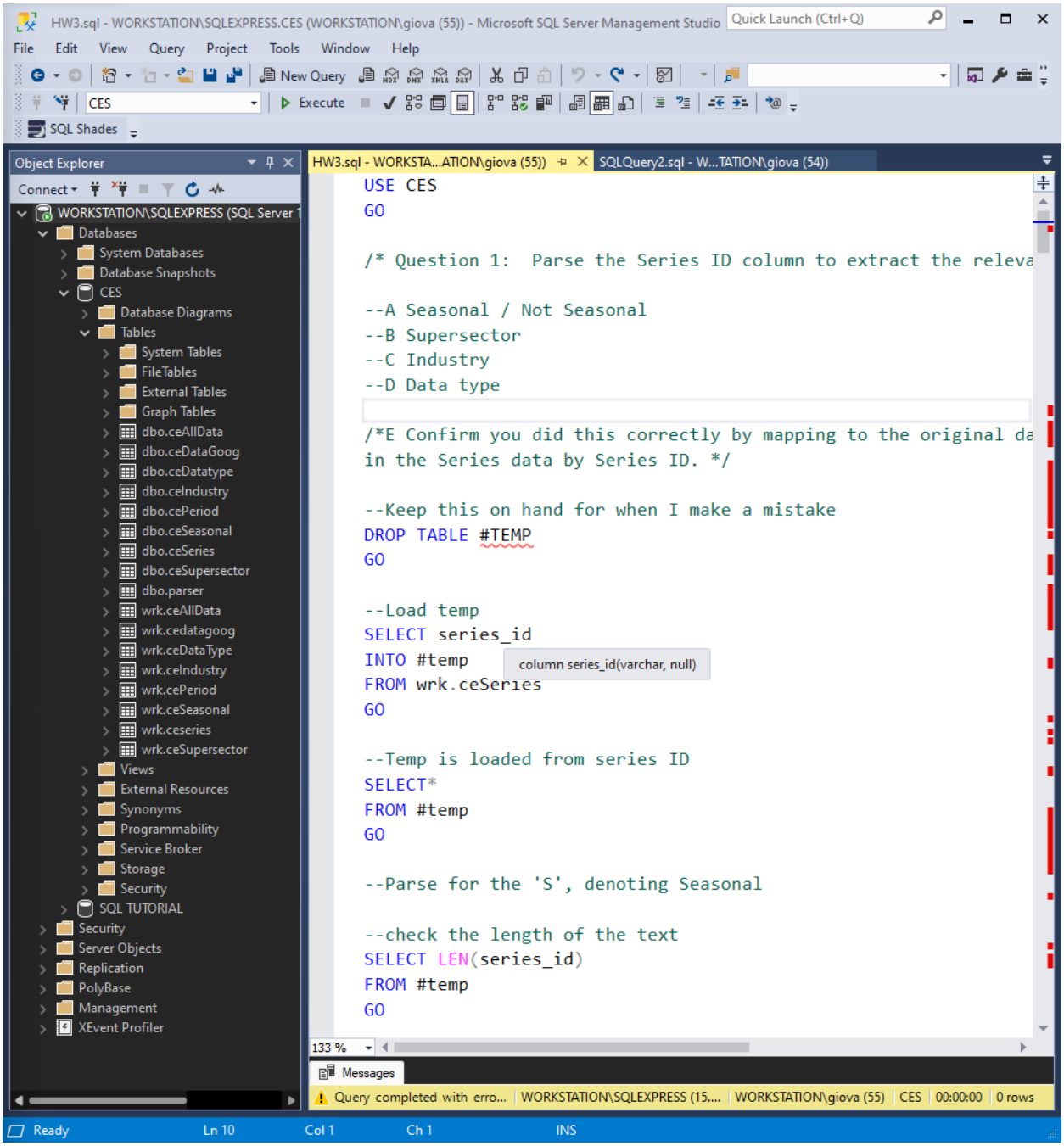
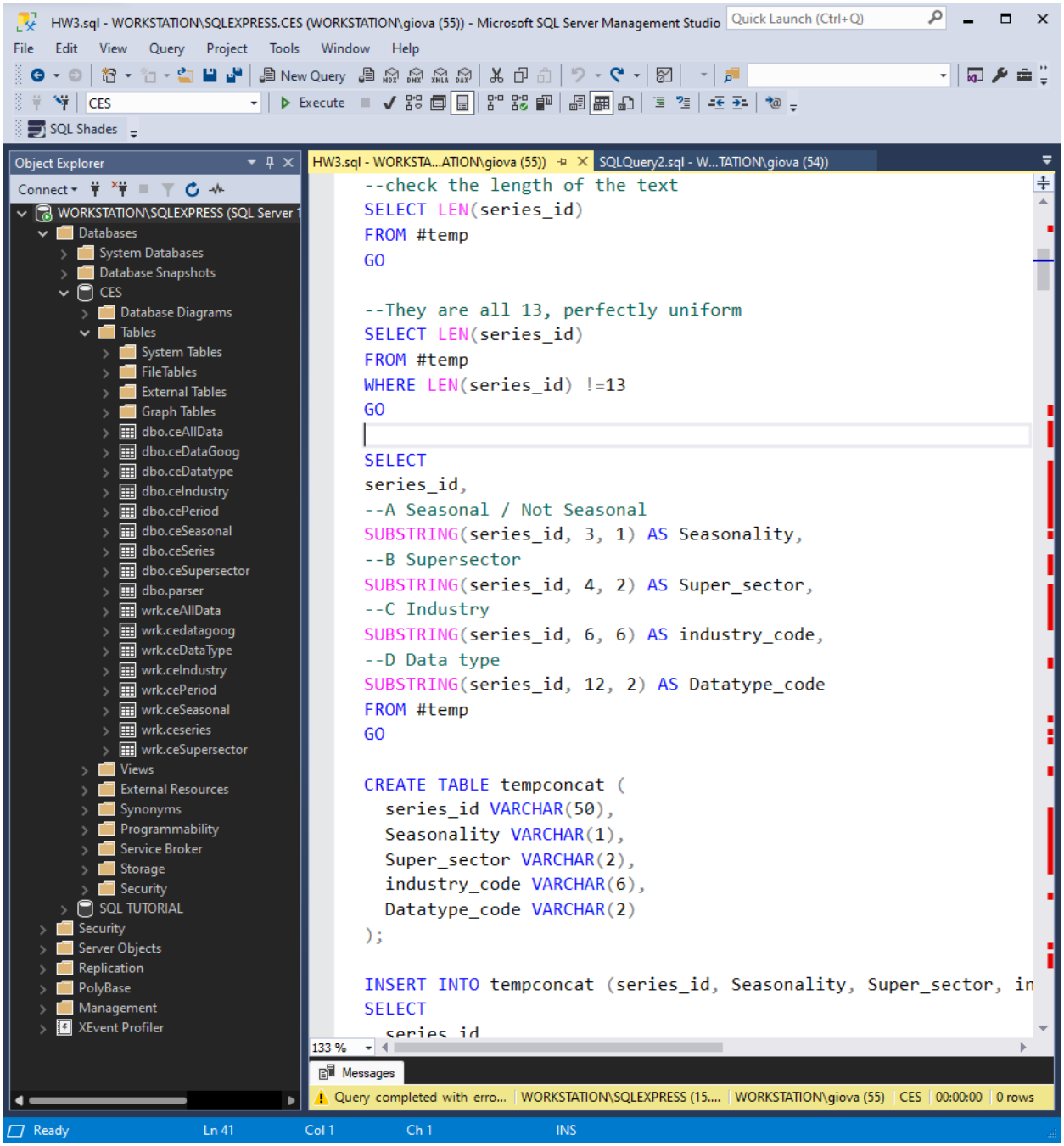
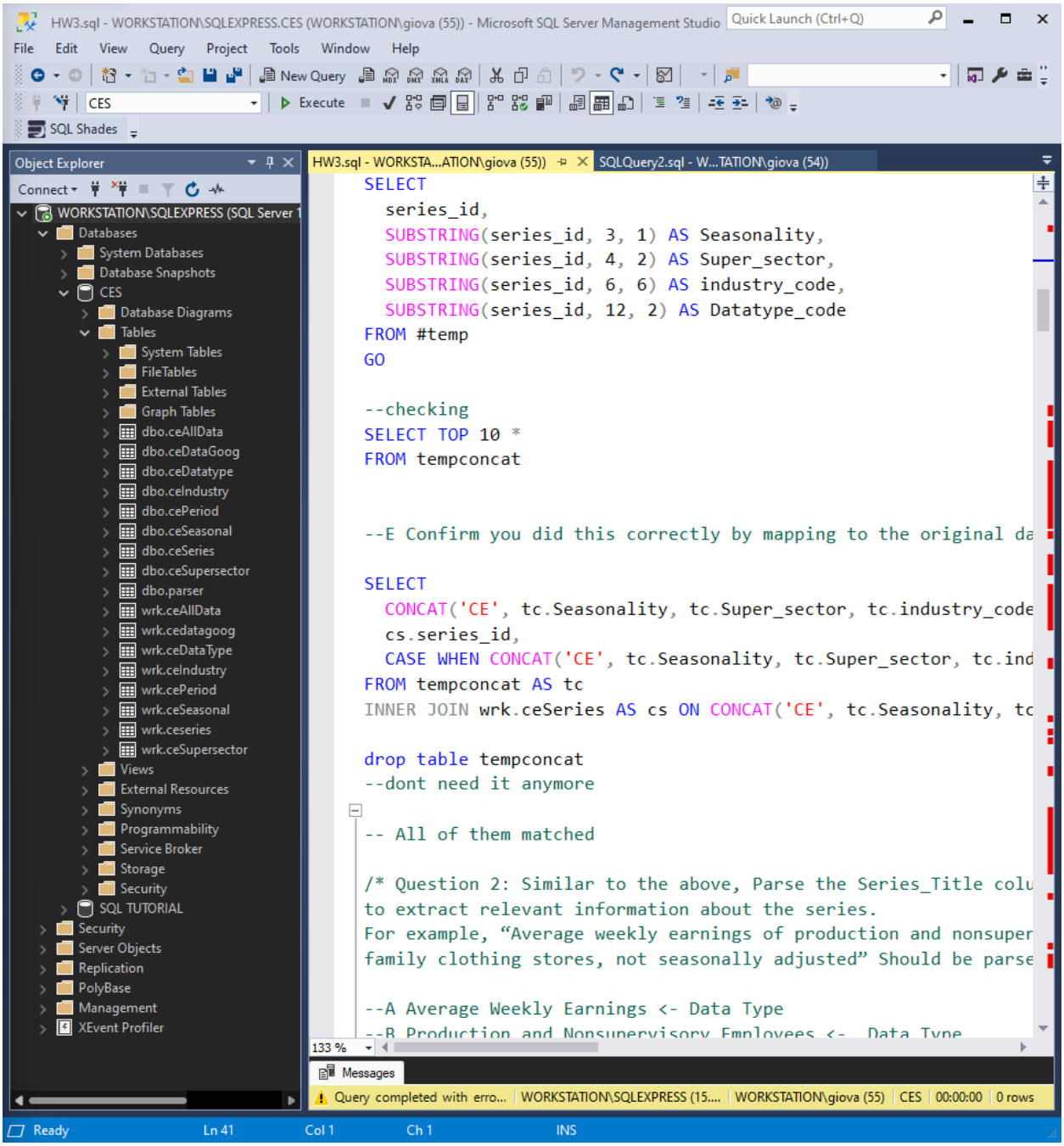
```
USE CES
GO

/* Question 1:  Parse the Series ID column to extract the relevant information. */

--A Seasonal / Not Seasonal
--B Supersector
--C Industry
--D Data type

/*E Confirm you did this correctly by mapping to the original data
in the Series data by Series ID. */

--Keep this on hand for when I make a mistake
DROP TABLE #TEMP
GO

--Load temp
SELECT series_id
INTO #temp
FROM wrk.ceSeries
GO

--Temp is loaded from series ID
SELECT*
FROM #temp
GO

--Parse for the 'S', denoting Seasonal

--check the length of the text
SELECT LEN(series_id)
FROM #temp
GO

--They are all 13, perfectly uniform
SELECT LEN(series_id)
FROM #temp
WHERE LEN(series_id) !=13
GO

SELECT
series_id,
--A Seasonal / Not Seasonal
SUBSTRING(series_id, 3, 1) AS Seasonality,
--B Supersector
SUBSTRING(series_id, 4, 2) AS Super_sector,
--C Industry
SUBSTRING(series_id, 6, 6) AS industry_code,
--D Data type
SUBSTRING(series_id, 12, 2) AS Datatype_code
FROM #temp
GO

CREATE TABLE tempconcat (
```

```
  series_id VARCHAR(50),
  Seasonality VARCHAR(1),
  Super_sector VARCHAR(2),
  industry_code VARCHAR(6),
  Datatype_code VARCHAR(2)
);

INSERT INTO tempconcat (series_id, Seasonality, Super_sector, industry_code,
Datatype_code)
SELECT
  series_id,
  SUBSTRING(series_id, 3, 1) AS Seasonality,
  SUBSTRING(series_id, 4, 2) AS Super_sector,
  SUBSTRING(series_id, 6, 6) AS industry_code,
  SUBSTRING(series_id, 12, 2) AS Datatype_code
FROM #temp
GO

--checking
SELECT TOP 10 *
FROM tempconcat


--E Confirm you did this correctly by mapping to the original data in the Series data by
Series ID.

SELECT
  CONCAT('CE', tc.Seasonality, tc.Super_sector, tc.industry_code, tc.Datatype_code) AS
concatenated_column,
  cs.series_id,
  CASE WHEN CONCAT('CE', tc.Seasonality, tc.Super_sector, tc.industry_code,
tc.Datatype_code) = cs.series_id THEN 'Matched' ELSE 'Not Matched' END AS matching_status
FROM tempconcat AS tc
INNER JOIN wrk.ceSeries AS cs ON CONCAT('CE', tc.Seasonality, tc.Super_sector,
tc.industry_code, tc.Datatype_code) = cs.series_id;

drop table tempconcat
--dont need it anymore

-- All of them matched
```


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55)) - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))
SQLQuery2.sql - W...TATION\giova (54))
USE CES
GO
/* Question 1: Parse the Series ID column to extract the releva
--A Seasonal / Not Seasonal
--B Supersector
--C Industry
--D Data type
/*E Confirm you did this correctly by mapping to the original da
in the Series data by Series ID. */
--Keep this on hand for when I make a mistake
DROP TABLE #TEMP
GO
--Load temp
SELECT series_id
INTO #temp
column series_id(varchar, null)
FROM wrk.ceSeries
GO
--Temp is loaded from series ID
SELECT*
FROM #temp
GO
--Parse for the 'S', denoting Seasonal
--check the length of the text
SELECT LEN(series_id)
FROM #temp
GO
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 10
Col 1
Ch 1
INS

