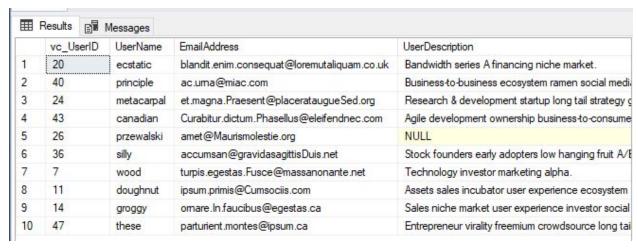
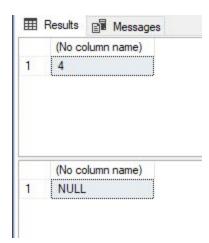
Function





Describe what lines 49 - 53: It's running the function we just created on lines 30 - 47 (named vc_VidCastCount, accepting @userID as parameters). Specifically, it's running the function 10 times, and then ordering by count of VidCasts.

How does the code know that @userID 20 has 22 vidcasts? On line 51 in the SELECT statement, that's running the function created on lines 30 - 47 with vc_UserID as the @userID parameter. So we are inputting @userID 20 into the function which then, when executed, counts the number of times that user ID is found in the vc_VidCast database.



Describe what lines 75 and 76: The function created on line 57 with the inputs 'Music' and 'Tunes'. Specifically, on line 66, the function runs a SELECT statement which selects the tagID from our vc_Tag db where the TagText = our input, in this case 'Music' or 'Tunes'.

Also, when line 76 executed, we received a NULL from SQL Server. How come? There are no tags with the text named 'Tunes'.



Describe what lines 79 through 87: Taking the select statement into a read-only view.



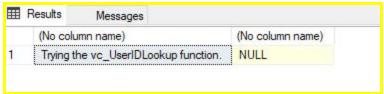
Describe what lines 91 through 104: Creating a function that takes two parameters, the userName and the newEmail and runs an update where the userName = the userName parameter.



Why is the UserLoginTimestamp value different from the lab document? Because the timestamp is recorded when the function is run, which for us is March 2021, but for the lab it's September 2020.

Simplify the code:

SELECT * from [dbo].[vc_AddUserLogin]('tardy', 'localhost')



	Results 🗐 Messa	ges
	TagText	VidCasts
1	Art	256
2	Audio Recording	266
3	Baseball	242
4	Basketball	236
5	Cat Videos	0
6	Collectibles	258
7	Consoles	260
8	Dog Videos	0
9	Fashion	240
10	Football	259
11	Games	254
12	Motors	0
13	Music	237
14	Personal	263
15	Professional	264
16	Sports - General	235

	vc_UserID	UserName	EmailAddress	UserDescription	WebSiteURL	UserRegisteredDate
1	1	ethanol	Donec.tempus@penatibusetmagnis.co.uk	Agile development non-disclosure agreement equit	http://ethanol.vidcast659.site	2017-12-30 22:19:12.000
2	2	dispatcher	quam@aptenttacitisociosqu.ca	A/B testing handshake disruptive seed money info	http://dispatcher.vidcast659.site	2017-12-08 03:36:00.000
3	3	camel	mauris@massanon.edu	User experience founders branding entrepreneur it	http://camel.vidcast659.site	2017-08-14 03:21:36.000
4	4	infatuated	mollis@Nam.org	Lean startup launch party angel investor branding	http://infatuated.vidcast659.site	2017-06-07 17:02:24.000
5	5	hygienist	magna.Ut@necumasuscipit.ca	Business model canvas accelerator pivot network	http://hygienist.vidcast659.site	2017-03-17 23:16:48.000
6	6	tardy	kmstudent@syr.edu	Startup leverage growth hacking bootstrapping scr	http://tardy.vidcast659.site	2017-03-12 15:36:00.000
7	7	wood	turpis.egestas.Fusce@massanonante.net	Technology investor marketing alpha.	http://wood.vidcast659.site	2017-06-21 15:36:00.000
8	8	mallard	vel.lectus.Cum@veliteget.edu	Assets sales success bandwidth business model c	http://mallard.vidcast659.site	2017-09-15 19:55:12.000
9	9	lifted	eu@elitsed.net	NULL	http://lifted.vidcast659.site	2017-04-15 20:24:00.000
10	10	gum	ut@pharetraQuisqueac.com	Infographic incubator hypotheses client conversio	http://gum.vidcast659.site	2017-02-24 09:07:12.000
	22	1 1 2			1 1	0047040404000000

