/Games  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'Gordon'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Professional')), --Gordon/Professional  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'HairCut'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Music')), --HairCut/Music  ((SELECT vc\_UserID FROM vc\_User WHERE UserName = 'TheDoctor'),(SELECT vc\_TagID FROM vc\_Tag WHERE TagText = 'Personal')) --TheDoctor/Personal    -- Retrieve all rows from vc\_UserTagList table  SELECT \* FROM vc\_UserTagList  -- Retrieve UserName, EmailAddress, TagText for all User records ordered by UserName, then Tag  SELECT  vc\_User.UserName,  vc\_User.EmailAddress,  vc\_Tag.TagText  FROM vc\_UserTagList  JOIN vc\_User ON vc\_UserTagList.vc\_UserID = vc\_User.vc\_UserID  JOIN vc\_Tag ON vc\_UserTagList.vc\_TagID = vc\_Tag.vc\_TagID  ORDER BY vc\_User.UserName, vc\_Tag.TagText |