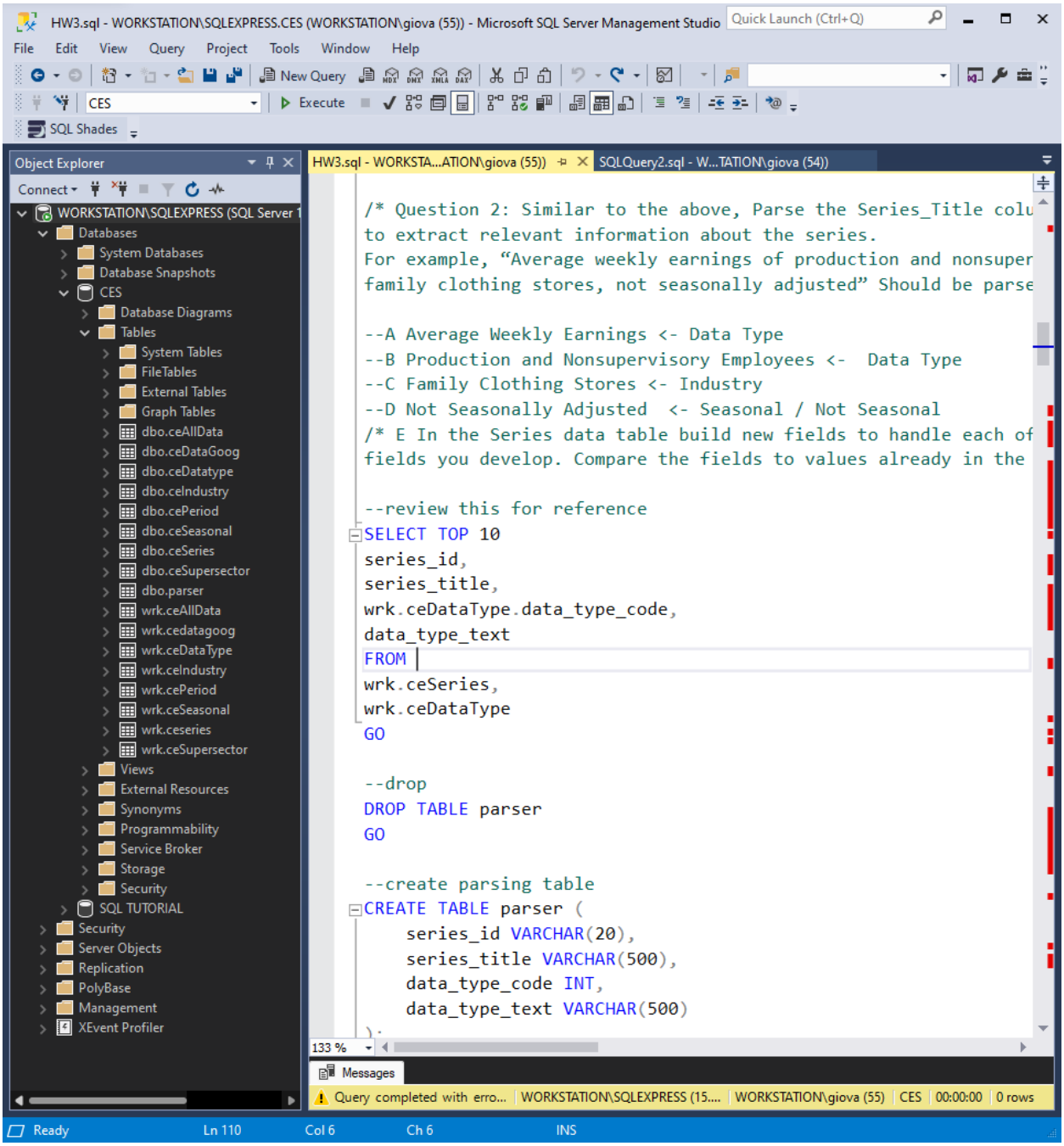

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55)) - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Execute
CES
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))
SQLQuery2.sql - W...TATION\giova (54))
--check the length of the text
SELECT LEN(series_id)
FROM #temp
GO
--They are all 13, perfectly uniform
SELECT LEN(series_id)
FROM #temp
WHERE LEN(series_id) !=13
GO
SELECT
series_id,
--A Seasonal / Not Seasonal
SUBSTRING(series_id, 3, 1) AS Seasonality,
--B Supersector
SUBSTRING(series_id, 4, 2) AS Super_sector,
--C Industry
SUBSTRING(series_id, 6, 6) AS industry_code,
--D Data type
SUBSTRING(series_id, 12, 2) AS Datatype_code
FROM #temp
GO
CREATE TABLE tempconcat (
series_id VARCHAR(50),
Seasonality VARCHAR(1),
Super_sector VARCHAR(2),
industry_code VARCHAR(6),
Datatype_code VARCHAR(2)
);
INSERT INTO tempconcat (series_id, Seasonality, Super_sector, in
SELECT
series_id
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 41
Col 1
Ch 1
INS

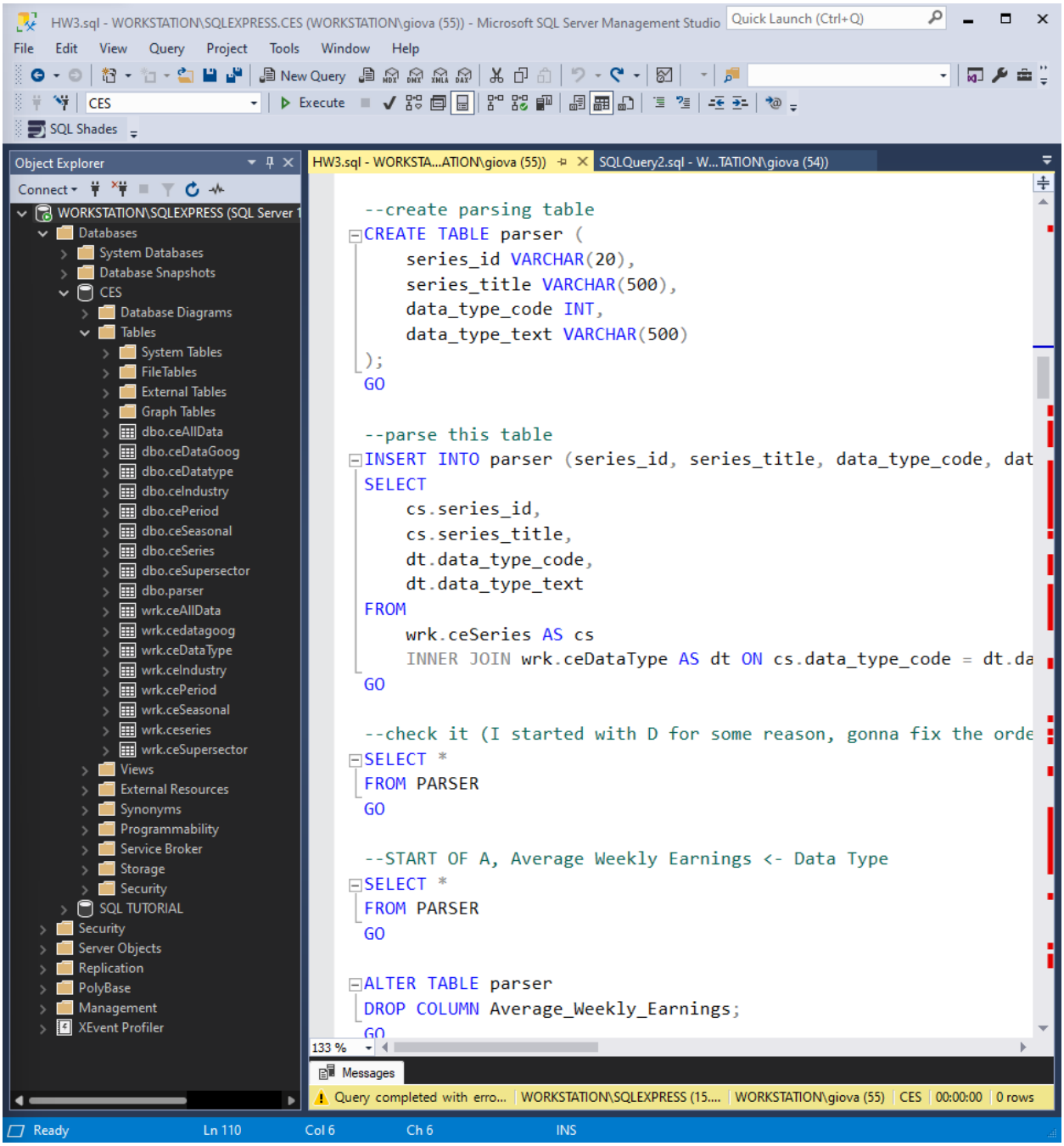


/* Question 2: Similar to the above, Parse the Series_Title column
to extract relevant information about the series.

For example, “Average weekly earnings of production and nonsupervisory employees, family clothing stores, not seasonally adjusted” Should be parsed into the following: */

```
--A Average Weekly Earnings <- Data Type
--B Production and Nonsupervisory Employees <-  Data Type
--C Family Clothing Stores <- Industry
--D Not Seasonally Adjusted  <- Seasonal / Not Seasonal
/* E In the Series data table build new fields to handle each of the above
fields you develop. Compare the fields to values already in the table. */

--review this for reference
SELECT TOP 10
series_id,
series_title,
wrk.ceDataType.data_type_code,
data_type_text
FROM
wrk.ceSeries,
wrk.ceDataType
GO

--drop
DROP TABLE parser
GO

--create parsing table
CREATE TABLE parser (
    series_id VARCHAR(20),
    series_title VARCHAR(500),
    data_type_code INT,
    data_type_text VARCHAR(500)
);
GO

--parse this table
INSERT INTO parser (series_id, series_title, data_type_code, data_type_text)
SELECT
    cs.series_id,
    cs.series_title,
    dt.data_type_code,
    dt.data_type_text
FROM
    wrk.ceSeries AS cs
    INNER JOIN wrk.ceDataType AS dt ON cs.data_type_code = dt.data_type_code;
GO
```

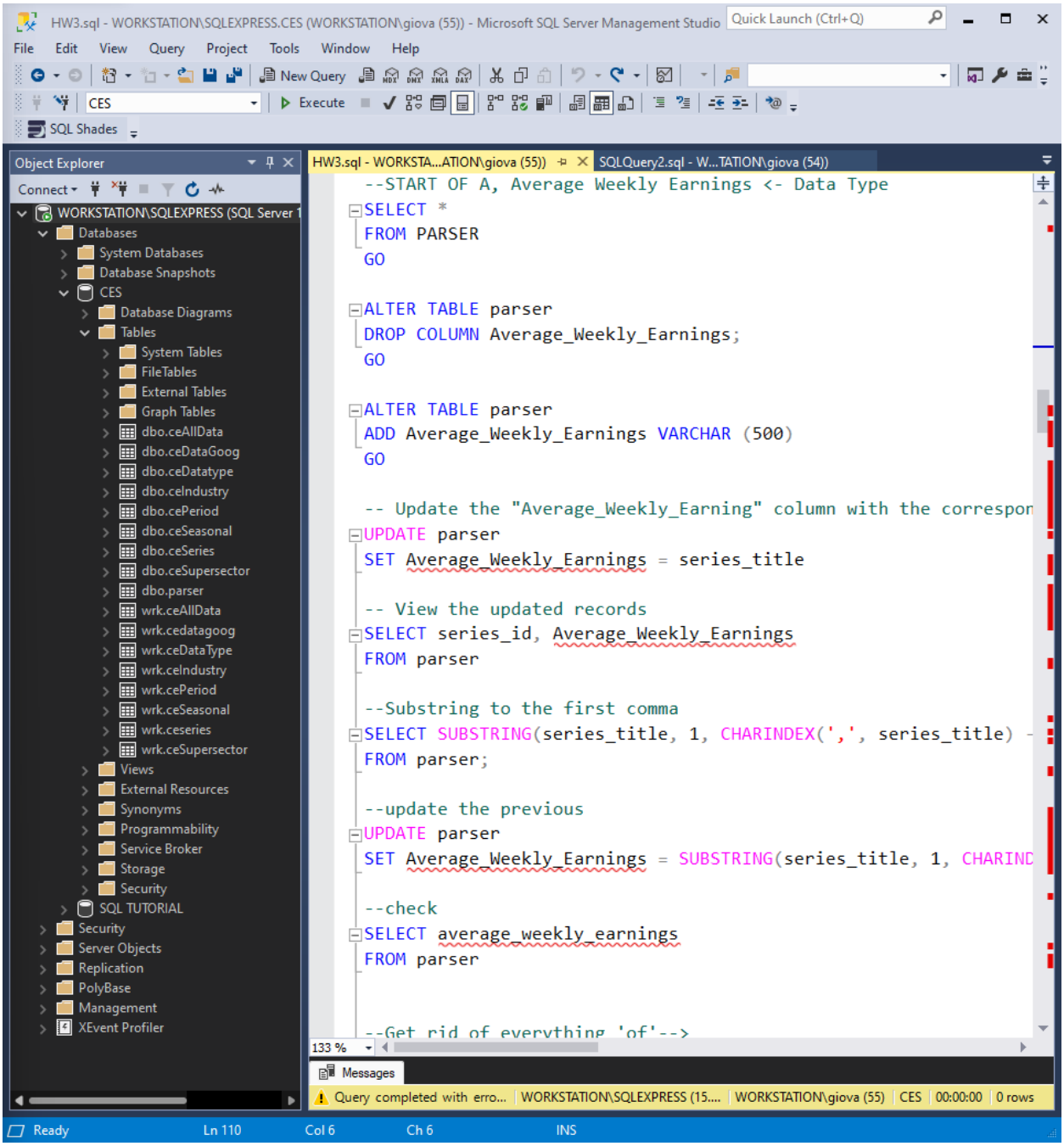

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55)) - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))
SQLQuery2.sql - W...TATION\giova (54))
/* Question 2: Similar to the above, Parse the Series_Title colu
to extract relevant information about the series.
For example, “Average weekly earnings of production and nonsuper
family clothing stores, not seasonally adjusted” Should be parse
--A Average Weekly Earnings <- Data Type
--B Production and Nonsupervisory Employees <- Data Type
--C Family Clothing Stores <- Industry
--D Not Seasonally Adjusted <- Seasonal / Not Seasonal
/* E In the Series data table build new fields to handle each of
fields you develop. Compare the fields to values already in the
--review this for reference
SELECT TOP 10
series_id,
series_title,
wrk.ceDataType.data_type_code,
data_type_text
FROM
wrk.ceSeries,
wrk.ceDataType
GO
--drop
DROP TABLE parser
GO
--create parsing table
CREATE TABLE parser (
series_id VARCHAR(20),
series_title VARCHAR(500),
data_type_code INT,
data_type_text VARCHAR(500)
);
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 110
Col 6
Ch 6
INS