	Results 📳 Messa	ges
	TagText	VidCasts
1	Art	256
2	Audio Recording	266
3	Baseball	242
4	Basketball	236
5	Cat Videos	0
6	Collectibles	258
7	Consoles	260
8	Dog Videos	0
9	Fashion	240
10	Football	259
11	Games	254
12	Motors	0

	UserName	VidCastCount	TotalMinutes
1	ecstatic	22	2682
2	principle	19	3413
3	canadian	18	1928
4	metacarpal	18	3053
5	przewalski	17	2664
6	silly	17	2851
7	archives	16	2374
8	doughnut	16	1830
9	groggy	16	2464
10	sines	16	2316

	vc_Ta	gID	TagText	TagDescription
į.	17		SQL	Finally, a SQL Tag

٧	/c_VidCastID	VidCast Title	Start Date Time	EndDateTime	ScheduleDu	uration Minutes	RecordingU	RL vc_UserI	vc_Status	sID
1 1	859	Finally done with sprocs	2021-03-09 21:21:49.917	NULL	45		NULL	6	2	
12	o Macadin	16dCootTitle	Chart Data Tona	End Data Tono		Calcadada Di mati	an Min. too	December 1101	us HeadD	un Chatrain
	vc_VidCastID 859	VidCastTitle Finally done with sprocs	StartDateTime 2021-03-09 21:21:49.917	EndDateTime 2021-03-09 22:		Schedule Duration		Recording URL NULL	vc_UserID	vc_StatusIE

```
Create function dbo.vc_VidCastCount(@userID int)
Returns int as
Beain
Declare @returnValue int
Select @returnValue= Count(vc_UserID) from vc_VidCast
where vc VidCast.vc UserID=@userID
Return @returnValue
End
Go
Select Top 10
, dbo.vc_VidCastCount(vc_UserID) as VidCastCount
From vc User
Order by VidCastCount DESC
Create function dbo.vc_TagIDLookup(@tagText varchar(20))
Returns int as
Begin
Declare @returnValue int
Select @returnValue= vc_TagID from vc_Tag
where TagText= @tagText
Return @returnValue
End
Go
select dbo.vc_TagIDLookup('Music')
select dbo.vc TagIDLookup('Tunes')
Create view vc_MostProlificUsers as
Select Top 10
,dbo.vc_VidCastCount(vc_UserID) as VidCastCount
from vc User
Order By VidCastCount DESC
Go
select * from vc_MostProlificUsers
Create Procedure vc_ChangeUserEmail(@userName varchar(20), @newEmail
varchar(50))
AS
Begin
```

```
Update vc User set EmailAddress=@newEmail
Where UserName=@userName
End
Go
EXEC vc ChangeUserEmail 'tardy', 'kmstudent@syr.edu'
select * from vc_User where UserName= 'tardy'
Insert into vc Tag (TagText) Values ('Cat Videos')
Select * from vc_Tag where vc_TagID= @@IDENTITY
Insert into vc_Tag (TagText) VALUES ('Dog Videos')
select * from vc_Tag where vc_TagID= SCOPE_IDENTITY()
Create or Alter Procedure vc_AddUserLogin(@userName varchar(20), @loginFrom
varchar(50))
AS
Begin
Declare @userID int
Select @userID=vc UserID from vc User
where UserName= @userName
INSERT INTO vc_UserLogin (vc_UserID, LoginLocation)
Values(@userID, @loginFrom)
Return SCOPE IDENTITY()
END
GO
Declare @addedValue int
EXEC @addedValue= vc_AddUserLogin 'tardy', 'localhost'
Select
vc User.vc UserID,
vc User.UserName,
vc_UserLogin.UserLoginTimestamp,
vc UserLogin.LoginLocation
From vc User
join vc_UserLogin on vc_User.vc_UserID= vc_UserLogin.vc_UserID
where vc UserLoginID=@addedValue
SELECT * from [dbo].[vc AddUserLogin]('tardy', 'localhost')
