

# Exercise 8 – Data Definition Language

Stewart Johnston  
CIS 150 – Intro to Database Administration  
NCMC  
johnstons1@student.ncmich.edu

October 29, 2018

## Contents

<b>1</b>	<b>Add an index to the MyGuitarShop for zip code</b>	<b>1</b>
<b>2</b>	<b>Implement a design on a database</b>	<b>3</b>
<b>3</b>	<b>Add rows to database</b>	<b>4</b>
<b>4</b>	<b>Alter Products table</b>	<b>6</b>
<b>5</b>	<b>Alter FirstName column in Users</b>	<b>7</b>
5.1	Failed update command for Users . . . . .	8
5.2	Failed update command for Users 2: electric boogaloo . . . . .	9

## List of Figures

1	Per 1 . . . . .	2
2	Per 2 . . . . .	4
3	Per 3 . . . . .	6
4	Per 4 . . . . .	7
5	Per 5 . . . . .	8
6	Per 5.1 . . . . .	9
7	Per 5.2 . . . . .	10

## 1 Add an index to the MyGuitarShop for zip code

See figure 1

```

USE MyGuitarShop;

CREATE INDEX IX_ZipCode
    ON Addresses (ZipCode);

SELECT *
    FROM sys.indexes
    WHERE name = 'IX_ZipCode';

```

Figure 1: Per 1

The screenshot shows a SQL Server Enterprise Manager window with the following components:

- SQLQuery1.sql - EC2AMAZ-3ISQLLJ\scj (55)\***: The title bar of the query window.
- Query Text**: The SQL code entered in the editor:
 

```

USE MyGuitarShop;

CREATE INDEX IX_ZipCode
    ON Addresses (ZipCode);

SELECT *
    FROM sys.indexes
    WHERE name = 'IX_ZipCode';

```
- Results**: A table showing the output of the SELECT statement.
 

	object_id	name	index_id	type	type_desc	is_unique	data_space_id	ignore_dup_key	is_primary_key
1	1125579048	IX_ZipCode	2	2	NONCLUSTERED	0	1	0	0
- Status Bar**: A green bar at the bottom indicating "Query executed successf..." and providing details: "EC2AMAZ-3ISQLLJ (14.0 RTM) | EC2AMAZ-3ISQLLJ\scj (55) | MyGuitarShop | 00:00:00 | 1 rows".

## 2 Implement a design on a database

See figure 2

```
IF DB_ID('MyWebDB') IS NOT NULL
    DROP DATABASE MyWebDB;
CREATE DATABASE MyWebDB;
USE MyWebDB;

CREATE TABLE Users(
    UserID INT PRIMARY KEY IDENTITY
    ,EmailAddress VARCHAR(50) NOT NULL
    ,FirstName VARCHAR(50) NOT NULL
    ,LastName VARCHAR(50) NOT NULL);

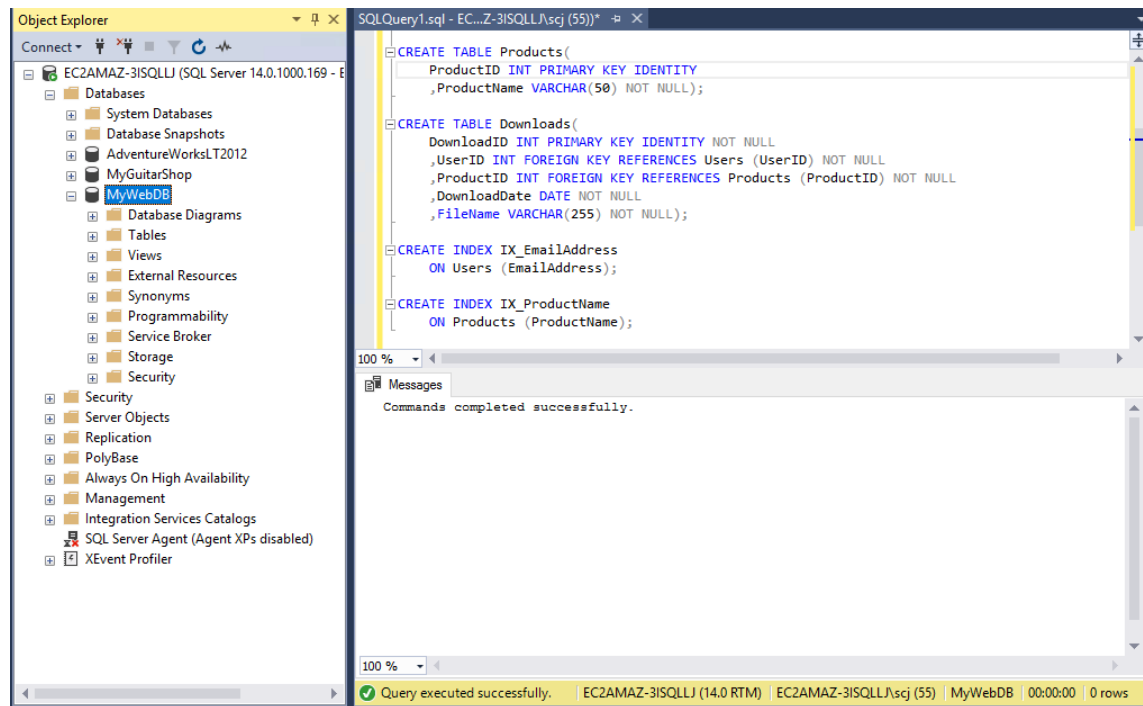
CREATE TABLE Products(
    ProductID INT PRIMARY KEY IDENTITY
    ,ProductName VARCHAR(50) NOT NULL);

CREATE TABLE Downloads(
    DownloadID INT PRIMARY KEY IDENTITY NOT NULL
    ,UserID INT FOREIGN KEY REFERENCES Users (UserID) NOT NULL
    ,ProductID INT FOREIGN KEY REFERENCES Products (ProductID) NOT NULL
    ,DownloadDate DATETIME2(7) NOT NULL
    ,FileName VARCHAR(255) NOT NULL);

CREATE INDEX IX_EmailAddress
    ON Users (EmailAddress);

CREATE INDEX IX_ProductName
    ON Products (ProductName);
```