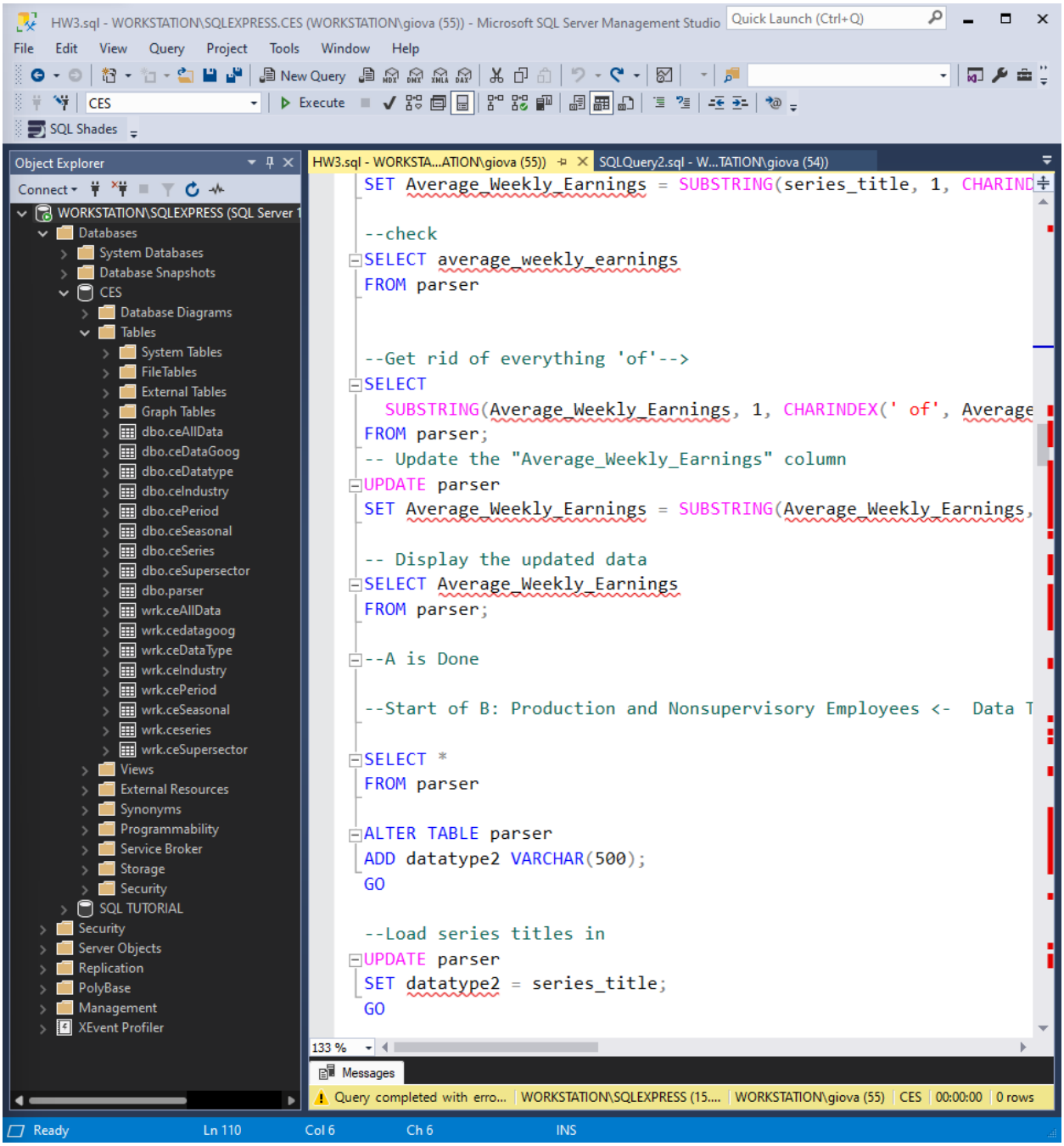



```
--START OF A, Average Weekly Earnings <- Data Type
SELECT *
FROM PARSER
GO

ALTER TABLE parser
DROP COLUMN Average_Weekly_Earnings;
GO
```

```
ALTER TABLE parser
ADD Average_Weekly_Earnings VARCHAR (500)
GO

-- Update the "Average_Weekly_Earning" column with the corresponding values
UPDATE parser
SET Average_Weekly_Earnings = series_title

-- View the updated records
SELECT series_id, Average_Weekly_Earnings
FROM parser

--Substring to the first comma
SELECT SUBSTRING(series_title, 1, CHARINDEX(',', series_title) - 1) AS First_Part
FROM parser;

--update the previous
UPDATE parser
SET Average_Weekly_Earnings = SUBSTRING(series_title, 1, CHARINDEX(',', series_title) -
1)

--check
SELECT average_weekly_earnings
FROM parser


--Get rid of everything 'of'-->
SELECT
  SUBSTRING(Average_Weekly_Earnings, 1, CHARINDEX(' of', Average_Weekly_Earnings + ' of')
- 1) AS Average_Weekly_Earnings
FROM parser;
-- Update the "Average_Weekly_Earnings" column
UPDATE parser
SET Average_Weekly_Earnings = SUBSTRING(Average_Weekly_Earnings, 1, CHARINDEX(' of',
Average_Weekly_Earnings + ' of') - 1);

-- Display the updated data
SELECT Average_Weekly_Earnings
FROM parser;
```

--A is Done


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55)) - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Execute
CES
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))
SQLQuery2.sql - W...TATION\giova (54))
--START OF A, Average Weekly Earnings <- Data Type
SELECT *
FROM PARSER
GO
ALTER TABLE parser
DROP COLUMN Average_Weekly_Earnings;
GO
ALTER TABLE parser
ADD Average_Weekly_Earnings VARCHAR (500)
GO
-- Update the "Average_Weekly_Earning" column with the correspon
UPDATE parser
SET Average_Weekly_Earnings = series_title
-- View the updated records
SELECT series_id, Average_Weekly_Earnings
FROM parser
--Substring to the first comma
SELECT SUBSTRING(series_title, 1, CHARINDEX(',', series_title) -
FROM parser;
--update the previous
UPDATE parser
SET Average_Weekly_Earnings = SUBSTRING(series_title, 1, CHARIND
--check
SELECT average_weekly_earnings
FROM parser
--Get rid of everything 'of'-->
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 110
Col 6
Ch 6
INS



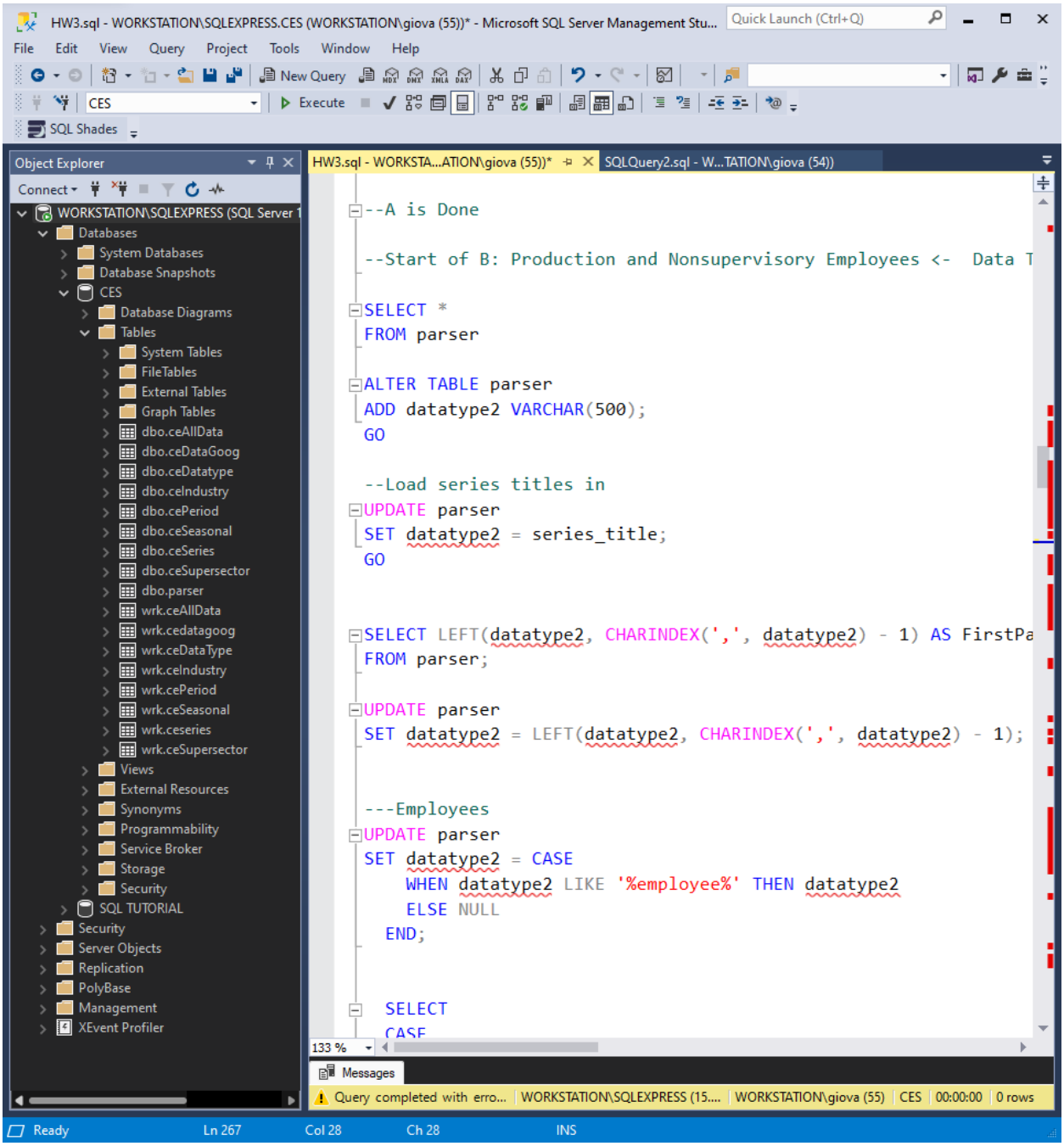
--Start of B: Production and Nonsupervisory Employees <- Data Type

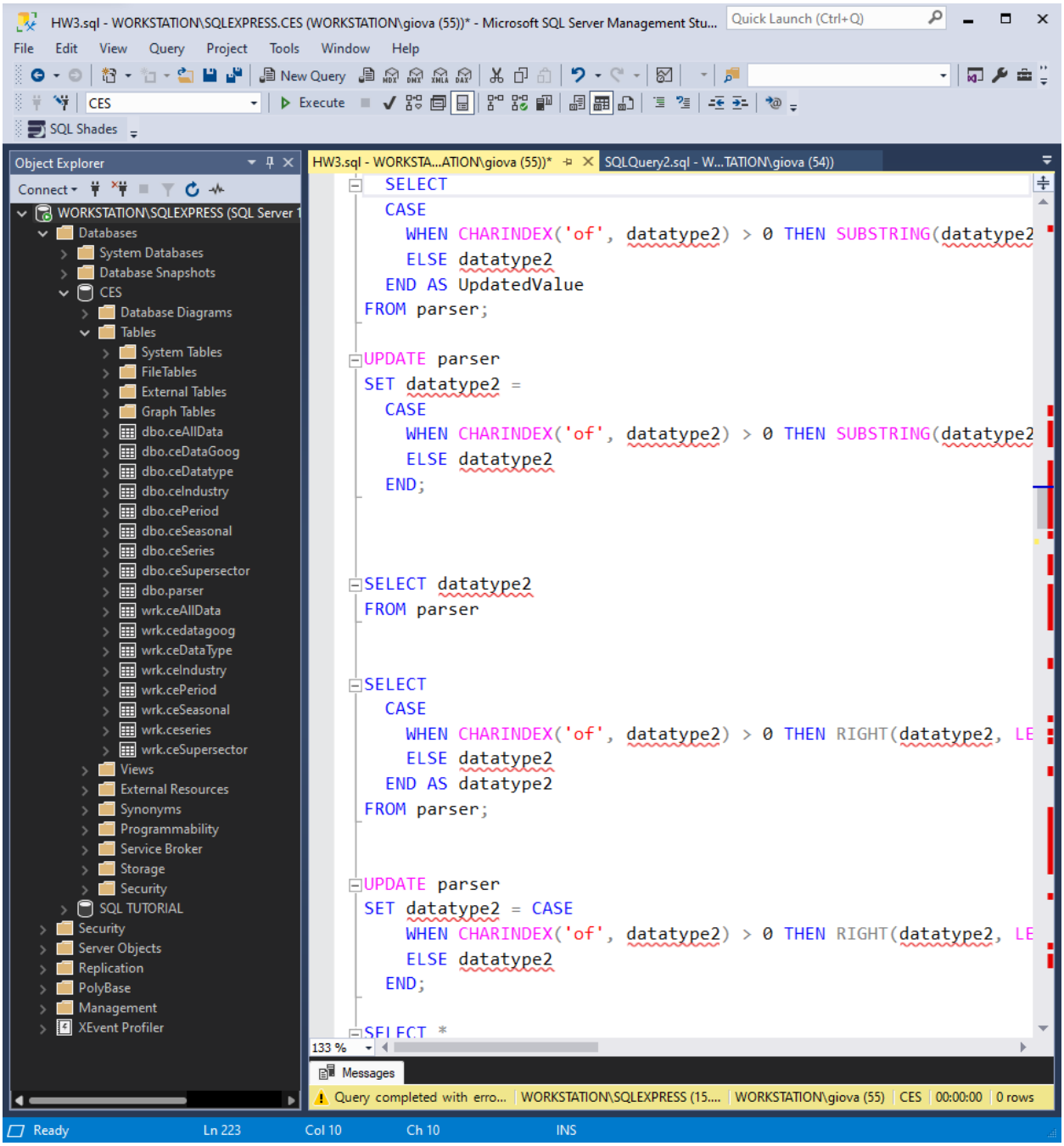
```
SELECT *
FROM parser

ALTER TABLE parser
ADD datatype2 VARCHAR(500);
```