--Part 2
Create function dbo.vc UserIDLookup(@userName varchar(20))
Returns int as
```

```
Begin
Declare @returnValue int
Return @returnValue
End
Go
select 'Trying the vc UserIDLookup function.', dbo.vc UserIDLookup('tardy')
--function called dbo.vc TagVidCastCount that calculates the count of vc VidCastIDs for
a given vc_TagID
CREATE FUNCTION dbo.vc TagVidCastCount(@tagID int)
RETURNS int AS
BEGIN
DECLARE @returnValue int
SELECT @returnValue = COUNT(vc TagID) FROM vc VidCastTagList
WHERE vc_VidCastTagList.vc_TagID = @tagID
RETURN @returnValue
END
GO
---Testing the TagVidCastCount Function
SELECT
vc_Tag.TagText
, dbo.vc TagVidCastCount(vc Tag.vc TagID) as VidCasts
FROM vc_Tag
-- function called vc_VidCastDuration that SUMs the total number of minutes of actual
duration for VidCasts with a Finished status given a vc UserID as a parameter.
CREATE FUNCTION dbo.vc_VidCastDuration(@userID int)
RETURNS int AS
BEGIN
DECLARE @returnValue int
SELECT @returnValue = SUM(DATEDIFF(n, StartDateTime, EndDateTime)) FROM
vc VidCast
WHERE vc VidCast.vc UserID = @userID
RETURN @returnValue
END
GO
-- Testing VidCastDuration function
SELECT
, dbo.vc_VidCastDuration(vc_UserID) as TotalMinutes
FROM vc User
```

```
-- Create a view called vc TagReport
CREATE VIEW vc_TagReport AS
SELECT
vc Tag.TagText
, dbo.vc_TagVidCastCount(vc_Tag.vc_TagID) as VidCasts
FROM vc_Tag
GO
-- Testng Tag Report View
SELECT
vc_Tag.TagText
, dbo.vc_TagVidCastCount(vc_Tag.vc_TagID) as VidCasts
FROM vc_Tag
--Alter the view called vc MostProlificUsers
ALTER VIEW vc_MostProlificUsers AS
SELECT TOP 10
, dbo.vc_VidCastCount(vc_UserID) as VidCastCount
, dbo.vc_VidCastDuration(vc_UserID) as TotalMinutes
FROM vc User
ORDER BY VidCastCount DESC
GO
--Testing Alter View
SELECT UserName, VidCastCount, TotalMinutes FROM vc MostProlificUsers
--Stored Procedure
CREATE PROCEDURE vc_AddTag(@tagText varchar(20), @description
varchar(100)=NULL) AS
BEGIN
INSERT INTO vc_Tag (TagText, TagDescription)
VALUES(@tagText, @description)
RETURN @@identity
END
--Test procedure
DECLARE @newTagID int
EXEC @newTagID = vc_AddTag 'SQL', 'Finally, a SQL Tag'
SELECT * FROM vc Tag where vc TagID = @newTagID
--Stored procedure called vc FinishVidCast
CREATE PROCEDURE vc FinishVidCast(@vidCastID int)
AS
```

```
BEGIN
UPDATE vc_VidCast SET EndDateTime = GetDate(), vc_StatusID = 3
WHERE vc_VidCastID = @vidCastID
END
GO
-- Execute the procedure
DECLARE @newVC int
INSERT INTO vc_VidCast
(VidCastTitle, StartDateTime, ScheduleDurationMinutes, vc_UserID,
vc_StatusID)
VALUES (
'Finally done with sprocs'
, DATEADD(n, -45, GETDATE())
, 45
, (SELECT vc_UserID FROM vc_User WHERE UserName = 'tardy')
, (SELECT vc_StatusID FROM vc_Status WHERE StatusText='Started')
SET @newVC = SCOPE_IDENTITY()
SELECT * FROM vc_VidCast WHERE vc_VidCastID = @newVC
EXEC vc FinishVidCast @newVC
SELECT * FROM vc_VidCast WHERE vc_VidCastID = @newVC
```