Figure 2: Per 2



### 3 Add rows to database

See figure 3

```

USE MyWebDB;

INSERT INTO USERS (
    EmailAddress,
    FirstName,
    LastName )
VALUES ('johnsmith@gmail.com', 'John', 'Smith'),
       ('janedoe@yahoo.com', 'Jane', 'Doe');

INSERT INTO Products (
    ProductName )
VALUES ('Local_Music_Vol_2'),
       ('Local_Music_Vol_1');

INSERT INTO Downloads (
    UserID,
    ProductID,
    DownloadDate,
    FileName )
VALUES (1, 2, CURRENT_TIMESTAMP, 'petals_are_falling.mp3'),
       (2, 1, CURRENT_TIMESTAMP, 'turn_signal.mp3'),
       (2, 2, CURRENT_TIMESTAMP, 'one_horse_town.mp3');

SELECT EmailAddress, FirstName, LastName,
       DownloadDate, FileName,
       ProductName
FROM Users as usr
     INNER JOIN Downloads as dl
           ON usr.UserID = dl.UserID
     INNER JOIN Products as pd
           ON dl.ProductID = pd.ProductID
ORDER BY usr.EmailAddress DESC, pd.ProductName ASC;

```

Figure 3: Per 3

The screenshot shows a SQL query editor window titled "SQLQuery2.sql - EC2AMAZ-3ISQLLJ\scj (53)". The query contains three statements: two INSERT statements and one SELECT statement. The first INSERT statement inserts into the Products table. The second INSERT statement inserts into the Downloads table. The third SELECT statement joins the Users, Downloads, and Products tables to retrieve user information, download dates, file names, and product names, ordered by user email address in descending order and product name in ascending order.

```
INSERT INTO Products (
    ProductName )
VALUES ('Local Music Vol 2'),
      ('Local Music Vol 1');

INSERT INTO Downloads (
    UserID,
    ProductID,
    DownloadDate,
    FileName )
VALUES (1, 2, CURRENT_TIMESTAMP, 'petals_are_falling.mp3'),
      (2, 1, CURRENT_TIMESTAMP, 'turn_signal.mp3'),
      (2, 2, CURRENT_TIMESTAMP, 'one_horse_town.mp3');

SELECT EmailAddress, FirstName, LastName,
       DownloadDate, FileName,
       ProductName
FROM Users as usr
     INNER JOIN Downloads as dl
         ON usr.UserID = dl.UserID
     INNER JOIN Products as pd
         ON dl.ProductID = pd.ProductID
ORDER BY usr.EmailAddress DESC, pd.ProductName ASC;
```

The query results are displayed in a table with 7 columns: EmailAddress, FirstName, LastName, DownloadDate, FileName, and ProductName. The results show three rows of data.

	EmailAddress	FirstName	LastName	DownloadDate	FileName	ProductName
1	johnsmith@gmail.com	John	Smith	2018-10-30 00:58:54.6200000	petals_are_falling.mp3	Local Music Vol 1
2	janedoe@yahoo.com	Jane	Doe	2018-10-30 00:58:54.6200000	one_horse_town.mp3	Local Music Vol 1
3	janedoe@yahoo.com	Jane	Doe	2018-10-30 00:58:54.6200000	tum_signal.mp3	Local Music Vol 2

The status bar at the bottom indicates: "Query executed successfully. EC2AMAZ-3ISQLLJ (14.0 RTM) | EC2AMAZ-3ISQLLJ\scj (53) | MyWebDB | 00:00:00 | 3 rows".