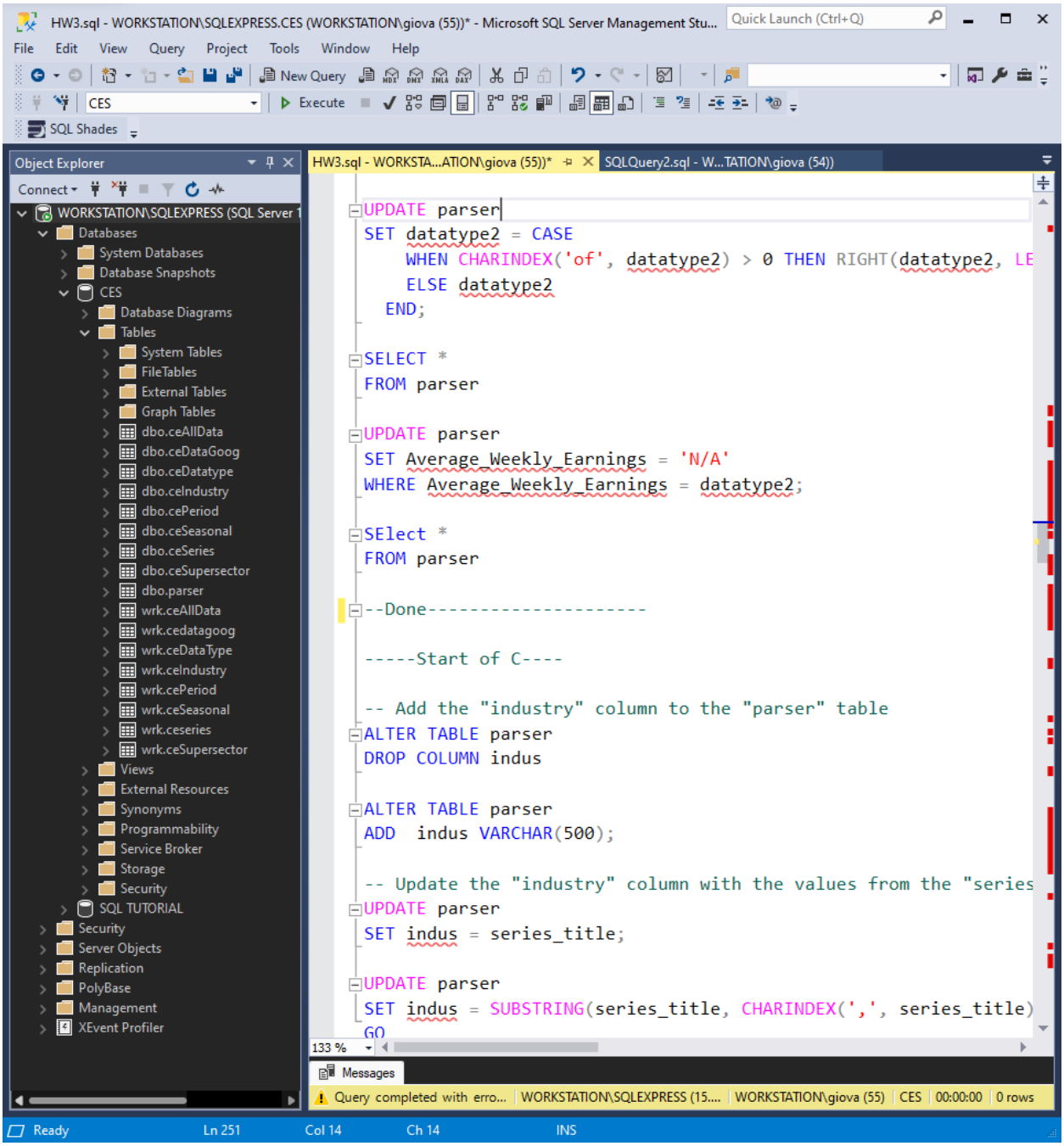
```
GO

--Load series titles in
UPDATE parser
SET datatype2 = series_title;
GO


SELECT LEFT(datatype2, CHARINDEX(',', datatype2) - 1) AS FirstPart
FROM parser;

UPDATE parser
SET datatype2 = LEFT(datatype2, CHARINDEX(',', datatype2) - 1);


---Employees
UPDATE parser
SET datatype2 = CASE
    WHEN datatype2 LIKE '%employee%' THEN datatype2
    ELSE NULL
  END;


  SELECT
  CASE
    WHEN CHARINDEX('of', datatype2) > 0 THEN SUBSTRING(datatype2, CHARINDEX('of',
datatype2) + 3, LEN(datatype2))
    ELSE datatype2
  END AS UpdatedValue
FROM parser;

UPDATE parser
SET datatype2 =
  CASE
    WHEN CHARINDEX('of', datatype2) > 0 THEN SUBSTRING(datatype2, CHARINDEX('of',
datatype2) + 3, LEN(datatype2))
    ELSE datatype2
  END;



SELECT datatype2
FROM parser


SELECT
  CASE
    WHEN CHARINDEX('of', datatype2) > 0 THEN RIGHT(datatype2, LEN(datatype2) -
CHARINDEX('of', datatype2) - 2)
    ELSE datatype2
  END AS datatype2
FROM parser;


UPDATE parser
SET datatype2 = CASE
    WHEN CHARINDEX('of', datatype2) > 0 THEN RIGHT(datatype2, LEN(datatype2) -
CHARINDEX('of', datatype2) - 2)
```

```
    ELSE datatype2
  END;

SELECT *
FROM parser

UPDATE parser
SET Average_Weekly_Earnings = 'N/A'
WHERE Average_Weekly_Earnings = datatype2;

SElect *
FROM parser
```

--Done


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
--A is Done
--Start of B: Production and Nonsupervisory Employees <- Data T
SELECT *
FROM parser
ALTER TABLE parser
ADD datatype2 VARCHAR(500);
GO
--Load series titles in
UPDATE parser
SET datatype2 = series_title;
GO
SELECT LEFT(datatype2, CHARINDEX(',', datatype2) - 1) AS FirstPa
FROM parser;
UPDATE parser
SET datatype2 = LEFT(datatype2, CHARINDEX(',', datatype2) - 1);
---Employees
UPDATE parser
SET datatype2 = CASE
WHEN datatype2 LIKE '%employee%' THEN datatype2
ELSE NULL
END;
SELECT
CASE
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 267
Col 28
Ch 28
INS


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Execute
CES
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
SELECT
CASE
WHEN CHARINDEX('of', datatype2) > 0 THEN SUBSTRING(datatype2
ELSE datatype2
END AS UpdatedValue
FROM parser;
UPDATE parser
SET datatype2 =
CASE
WHEN CHARINDEX('of', datatype2) > 0 THEN SUBSTRING(datatype2
ELSE datatype2
END;
SELECT datatype2
FROM parser
SELECT
CASE
WHEN CHARINDEX('of', datatype2) > 0 THEN RIGHT(datatype2, LE
ELSE datatype2
END AS datatype2
FROM parser;
UPDATE parser
SET datatype2 = CASE
WHEN CHARINDEX('of', datatype2) > 0 THEN RIGHT(datatype2, LE
ELSE datatype2
END;
SELECT *
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 223
Col 10
Ch 10
INS

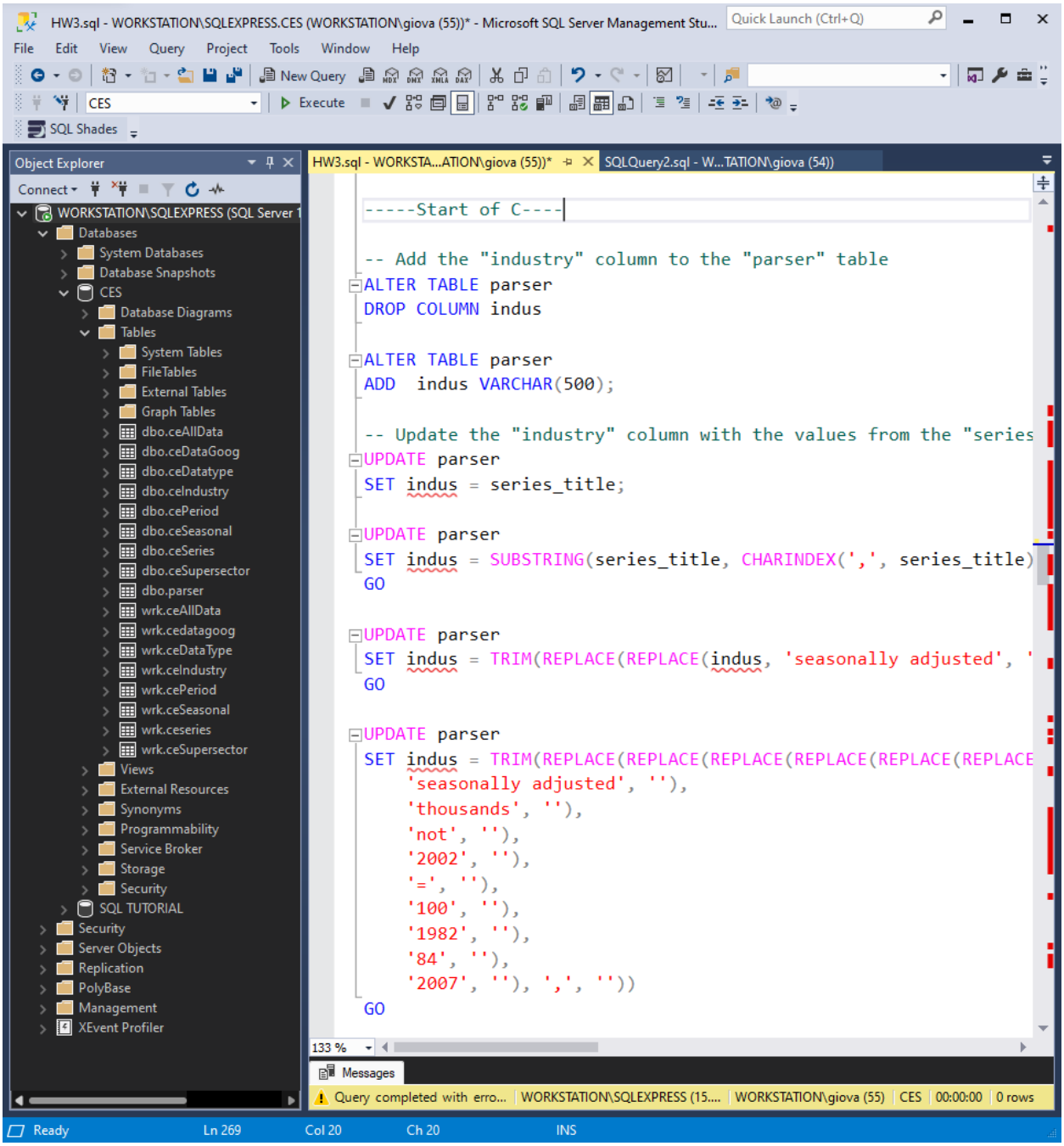

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Execute
CES
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
UPDATE parser
SET datatype2 = CASE
WHEN CHARINDEX('of', datatype2) > 0 THEN RIGHT(datatype2, LE
ELSE datatype2
END;
SELECT *
FROM parser
UPDATE parser
SET Average_Weekly_Earnings = 'N/A'
WHERE Average_Weekly_Earnings = datatype2;
SElect *
FROM parser
--Done----------
-----Start of C----
-- Add the "industry" column to the "parser" table
ALTER TABLE parser
DROP COLUMN indus
ALTER TABLE parser
ADD indus VARCHAR(500);
-- Update the "industry" column with the values from the "series
UPDATE parser
SET indus = series_title;
UPDATE parser
SET indus = SUBSTRING(series_title, CHARINDEX(',', series_title)
GO
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 251
Col 14
Ch 14
INS

