

Lab 12: Functions

Nidhi Surya Prakash
SUID:215895619

Purpose:

Create a function and execute it.

Deliverables:

Multiple screenshots, as described through the lab.
Scripts that you used to carry out the actions.

Steps:

1. Create a function that will accept CourseId and Numerical Grade and will calculate the letter grade. The function will return the appropriate *Letter Grade Description* based on the input and the **CourseGrade & LetterGrades** table¹² you created in Lab 7 & 8. Provide a **screenshot** of the function being created.

Script:

```
CREATE FUNCTION calcGrade3(@courseid AS INT, @numgrade AS INT)
RETURNS VARCHAR(256)
AS
BEGIN
DECLARE @letterGradeID INT
DECLARE @result varchar(256)
SELECT @letterGradeID = (SELECT TOP 1 LetterGradeID FROM nsuryapr.CourseGrade
                        WHERE Courseid = @courseid and GradeValue <= @numgrade);
SELECT @result = (SELECT Description FROM nsuryapr.LetterGrades WHERE LetterGradeID =
@letterGradeID);
RETURN @result
end
```

¹ Please consider how grades are calculated. For example, if 90-92.9 is an A-, and you get a 92, then you get an A-.

² Please do not hardcode values into your function. Use the table to make the decisions.

SERVERS

Search server names

- nsuryapr.CourseGrade
- nsuryapr.Courses
- nsuryapr.LetterGrades
 - Columns
 - LetterGradeId (PK, int, not null)
 - LetterGrade (varchar(2), not null)
 - Description (varchar(50), null)
 - Keys
 - Constraints
 - Triggers
 - Indexes
 - Statistics
- nsuryapr.users
- pmjaradi.CourseEnrollment
- pmjaradi.CourseGrade
- pmjaradi.Courses
- pmjaradi.LetterGrades
- pmjaradi.Users
- pperez01.CourseEnrollment
- pperez01.CourseGrade
- pperez01.Courses
- pperez01.LetterGrades
- pperez01.Users
- pxu105.CourseEnrollment
- pxu105.CourseGrade
- pxu105.Courses
- pxu105.LetterGrades
- pxu105.Users

nsuryapr_lab12.sql

SQLQuery_1 • SQLQuery_2 •

Run Cancel Disconnect Change Connection CSE581labs Explain

```
1 CREATE FUNCTION calcGrade3(@courseid AS INT, @numgrade AS INT)
2 RETURNS VARCHAR(256)
3 AS
4 BEGIN
5 DECLARE @letterGradeID INT
6 DECLARE @result varchar(256)
7 SELECT @letterGradeID = (SELECT TOP 1 LetterGradeId FROM nsuryapr.CourseGrade
8 WHERE CourseId = @courseid and GradeValue <= @numgrade);
9 SELECT @result = (SELECT Description FROM nsuryapr.LetterGrades WHERE LetterGradeId = @letterGradeID);
10 RETURN @result
11 end
12
13
```

MESSAGES

12:43:52 AM Started executing query at Line 1
Commands completed successfully.
Total execution time: 00:00:00.034

0 rows 00:00:00 LCS-VC-MSSQLad.syr.edu : CSE581labs Ln 11, Col 4 (414 selected) Spaces: 4 UTF-8 LF SQL MSSQL

- Take a **screenshot** of the table containing Letter Grade Description, Course Id and Numerical grade, so that way both you and I can check that the function works properly.

SCRIPT:

SELECT * FROM nsuryapr.CourseGrade

SELECT * FROM nsuryapr.LetterGrades

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Servers' tree is expanded to 'nsuryapr.LetterGrades', showing columns: LetterGradeId (PK, int, not null), LetterGrade (varchar(2), not null), and Description (varchar(50), null). The main window displays a SQL query window with the following script:

```
1 CREATE FUNCTION calcGrade3(@courseid AS INT, @numgrade AS INT)
2 RETURNS VARCHAR(256)
3 AS
4 BEGIN
5 DECLARE @LetterGradeID INT
6 DECLARE @result varchar(256)
7 SELECT @LetterGradeID = (SELECT TOP 1 LetterGradeId FROM nsuryapr.CourseGrade
8 WHERE CourseId = @courseid and GradeValue <= @numgrade);
9 SELECT @result = (SELECT Description FROM nsuryapr.LetterGrades WHERE LetterGradeId = @LetterGradeID);
10 RETURN @result
11 end
12
13
14 SELECT * FROM nsuryapr.CourseGrade
15 SELECT * FROM nsuryapr.LetterGrades
```

The 'RESULTS' pane shows a table with 12 rows and 5 columns: GradingScaleId, LetterGradeId, CourseId, GradeValue, and Description. The data is as follows:

GradingScaleId	LetterGradeId	CourseId	GradeValue	Description
1	1	1	95	
2	2	1	90	
3	3	1	80	
4	4	1	70	
5	5	1	60	
6	6	1	0	
7	7	2	100	
8	8	2	90	
9	9	2	85	
10	10	2	75	
11	11	2	64	
12	12	2	0	

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Servers' tree is expanded to 'nsuryapr.LetterGrades', showing columns: LetterGradeId (PK, int, not null), LetterGrade (varchar(2), not null), and Description (varchar(50), null). The main window displays a SQL query window with the following script:

```
2 RETURNS VARCHAR(256)
3 AS
4 BEGIN
5 DECLARE @LetterGradeID INT
6 DECLARE @result varchar(256)
7 SELECT @LetterGradeID = (SELECT TOP 1 LetterGradeId FROM nsuryapr.CourseGrade
8 WHERE CourseId = @courseid and GradeValue <= @numgrade);
9 SELECT @result = (SELECT Description FROM nsuryapr.LetterGrades WHERE LetterGradeId = @LetterGradeID);
10 RETURN @result
11 end
12
13
14 SELECT * FROM nsuryapr.CourseGrade
15 SELECT * FROM nsuryapr.LetterGrades
```

The 'RESULTS' pane shows a table with 6 rows and 3 columns: LetterGradeId, LetterGrade, and Description. The data is as follows:

LetterGradeId	LetterGrade	Description
1	O	Outstanding
2	E	Exceeds Expec...
3	A	Acceptable
4	P	Poor
5	D	Dreadful
6	T	Troll

The 'MESSAGES' pane shows the following message:

```
12:44:30 AM Started executing query at Line 15
(6 rows affected)
Total execution time: 00:00:00.034
```

3. Run the function for the CourseId = 1, NumericalGrade = 95. Provide a **screenshot** of execution, showing the result.

SCRIPT: `SELECT nsuryapr.calcGrade3(1,95);`

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'SERVERS' tree shows the 'nsuryapr' database expanded, with the 'Columns' folder selected. The main pane shows the query editor with the following SQL script:

```
11  
12  
13  
14 SELECT * FROM nsuryapr.CourseGrade  
15 SELECT * FROM nsuryapr.LetterGrades  
16  
17  
18 SELECT nsuryapr.calcGrade3(1,95);  
19
```

The 'RESULTS' pane shows the output of the query:

(No column name)
1 Outstanding

The 'MESSAGES' pane shows the execution details:

```
12:44:45 AM Started executing query at Line 18  
(1 row affected)  
Total execution time: 00:00:00.037
```

The status bar at the bottom indicates '1 rows', '00:00:00' execution time, and the connection 'LCS-VC-MSSQL.ad.syr.edu : CSE581labs'.

4. Run the function for the CourseId = 2, NumericalGrade = 95. Provide a **screenshot** of execution, showing the result.

SCRIPT: `SELECT nsuryapr.calcGrade3(2,95);`

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'SERVERS' pane shows a tree view of the 'nsuryapr' database, including tables like 'CourseGrade', 'Courses', and 'LetterGrades', as well as columns and keys. The main window shows a query script with the following lines:

```
13
14 SELECT * FROM nsuryapr.CourseGrade
15 SELECT * FROM nsuryapr.LetterGrades
16
17
18 SELECT nsuryapr.calcGrade3(1,95);
19
20 SELECT nsuryapr.calcGrade3(2,95);
21
```

The query is executed, and the 'RESULTS' pane shows a single row with the message 'Exceeds Expectations'. The 'MESSAGES' pane shows the execution details:

12:45:05 AM Started executing query at Line 20
(1 row affected)
Total execution time: 00:00:00.037

The status bar at the bottom indicates '1 rows', '00:00:00' execution time, and the connection 'LCS-VC-MSSQL.ad.syr.edu : CSE581labs'.

5. Run the function for the CourseId = 2, NumericalGrade = 85. Provide a **screenshot** of execution, showing the result.

SCRIPT: `SELECT nsuryapr.calcGrade3(2,85);`

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'SERVERS' pane shows a tree view of the 'nsuryapr' database, with 'nsuryapr.LetterGrades' selected. The 'Columns' list for 'LetterGrades' is expanded, showing 'LetterGradeId (PK, int, not null)', 'LetterGrade (varchar(2), not null)', and 'Description (varchar(50), null)'. The 'Keys' list is also expanded, showing 'LetterGradeId (PK, int, not null)'. The 'Messages' pane at the bottom shows the execution of the query 'SELECT nsuryapr.calcGrade3(2,85);' at 12:45:29 AM, resulting in 1 row affected. The status bar at the bottom indicates '1 rows', '00:00:00' execution time, and 'Ln 23, Col 1 (34 selected)'.

RESULTS
(No column name)
1 Acceptable

MESSAGES
12:45:29 AM Started executing query at Line 22 (1 row affected) Total execution time: 00:00:00.032