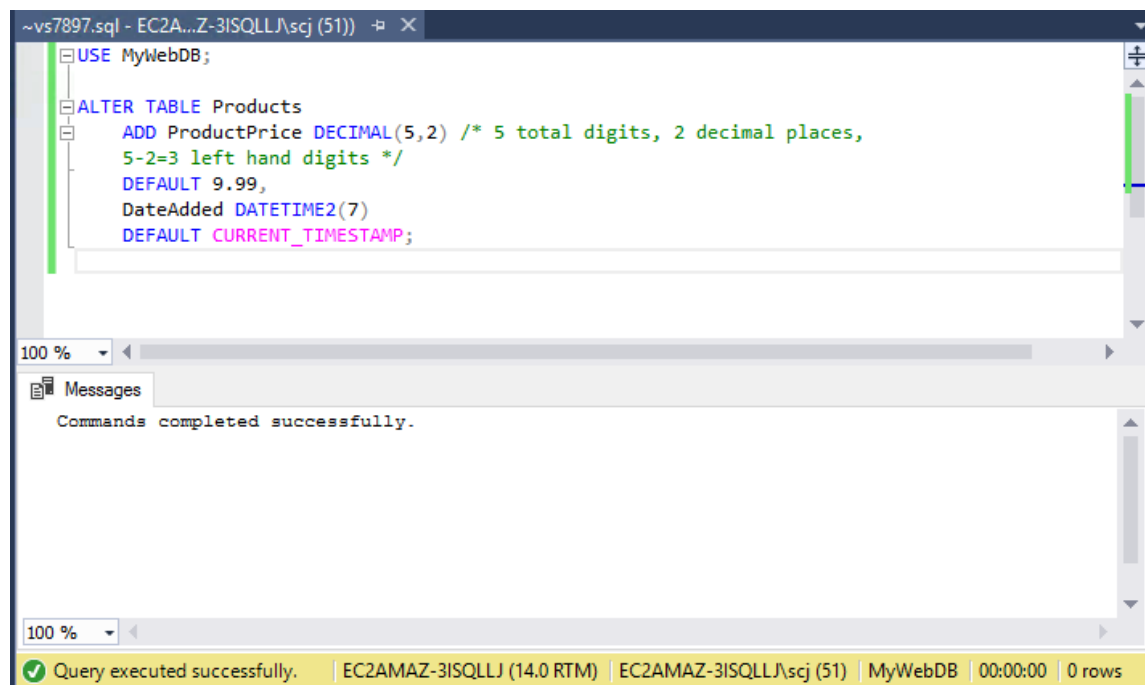
## 4 Alter Products table

See figure 4

```
USE MyWebDB;
```

```
ALTER TABLE Products  
    ADD ProductPrice DECIMAL(5,2) /* 5 total digits, 2 decimal places,  
    5-2=3 left hand digits */  
    DEFAULT 9.99,  
    DateAdded DATETIME2(7)  
    DEFAULT CURRENT_TIMESTAMP;
```