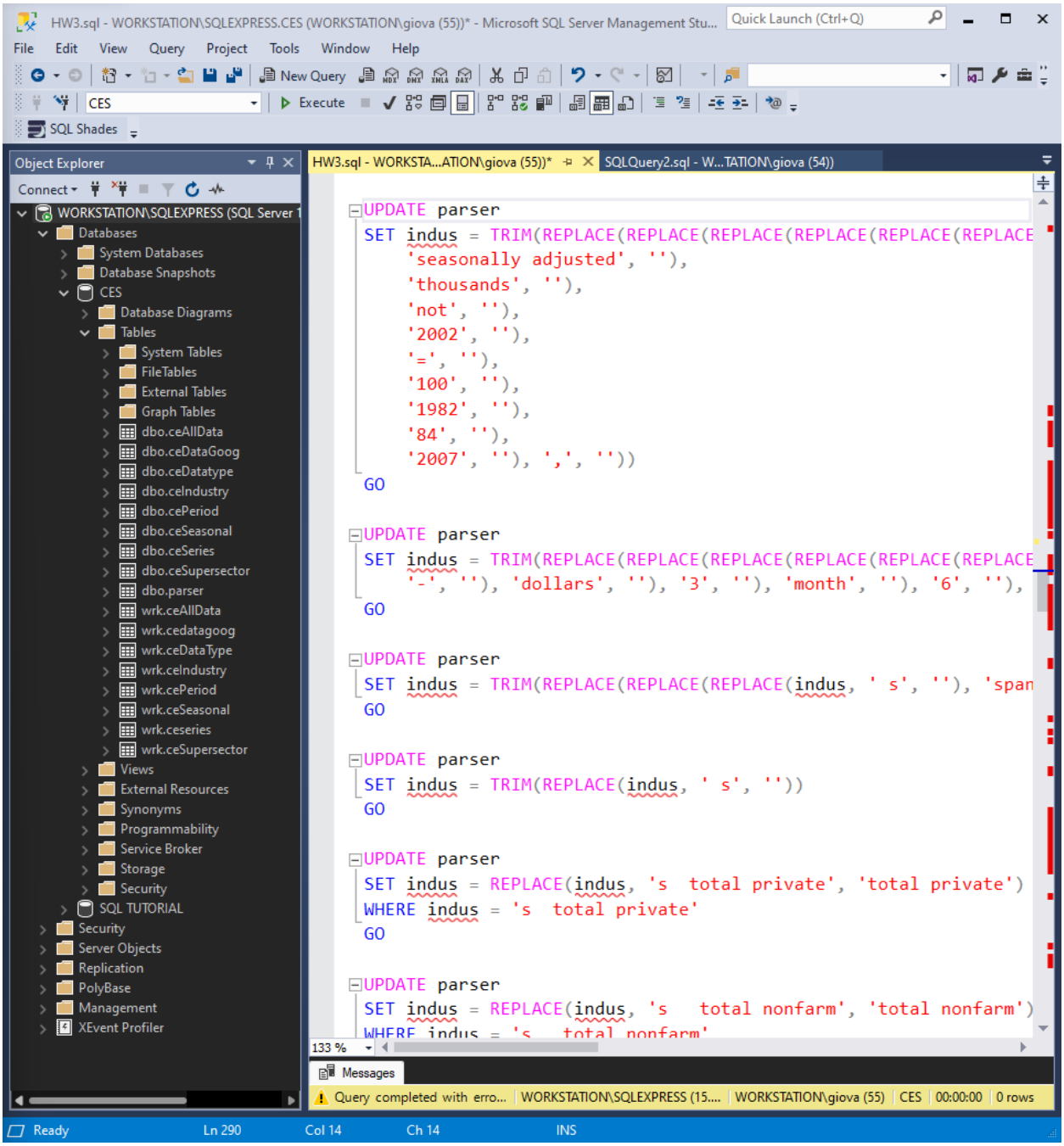
-----Start of C----

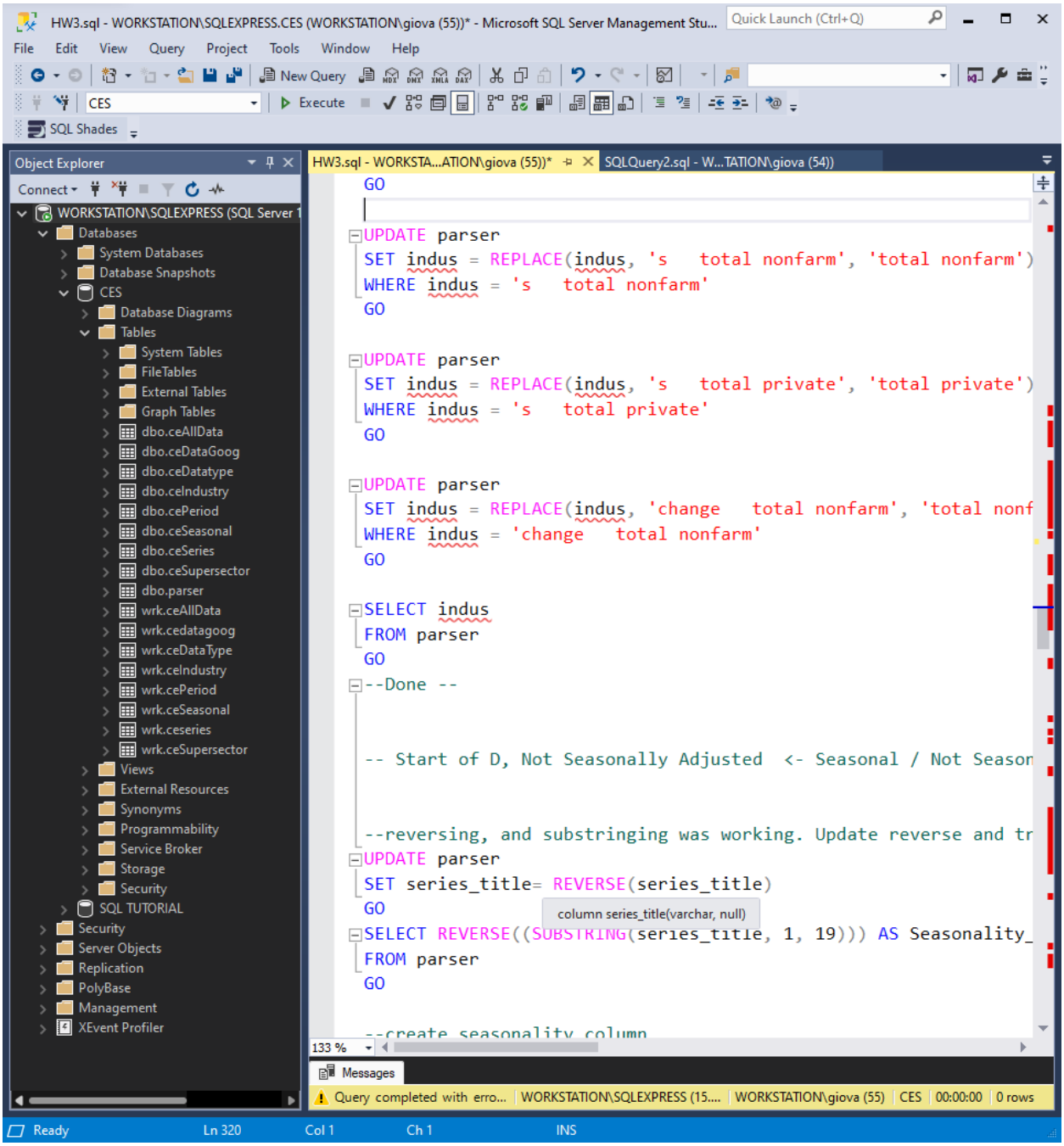
```sql
-- Add the "industry" column to the "parser" table
ALTER TABLE parser
DROP COLUMN indus

ALTER TABLE parser
ADD  indus VARCHAR(500);

-- Update the "industry" column with the values from the "series_title" column
UPDATE parser
SET indus = series_title;

UPDATE parser
SET indus = SUBSTRING(series_title, CHARINDEX(',', series_title) + 1, LEN(series_title));
GO

UPDATE parser
SET indus = TRIM(REPLACE(REPLACE(indus, 'seasonally adjusted', ''), ',', ''))
GO

UPDATE parser
SET indus =
TRIM(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(indu
s,
    'seasonally adjusted', ''),
    'thousands', ''),
    'not', ''),
    '2002', ''),
    '=', ''),
    '100', ''),
    '1982', ''),
    '84', ''),
    '2007', ''), ',', ''))
GO

UPDATE parser
SET indus =
TRIM(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPL
ACE(indus,
    '-', ''), 'dollars', ''), '3', ''), 'month', ''), '6', ''), 'quarterly', ''),
'average', ''), '1', ''), '9', ''), ' ', ' '), ',',''))
GO

UPDATE parser
SET indus = TRIM(REPLACE(REPLACE(REPLACE(indus, ' s', ''), 'span', ''), ',', ''))
GO

UPDATE parser
SET indus = TRIM(REPLACE(indus, ' s', ''))
GO

UPDATE parser
SET indus = REPLACE(indus, 's  total private', 'total private')
WHERE indus = 's  total private'
GO

UPDATE parser
SET indus = REPLACE(indus, 's   total nonfarm', 'total nonfarm')
```

```
WHERE indus = 's   total nonfarm'
GO

UPDATE parser
SET indus = REPLACE(indus, 's   total private', 'total private')
WHERE indus = 's   total private'
GO

UPDATE parser
SET indus = REPLACE(indus, 'change   total nonfarm', 'total nonfarm')
WHERE indus = 'change   total nonfarm'
GO

SELECT indus
FROM parser
GO
```

```
--Done -
```


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
-----Start of C----
-- Add the "industry" column to the "parser" table
ALTER TABLE parser
DROP COLUMN indus
ALTER TABLE parser
ADD indus VARCHAR(500);
-- Update the "industry" column with the values from the "series
UPDATE parser
SET indus = series_title;
UPDATE parser
SET indus = SUBSTRING(series_title, CHARINDEX(',', series_title)
GO
UPDATE parser
SET indus = TRIM(REPLACE(REPLACE(indus, 'seasonally adjusted', '
GO
UPDATE parser
SET indus = TRIM(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE
'seasonally adjusted', ''),
'thousands', ''),
'not', ''),
'2002', ''),
'=', ''),
'100', ''),
'1982', ''),
'84', ''),
'2007', ''), ',', ''))
GO
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 269
Col 20
Ch 20
INS


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
UPDATE parser
SET indus = TRIM(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE
'seasonally adjusted', ''),
'thousands', ''),
'not', ''),
'2002', ''),
'=', ''),
'100', ''),
'1982', ''),
'84', ''),
'2007', ''), ',', ''))
GO
UPDATE parser
SET indus = TRIM(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE
'-', ''), 'dollars', ''), '3', ''), 'month', ''), '6', ''),
GO
UPDATE parser
SET indus = TRIM(REPLACE(REPLACE(REPLACE(indus, ' s', ''), 'span
GO
UPDATE parser
SET indus = TRIM(REPLACE(indus, ' s', ''))
GO
UPDATE parser
SET indus = REPLACE(indus, 's total private', 'total private')
WHERE indus = 's total private'
GO
UPDATE parser
SET indus = REPLACE(indus, 's total nonfarm', 'total nonfarm')
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 290
Col 14
Ch 14
INS

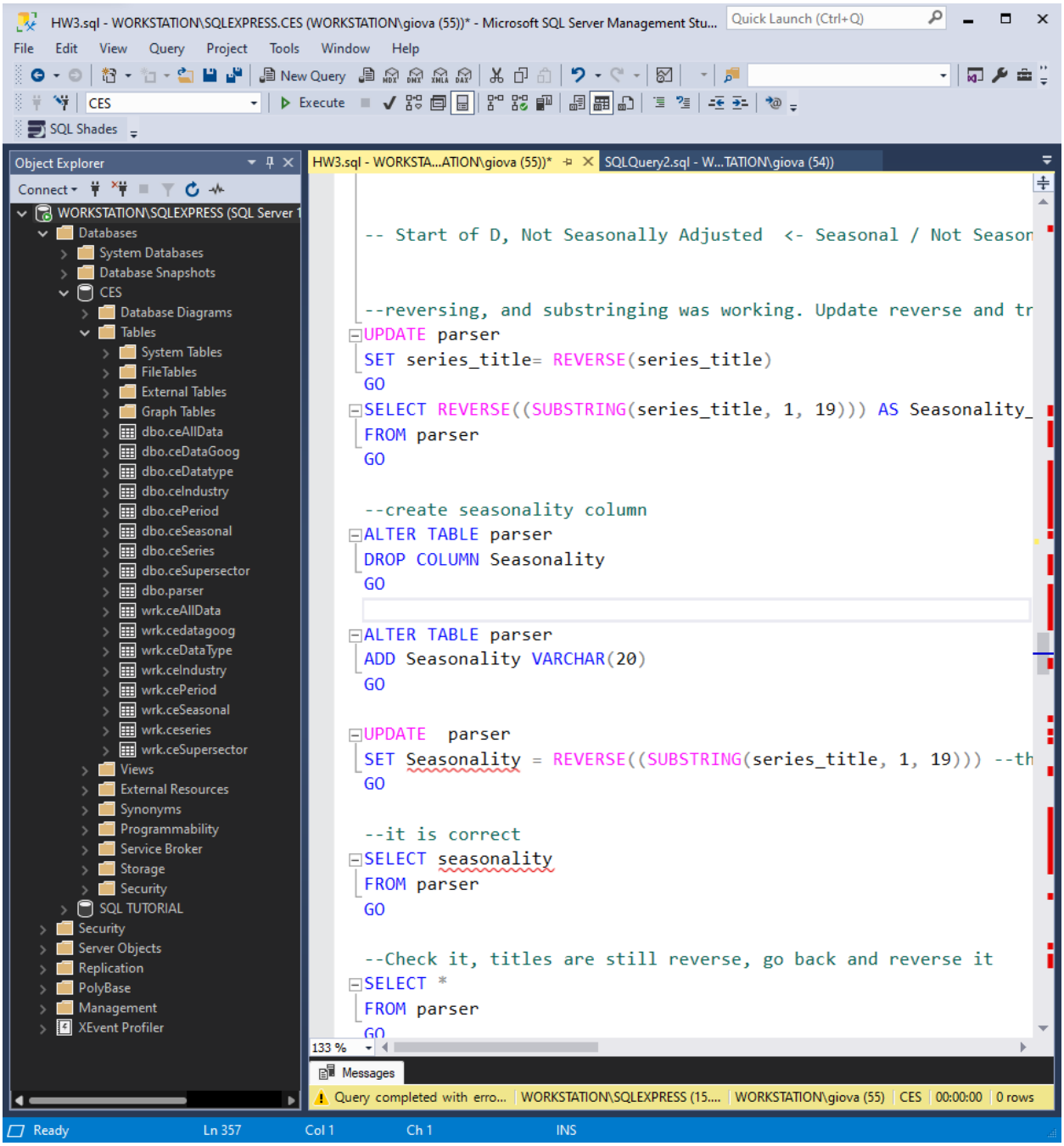


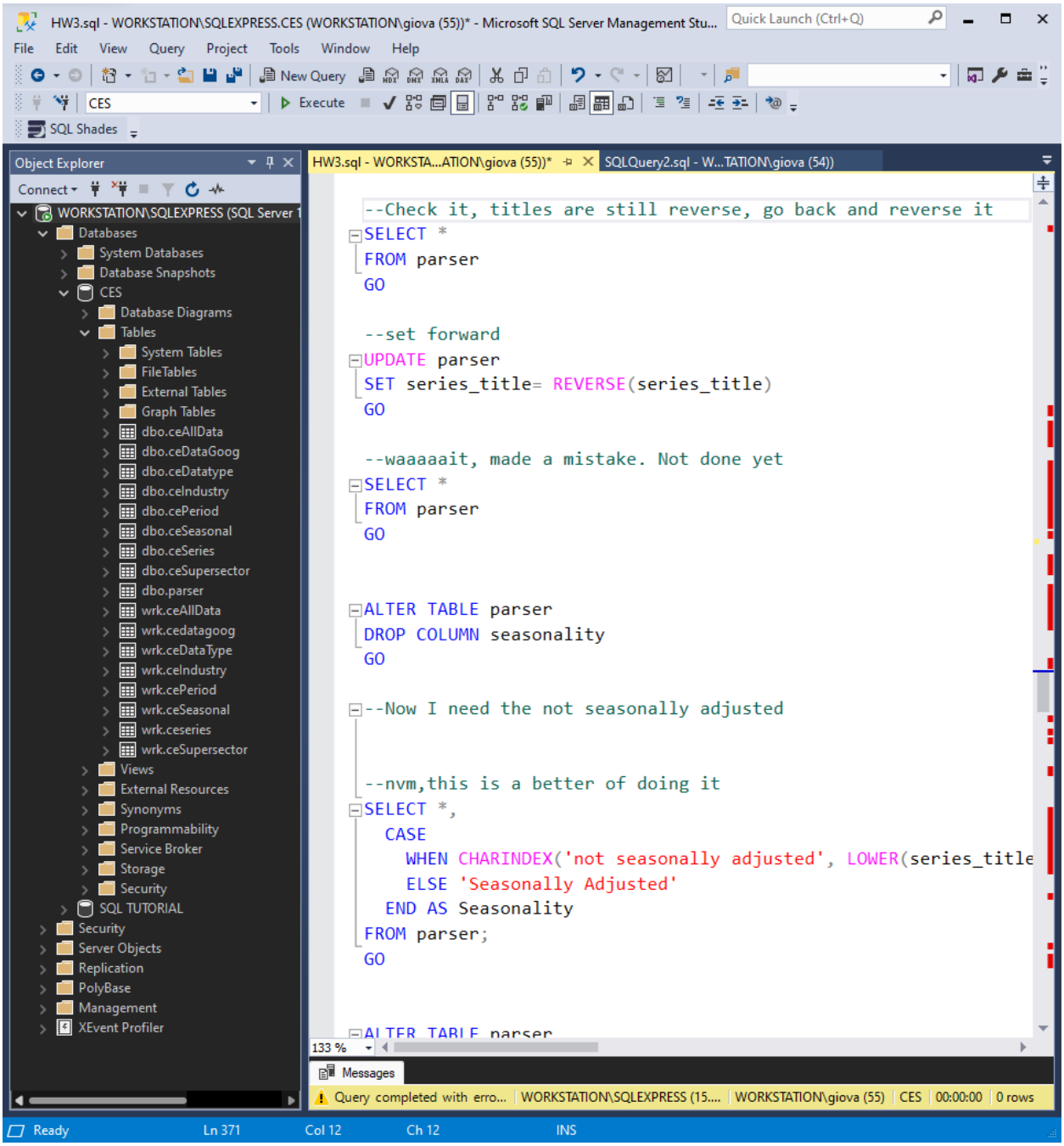

```
-- Start of D, Not Seasonally Adjusted  <- Seasonal / Not Seasonal---

--reversing, and substringing was working. Update reverse and try again
UPDATE parser
```

