## Name: Laura C. Larregui Date: February 19, 2020

## Lab 07 – Advanced Query

### Part 1 – Exploratory Data Analysis

1. First Exploratory Analysis

SELECT

vc\_User.UserName

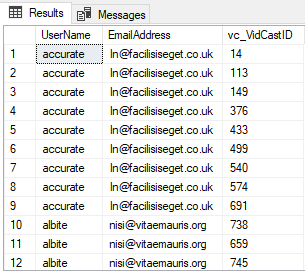
, vc\_User.EmailAddress

, vc\_VidCast.vc\_VidCastID

FROM vc\_VidCast

JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

ORDER BY vc\_User.UserName



1. Basic Summaries (Looking at the users without VidCasts

SELECT \* FROM vc\_User

WHERE vc\_UserID NOT IN (SELECT vc\_UserID FROM vc\_VidCast)

--INCLUDES ALL USERS

SELECT

vc\_User.UserName

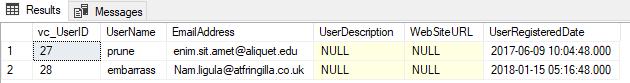
, vc\_User.EmailAddress

, vc\_VidCast.vc\_VidCastID

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

ORDER BY vc\_User.UserName



1. RIGHT JOIN

SELECT

vc\_User.UserName

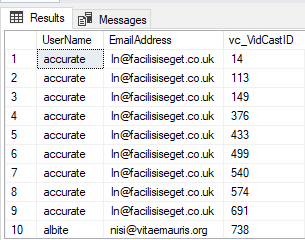
, vc\_User.EmailAddress

, vc\_VidCast.vc\_VidCastID

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

ORDER BY vc\_User.UserName



1. AGGREGATE FUNCTIONS

SELECT

COUNT(vc\_VidCastID) as NumberOfVidCasts

, SUM(ScheduleDurationMinutes) as TotalScheduledMinutes

, MIN(ScheduleDurationMinutes) as MinScheduledMinutes

, AVG(ScheduleDurationMinutes) as AvgScheduledMinutes

, MAX(ScheduleDurationMinutes) as MaxScheduledMinutes

FROM vc\_VidCast



1. Error Message

SELECT

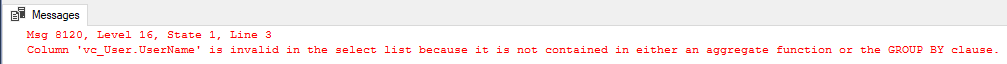
vc\_User.UserName

, vc\_User.EmailAddress

, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

ORDER BY vc\_User.UserName

1. Group BY Part 1

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

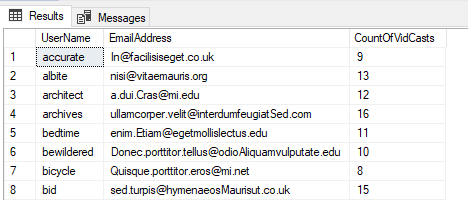
, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

ORDER BY vc\_User.UserName



1. GROUP BY Part 2

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

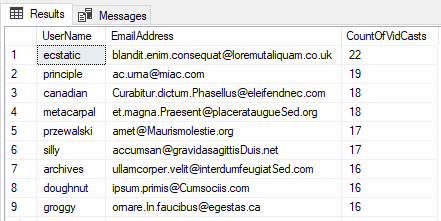
, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

ORDER BY CountOfVidCasts DESC, vc\_User.UserName



1. HAVING

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

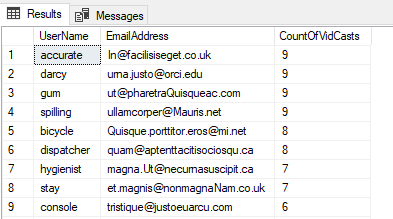
FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

HAVING COUNT(vc\_VidCast.vc\_VidCastID) <10

ORDER BY CountOfVidCasts DESC, vc\_User.UserName



1. Exploratory Data Analysis

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, SUM(DATEDIFF(n, StartDateTime, EndDateTime)) AS SumActualDurationMinutes

FROM vc\_VidCast

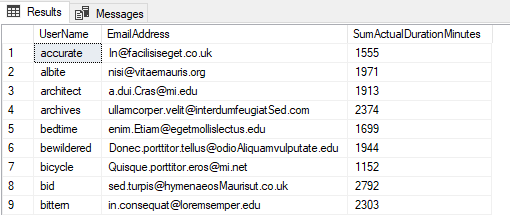
JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