Figure 4: Per 4



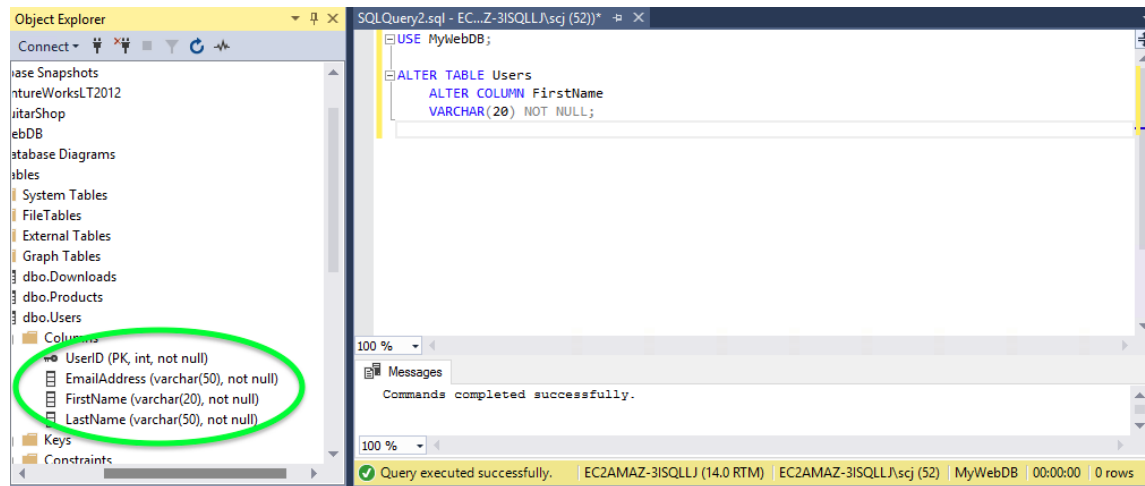
## 5 Alter FirstName column in Users

See figure 5

```
USE MyWebDB;
```

```
ALTER TABLE Users  
    ALTER COLUMN FirstName  
    VARCHAR(20) NOT NULL;
```

Figure 5: Per 5



## 5.1 Failed update command for Users

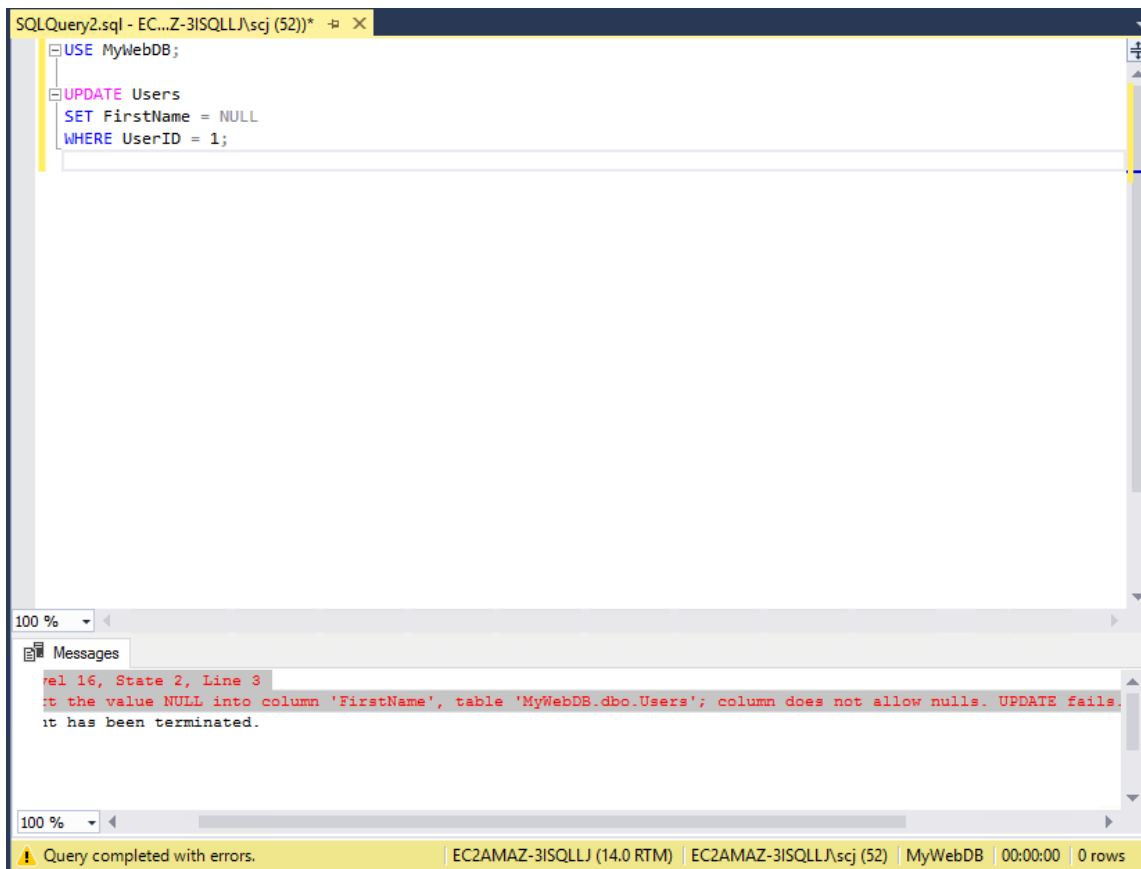
In which we attempt to update with a NULL in a value which is not allowed to be null  
See figure 6

```
USE MyWebDB;
```

```
UPDATE Users  
SET FirstName = NULL  
WHERE UserID = 1;
```



Figure 6: Per 5.1



## 5.2 Failed update command for Users 2: electric boogaloo

In which we attempt to update with a name which is too long for the data type  
See figure 7

```
USE MyWebDB;  
  
UPDATE Users  
SET Firstname = 'abcdefghijklmnopqrstuvwzyz'  
WHERE UserID = 1;
```

Figure 7: Per 5.2