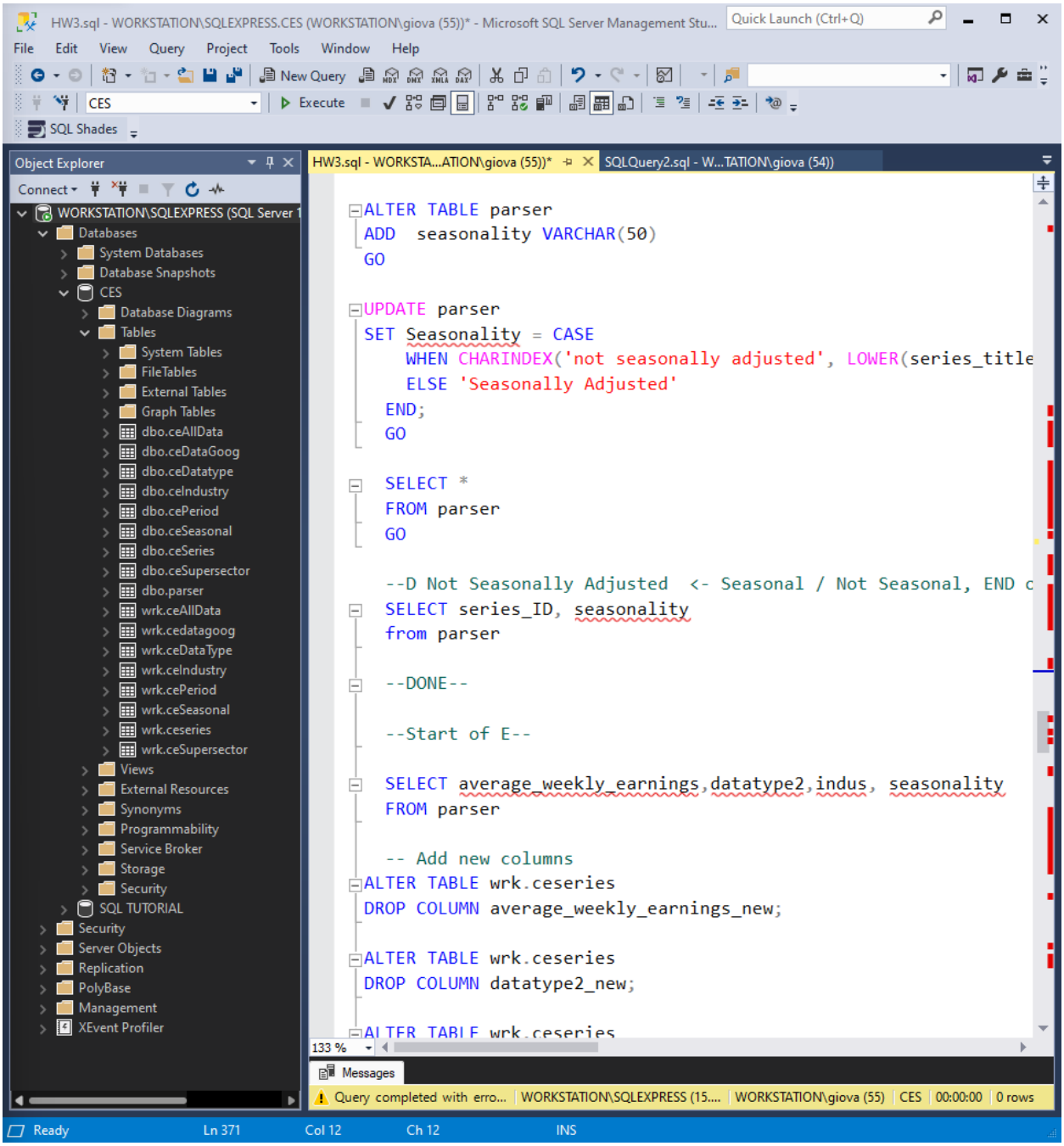
```
SET series_title= REVERSE(series_title)
GO
SELECT REVERSE((SUBSTRING(series_title, 1, 19))) AS Seasonality_forward
FROM parser
GO

--create seasonality column
ALTER TABLE parser
DROP COLUMN Seasonality
GO

ALTER TABLE parser
ADD Seasonality VARCHAR(20)
GO

UPDATE  parser
SET Seasonality = REVERSE((SUBSTRING(series_title, 1, 19))) --this is seasonality forward
GO

--it is correct
SELECT seasonality
FROM parser
GO

--Check it, titles are still reverse, go back and reverse it
SELECT *
FROM parser
GO

--set forward
UPDATE parser
SET series_title= REVERSE(series_title)
GO

--waaaaait, made a mistake. Not done yet
SELECT *
FROM parser
GO


ALTER TABLE parser
DROP COLUMN seasonality
GO

--Now I need the not seasonally adjusted


--nvm,this is a better of doing it
SELECT *,
  CASE
    WHEN CHARINDEX('not seasonally adjusted', LOWER(series_title)) > 0 THEN 'Not
Seasonally Adjusted'
    ELSE 'Seasonally Adjusted'
  END AS Seasonality
FROM parser;
GO
```

```sql
ALTER TABLE parser
ADD  seasonality VARCHAR(50)
GO

UPDATE parser
SET Seasonality = CASE
    WHEN CHARINDEX('not seasonally adjusted', LOWER(series_title)) > 0 THEN 'Not
Seasonally Adjusted'
    ELSE 'Seasonally Adjusted'
  END;
  GO

  SELECT *
  FROM parser
  GO

  --D Not Seasonally Adjusted  <- Seasonal / Not Seasonal, END of D
  SELECT series_ID, seasonality
  from parser
```

--DONE—


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
-- Start of D, Not Seasonally Adjusted <- Seasonal / Not Seasor
--reversing, and substringing was working. Update reverse and tr
UPDATE parser
SET series_title= REVERSE(series_title)
GO
SELECT REVERSE((SUBSTRING(series_title, 1, 19))) AS Seasonality_
FROM parser
GO
--create seasonality column
ALTER TABLE parser
DROP COLUMN Seasonality
GO
ALTER TABLE parser
ADD Seasonality VARCHAR(20)
GO
UPDATE parser
SET Seasonality = REVERSE((SUBSTRING(series_title, 1, 19))) --th
GO
--it is correct
SELECT seasonality
FROM parser
GO
--Check it, titles are still reverse, go back and reverse it
SELECT *
FROM parser
GO
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 357
Col 1
Ch 1
INS

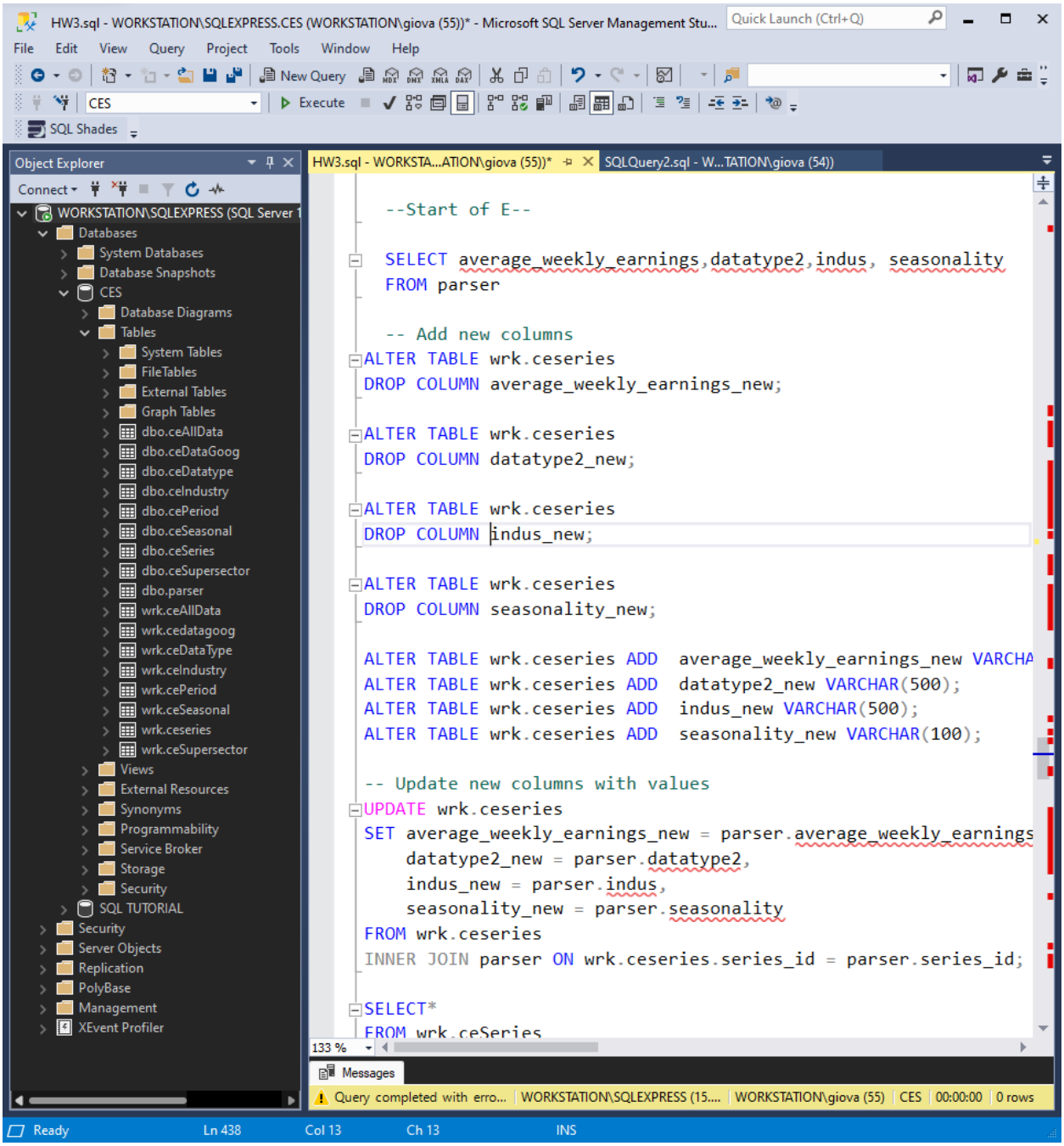

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
--Check it, titles are still reverse, go back and reverse it
SELECT *
FROM parser
GO
--set forward
UPDATE parser
SET series_title= REVERSE(series_title)
GO
--waaaaait, made a mistake. Not done yet
SELECT *
FROM parser
GO
ALTER TABLE parser
DROP COLUMN seasonality
GO
--Now I need the not seasonally adjusted
--nvm,this is a better of doing it
SELECT *,
CASE
WHEN CHARINDEX('not seasonally adjusted', LOWER(series_title
ELSE 'Seasonally Adjusted'
END AS Seasonality
FROM parser;
GO
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 371
Col 12
Ch 12
INS

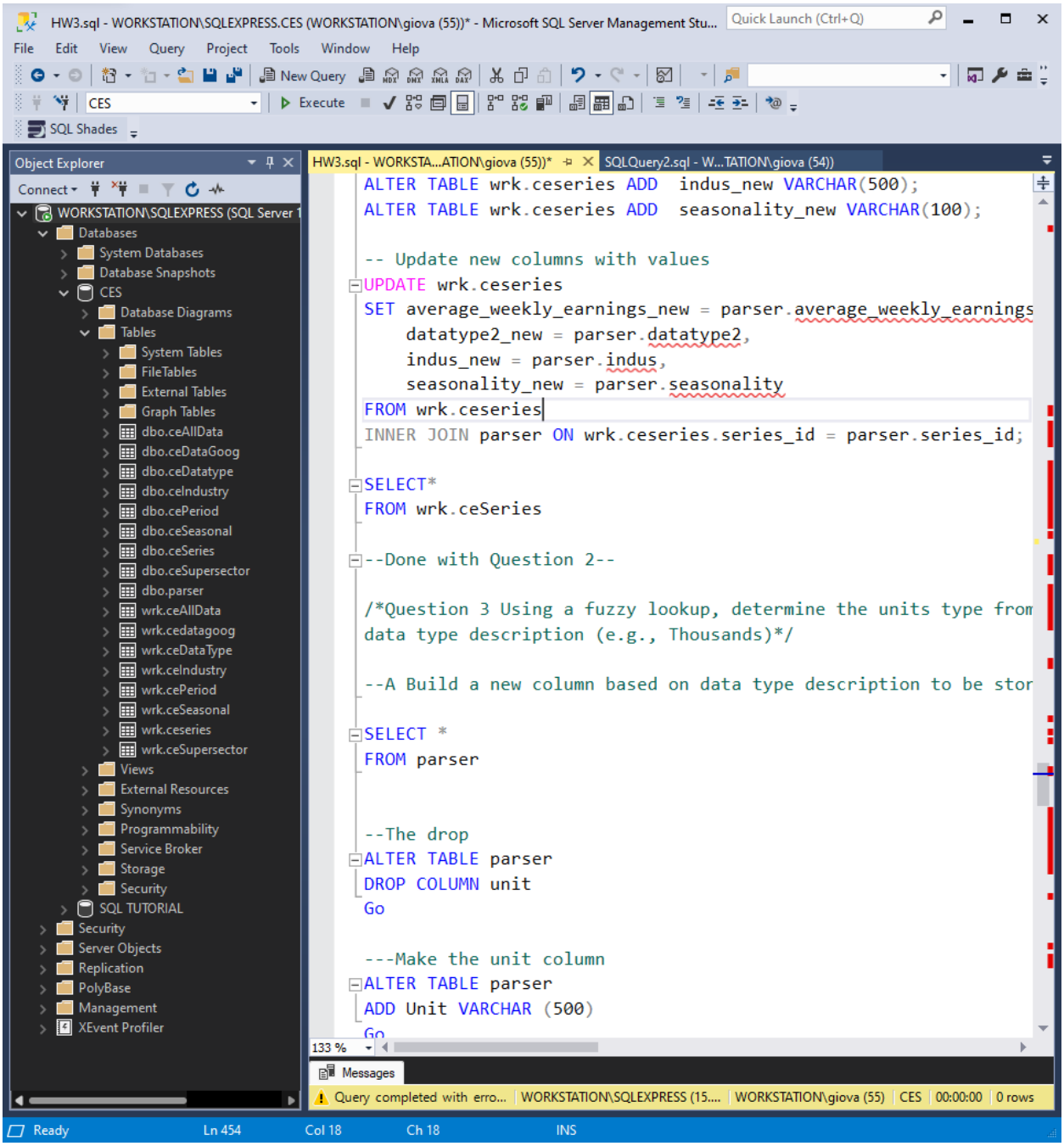

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
ALTER TABLE parser
ADD seasonality VARCHAR(50)
GO
UPDATE parser
SET Seasonality = CASE
WHEN CHARINDEX('not seasonally adjusted', LOWER(series_title
ELSE 'Seasonally Adjusted'
END;
GO
SELECT *
FROM parser
GO
--D Not Seasonally Adjusted <- Seasonal / Not Seasonal, END c
SELECT series_ID, seasonality
from parser
--DONE--
--Start of E--
SELECT average_weekly_earnings,datatype2,indus, seasonality
FROM parser
-- Add new columns
ALTER TABLE wrk.ceseries
DROP COLUMN average_weekly_earnings_new;
ALTER TABLE wrk.ceseries
DROP COLUMN datatype2_new;
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 371
Col 12
Ch 12
INS

```
--Start of E--
```

```
  SELECT average_weekly_earnings,datatype2,indus, seasonality
  FROM parser

  -- Add new columns
ALTER TABLE wrk.ceseries
DROP COLUMN average_weekly_earnings_new;

ALTER TABLE wrk.ceseries
DROP COLUMN datatype2_new;

ALTER TABLE wrk.ceseries
DROP COLUMN indus_new;

ALTER TABLE wrk.ceseries
DROP COLUMN seasonality_new;

ALTER TABLE wrk.ceseries ADD  average_weekly_earnings_new VARCHAR(100);
ALTER TABLE wrk.ceseries ADD  datatype2_new VARCHAR(500);
ALTER TABLE wrk.ceseries ADD  indus_new VARCHAR(500);
ALTER TABLE wrk.ceseries ADD  seasonality_new VARCHAR(100);

-- Update new columns with values
UPDATE wrk.ceseries
SET average_weekly_earnings_new = parser.average_weekly_earnings,
    datatype2_new = parser.datatype2,
    indus_new = parser.indus,
    seasonality_new = parser.seasonality
FROM wrk.ceseries
INNER JOIN parser ON wrk.ceseries.series_id = parser.series_id;

SELECT*
FROM wrk.ceSeries
```