JOIN vc\_Status ON vc\_Status.vc\_StatusID = vc\_VidCast.vc\_StatusID

WHERE vc\_Status.StatusText='Finished'

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

ORDER BY vc\_User.UserName



### Part 2 – Putting All Together

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, SUM(DATEDIFF(n, StartDateTime, EndDateTime)) as SumActualDurationMinutes

, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

, MIN(DATEDIFF(n, StartDateTime, EndDateTime)) as MinActualDurationMinutes

, AVG(DATEDIFF(n, StartDateTime, EndDateTime)) as AvgActualDurationMinutes

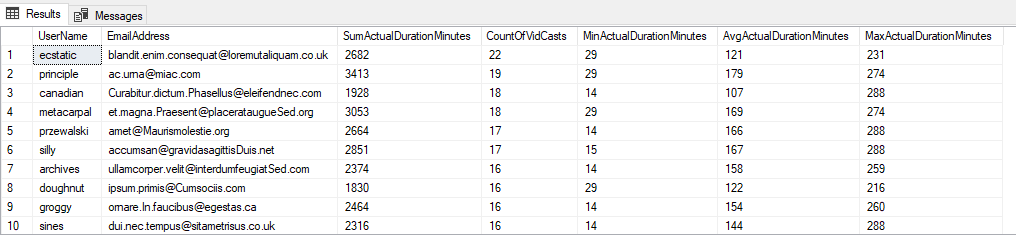
, MAX(DATEDIFF(n, StartDateTime, EndDateTime)) as MaxActualDurationMinutes

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

ORDER BY CountOfVidCasts DESC, vc\_User.UserName



### Query File

/\*

Name: Laura Larregui

Date: 2/19/2020

Lab 07: Advanced Query

\*/

--Exploratory Data Analysis

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, vc\_VidCast.vc\_VidCastID

FROM vc\_VidCast

JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

ORDER BY vc\_User.UserName

-- BASIC SUMMARIES

-- LOOK FOR USER THAT DON'T HAVE VIDEOS

SELECT \* FROM vc\_User

WHERE vc\_UserID NOT IN (SELECT vc\_UserID FROM vc\_VidCast)

--INCLUDES ALL USERS

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, vc\_VidCast.vc\_VidCastID

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

ORDER BY vc\_User.UserName

-- HIGH-LEVEL DESCRIPTIVE STATS

SELECT

COUNT(vc\_VidCastID) as NumberOfVidCasts

, SUM(ScheduleDurationMinutes) as TotalScheduledMinutes

, MIN(ScheduleDurationMinutes) as MinScheduledMinutes

, AVG(ScheduleDurationMinutes) as AvgScheduledMinutes

, MAX(ScheduleDurationMinutes) as MaxScheduledMinutes

FROM vc\_VidCast

--GROUP BY CLAUSE

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

ORDER BY CountOfVidCasts DESC, vc\_User.UserName

--HAVING CLAUSE

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

HAVING COUNT(vc\_VidCast.vc\_VidCastID) <10

ORDER BY CountOfVidCasts DESC, vc\_User.UserName

-- more EDA

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, SUM(DATEDIFF(n, StartDateTime, EndDateTime)) AS SumActualDurationMinutes

FROM vc\_VidCast

JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

JOIN vc\_Status ON vc\_Status.vc\_StatusID = vc\_VidCast.vc\_StatusID

WHERE vc\_Status.StatusText='Finished'

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

ORDER BY vc\_User.UserName

-- Part 2: Putting All together

SELECT

vc\_User.UserName

, vc\_User.EmailAddress

, SUM(DATEDIFF(n, StartDateTime, EndDateTime)) as SumActualDurationMinutes

, COUNT(vc\_VidCast.vc\_VidCastID) AS CountOfVidCasts

, MIN(DATEDIFF(n, StartDateTime, EndDateTime)) as MinActualDurationMinutes

, AVG(DATEDIFF(n, StartDateTime, EndDateTime)) as AvgActualDurationMinutes

, MAX(DATEDIFF(n, StartDateTime, EndDateTime)) as MaxActualDurationMinutes

FROM vc\_VidCast

RIGHT JOIN vc\_User ON vc\_User.vc\_UserID = vc\_VidCast.vc\_UserID

GROUP BY vc\_User.UserName, vc\_User.EmailAddress

ORDER BY CountOfVidCasts DESC, vc\_User.UserName