--Done with Question 2—


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Execute
CES
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
--Start of E--
SELECT average_weekly_earnings,datatype2,indus, seasonality
FROM parser
-- Add new columns
ALTER TABLE wrk.ceseries
DROP COLUMN average_weekly_earnings_new;
ALTER TABLE wrk.ceseries
DROP COLUMN datatype2_new;
ALTER TABLE wrk.ceseries
DROP COLUMN indus_new;
ALTER TABLE wrk.ceseries
DROP COLUMN seasonality_new;
ALTER TABLE wrk.ceseries ADD average_weekly_earnings_new VARCHA
ALTER TABLE wrk.ceseries ADD datatype2_new VARCHAR(500);
ALTER TABLE wrk.ceseries ADD indus_new VARCHAR(500);
ALTER TABLE wrk.ceseries ADD seasonality_new VARCHAR(100);
-- Update new columns with values
UPDATE wrk.ceseries
SET average_weekly_earnings_new = parser.average_weekly_earnings
datatype2_new = parser.datatype2,
indus_new = parser.indus,
seasonality_new = parser.seasonality
FROM wrk.ceseries
INNER JOIN parser ON wrk.ceseries.series_id = parser.series_id;
SELECT*
FROM wrk.ceSeries
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 438
Col 13
Ch 13
INS

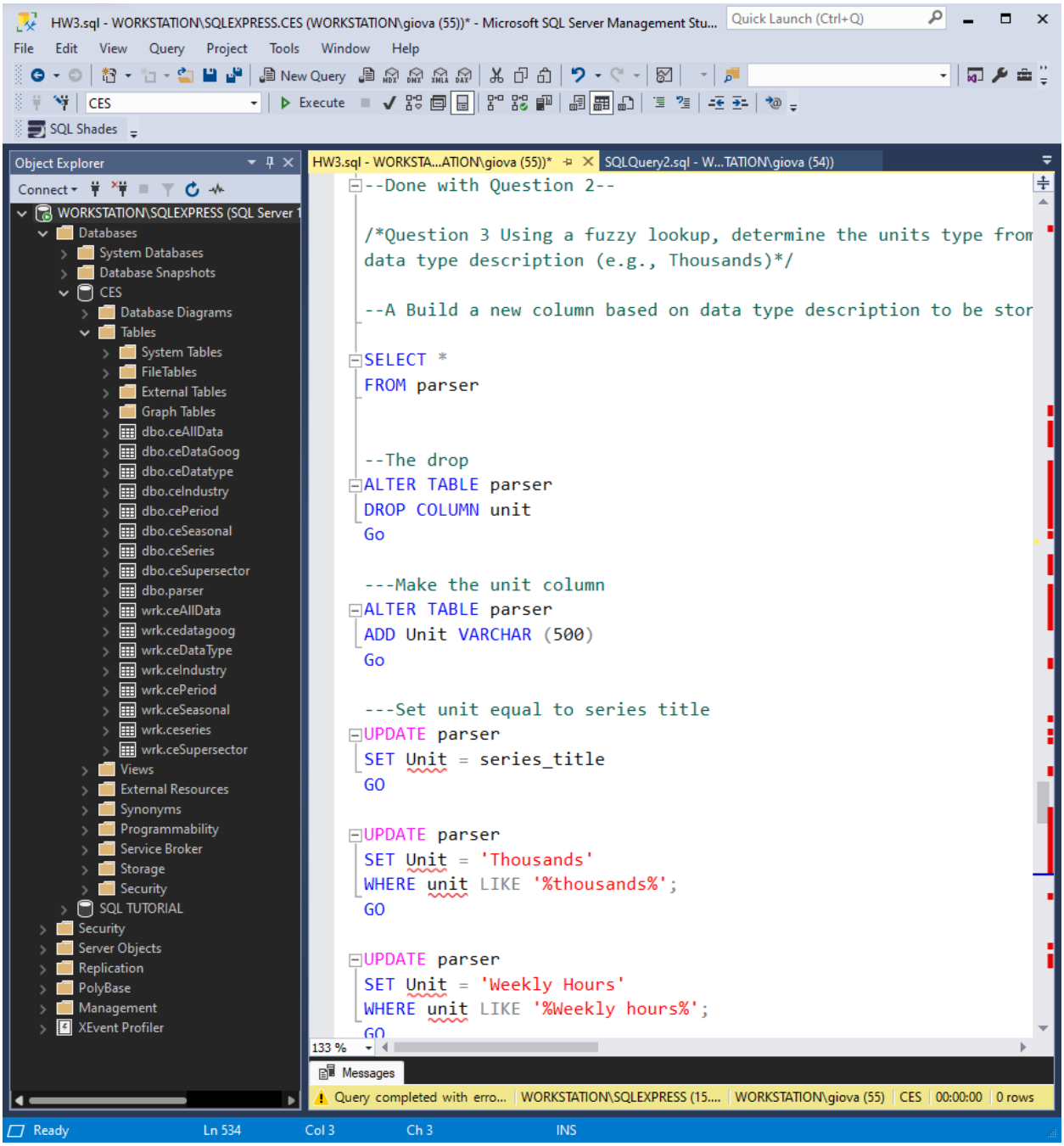


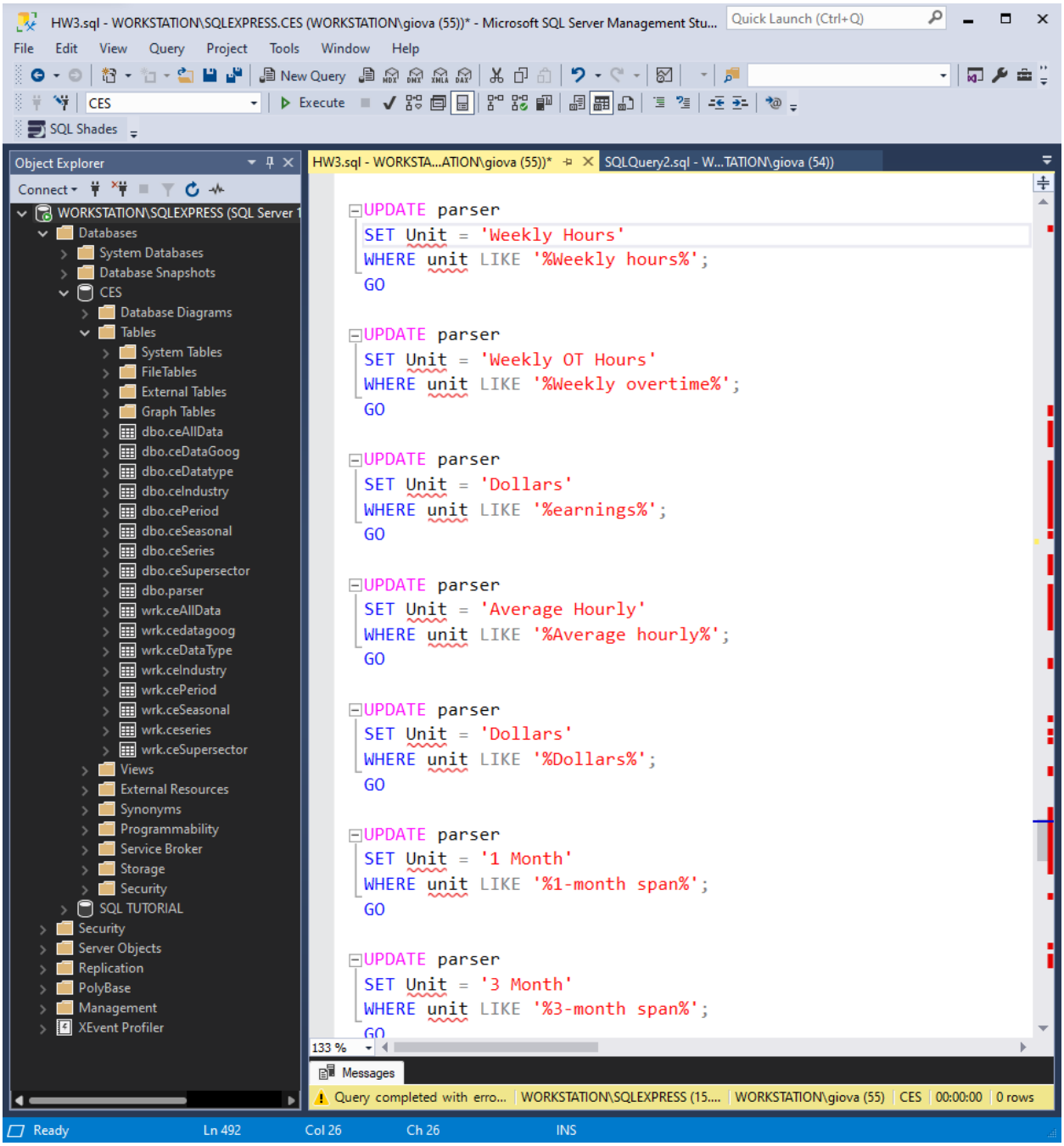

```
/*Question 3 Using a fuzzy lookup,
determine the units type from the
data type description (e.g., Thousands)*/

--A Build a new column based on data type
description to be stored in their true units
```

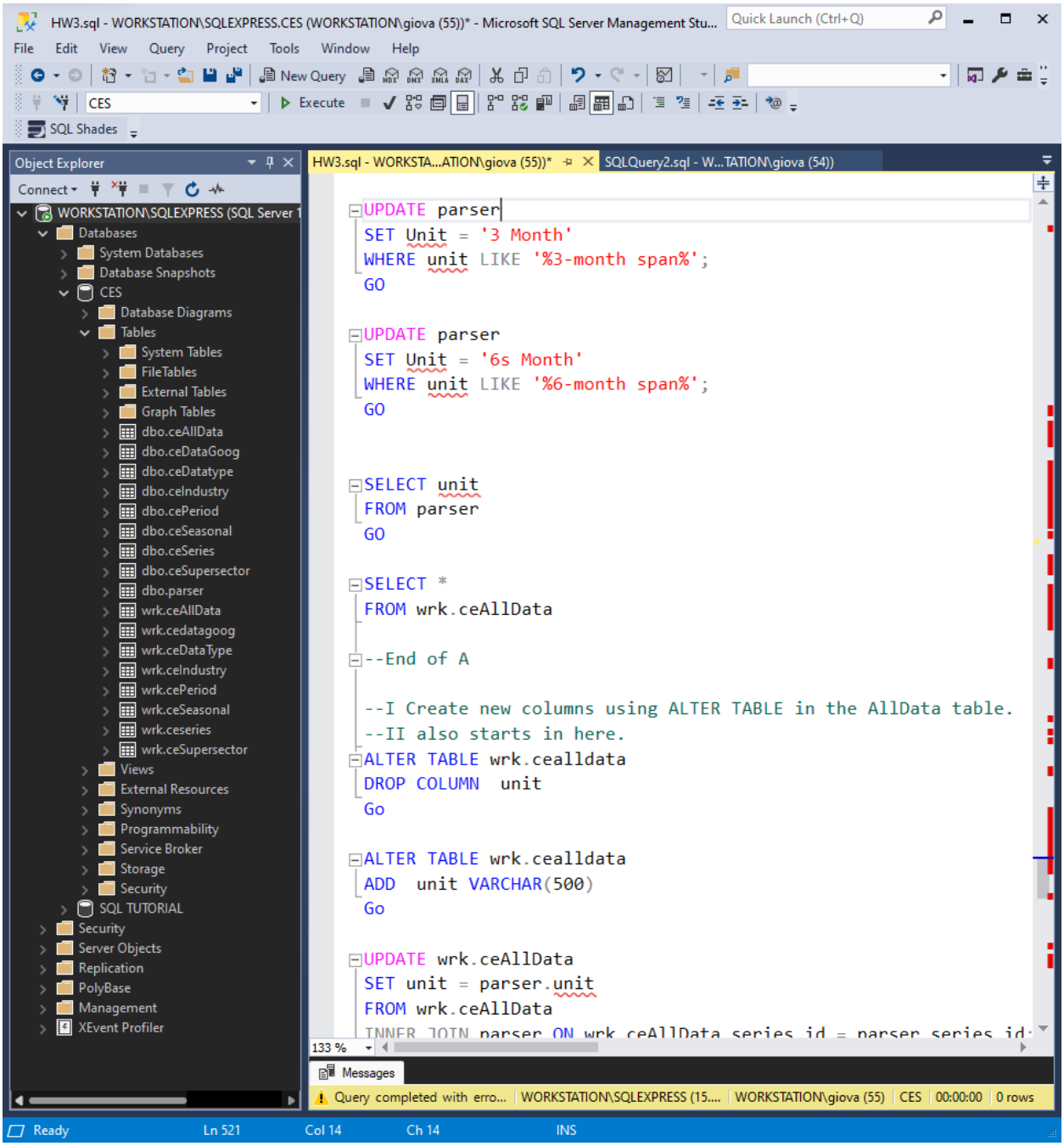
```sql
SELECT *
FROM parser


--The drop
ALTER TABLE parser
DROP COLUMN unit
Go

---Make the unit column
ALTER TABLE parser
ADD Unit VARCHAR (500)
Go

---Set unit equal to series title
UPDATE parser
SET Unit = series_title
GO

UPDATE parser
SET Unit = 'Thousands'
WHERE unit LIKE '%thousands%';
GO

UPDATE parser
SET Unit = 'Weekly Hours'
WHERE unit LIKE '%Weekly hours%';
GO

UPDATE parser
SET Unit = 'Weekly OT Hours'
WHERE unit LIKE '%Weekly overtime%';
GO

UPDATE parser
SET Unit = 'Dollars'
WHERE unit LIKE '%earnings%';
GO

UPDATE parser
SET Unit = 'Average Hourly'
WHERE unit LIKE '%Average hourly%';
GO

UPDATE parser
SET Unit = 'Dollars'
WHERE unit LIKE '%Dollars%';
GO

UPDATE parser
SET Unit = '1 Month'
WHERE unit LIKE '%1-month span%';
GO

UPDATE parser
SET Unit = '3 Month'
WHERE unit LIKE '%3-month span%';
GO
```

```
UPDATE parser
SET Unit = '6s Month'
WHERE unit LIKE '%6-month span%';
GO


SELECT unit
FROM parser
GO

SELECT *
FROM wrk.ceAllData
```

--End of A

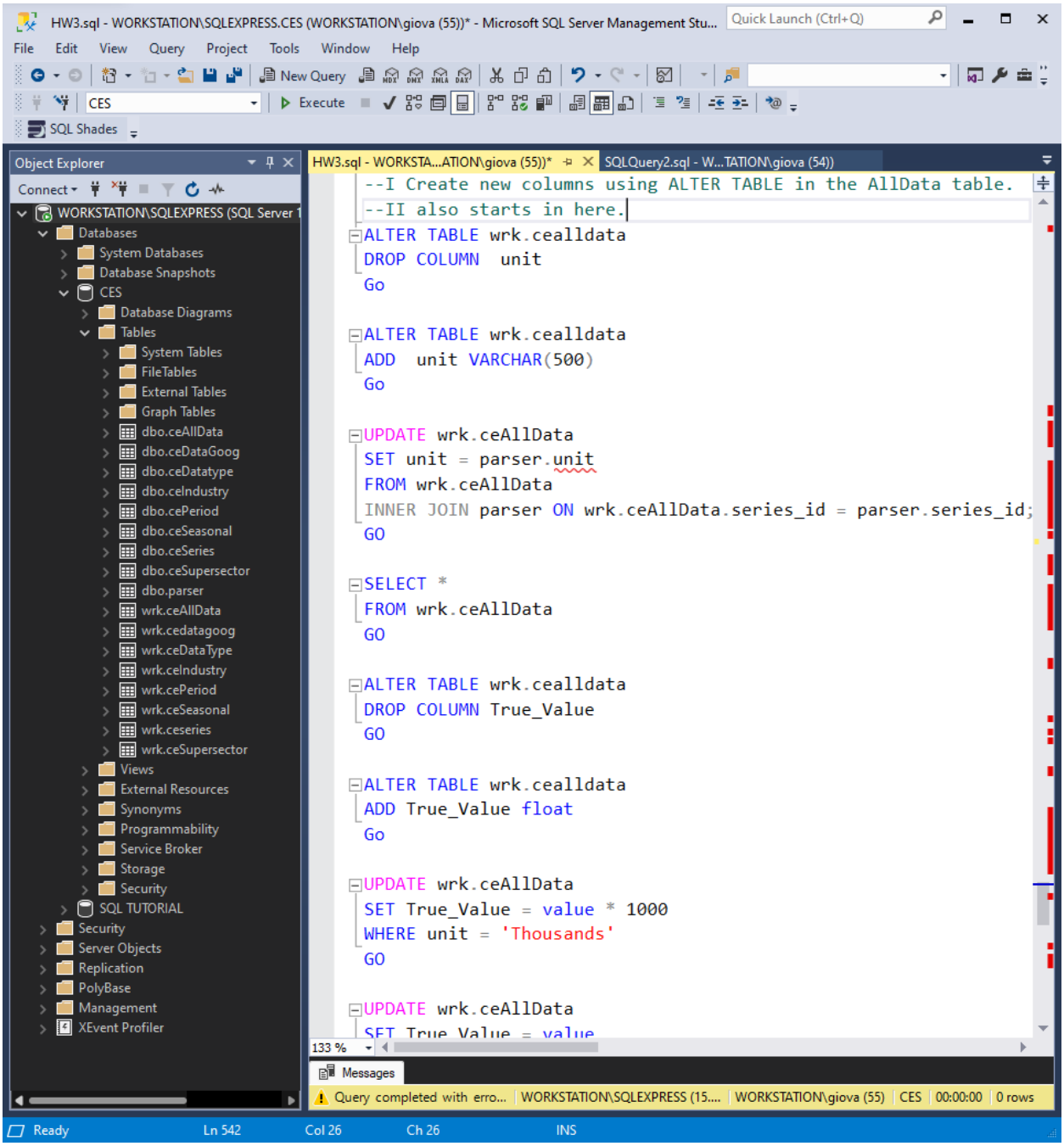

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
--Done with Question 2--
/*Question 3 Using a fuzzy lookup, determine the units type from
data type description (e.g., Thousands)*/
--A Build a new column based on data type description to be stor
SELECT *
FROM parser
--The drop
ALTER TABLE parser
DROP COLUMN unit
Go
---Make the unit column
ALTER TABLE parser
ADD Unit VARCHAR (500)
Go
---Set unit equal to series title
UPDATE parser
SET Unit = series_title
GO
UPDATE parser
SET Unit = 'Thousands'
WHERE unit LIKE '%thousands%';
GO
UPDATE parser
SET Unit = 'Weekly Hours'
WHERE unit LIKE '%Weekly hours%';
GO
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 534
Col 3
Ch 3
INS

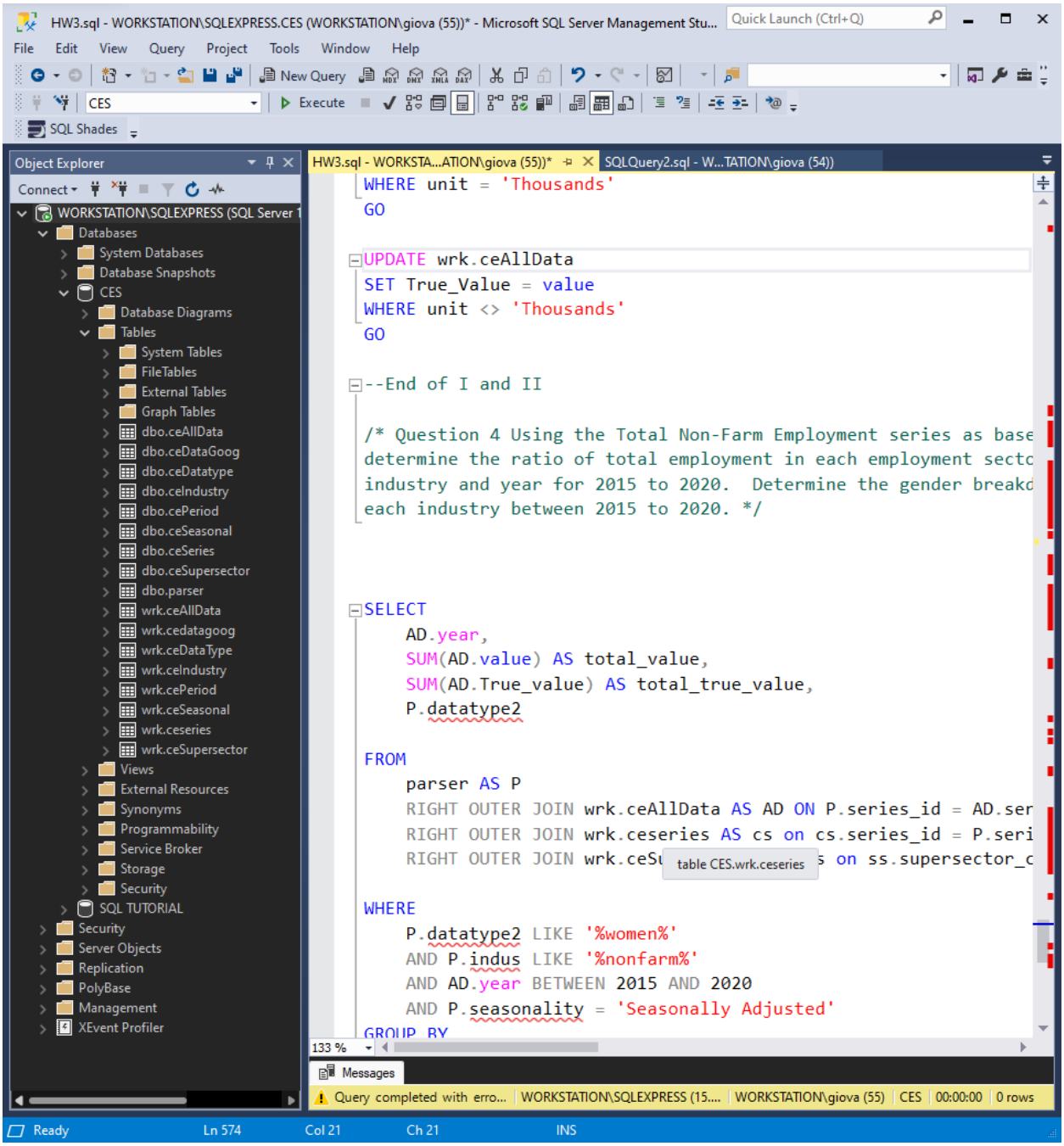

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
UPDATE parser
SET Unit = 'Weekly Hours'
WHERE unit LIKE '%Weekly hours%';
GO
UPDATE parser
SET Unit = 'Weekly OT Hours'
WHERE unit LIKE '%Weekly overtime%';
GO
UPDATE parser
SET Unit = 'Dollars'
WHERE unit LIKE '%earnings%';
GO
UPDATE parser
SET Unit = 'Average Hourly'
WHERE unit LIKE '%Average hourly%';
GO
UPDATE parser
SET Unit = 'Dollars'
WHERE unit LIKE '%Dollars%';
GO
UPDATE parser
SET Unit = '1 Month'
WHERE unit LIKE '%1-month span%';
GO
UPDATE parser
SET Unit = '3 Month'
WHERE unit LIKE '%3-month span%';
GO
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 492
Col 26
Ch 26
INS




```
--I Create new columns using ALTER TABLE
in the AllData table.
--II also starts in here.
ALTER TABLE wrk.cealldata
DROP COLUMN  unit
```

```
Go

ALTER TABLE wrk.cealldata
ADD  unit VARCHAR(500)
Go

UPDATE wrk.ceAllData
SET unit = parser.unit
FROM wrk.ceAllData
INNER JOIN parser ON wrk.ceAllData.series_id = parser.series_id;
GO

SELECT *
FROM wrk.ceAllData
GO

ALTER TABLE wrk.cealldata
DROP COLUMN True_Value
GO

ALTER TABLE wrk.cealldata
ADD True_Value float
Go

UPDATE wrk.ceAllData
SET True_Value = value * 1000
WHERE unit = 'Thousands'
GO

UPDATE wrk.ceAllData
SET True_Value = value
WHERE unit <> 'Thousands'
GO

--End of I and II
```

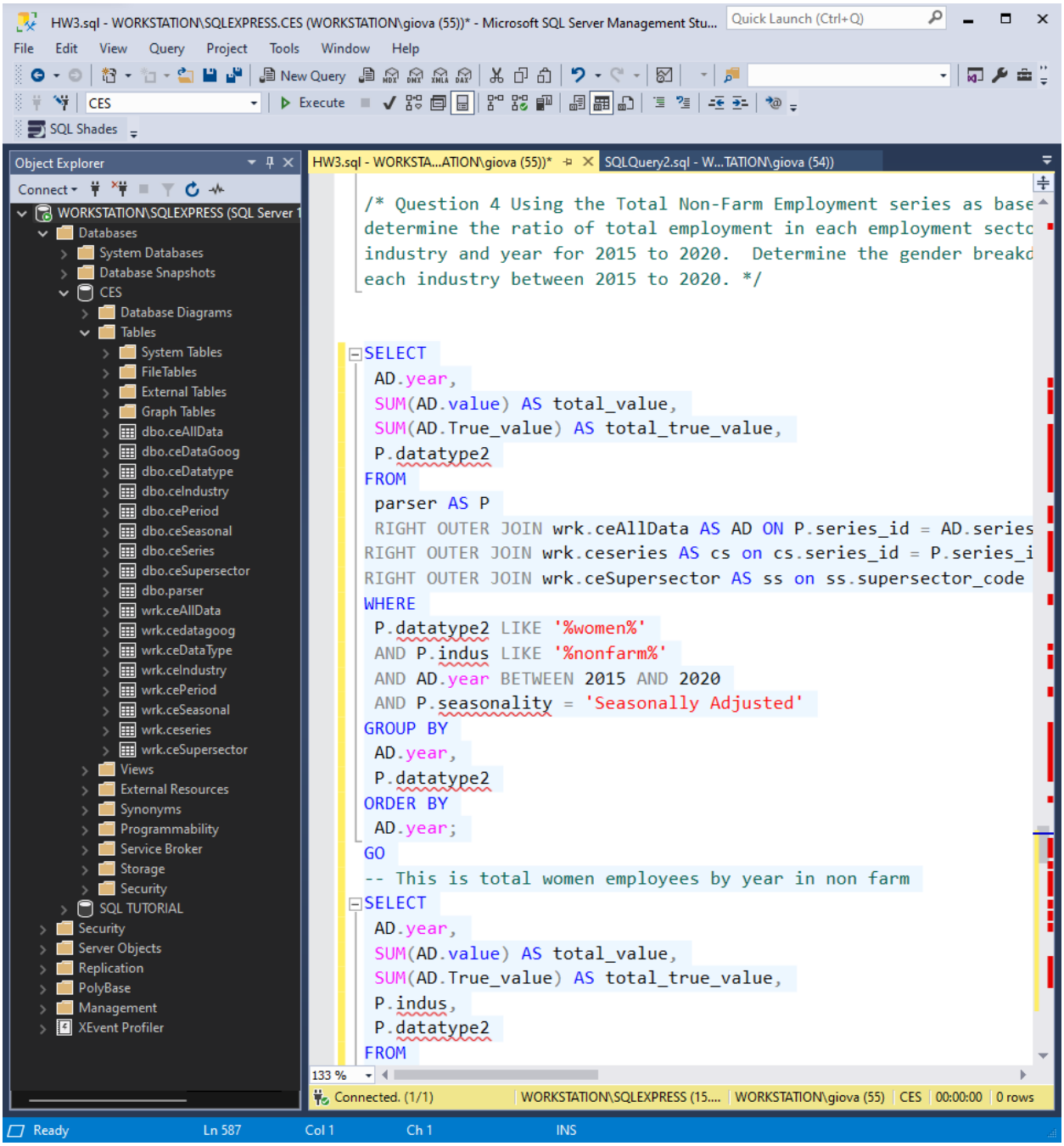

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
--I Create new columns using ALTER TABLE in the AllData table.
--II also starts in here.
ALTER TABLE wrk.cealldata
DROP COLUMN unit
Go
ALTER TABLE wrk.cealldata
ADD unit VARCHAR(500)
Go
UPDATE wrk.ceAllData
SET unit = parser.unit
FROM wrk.ceAllData
INNER JOIN parser ON wrk.ceAllData.series_id = parser.series_id;
GO
SELECT *
FROM wrk.ceAllData
GO
ALTER TABLE wrk.cealldata
DROP COLUMN True_Value
GO
ALTER TABLE wrk.cealldata
ADD True_Value float
Go
UPDATE wrk.ceAllData
SET True_Value = value * 1000
WHERE unit = 'Thousands'
GO
UPDATE wrk.ceAllData
SET True_Value = value
133 %
Messages
Query completed with erro...
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 542
Col 26
Ch 26
INS

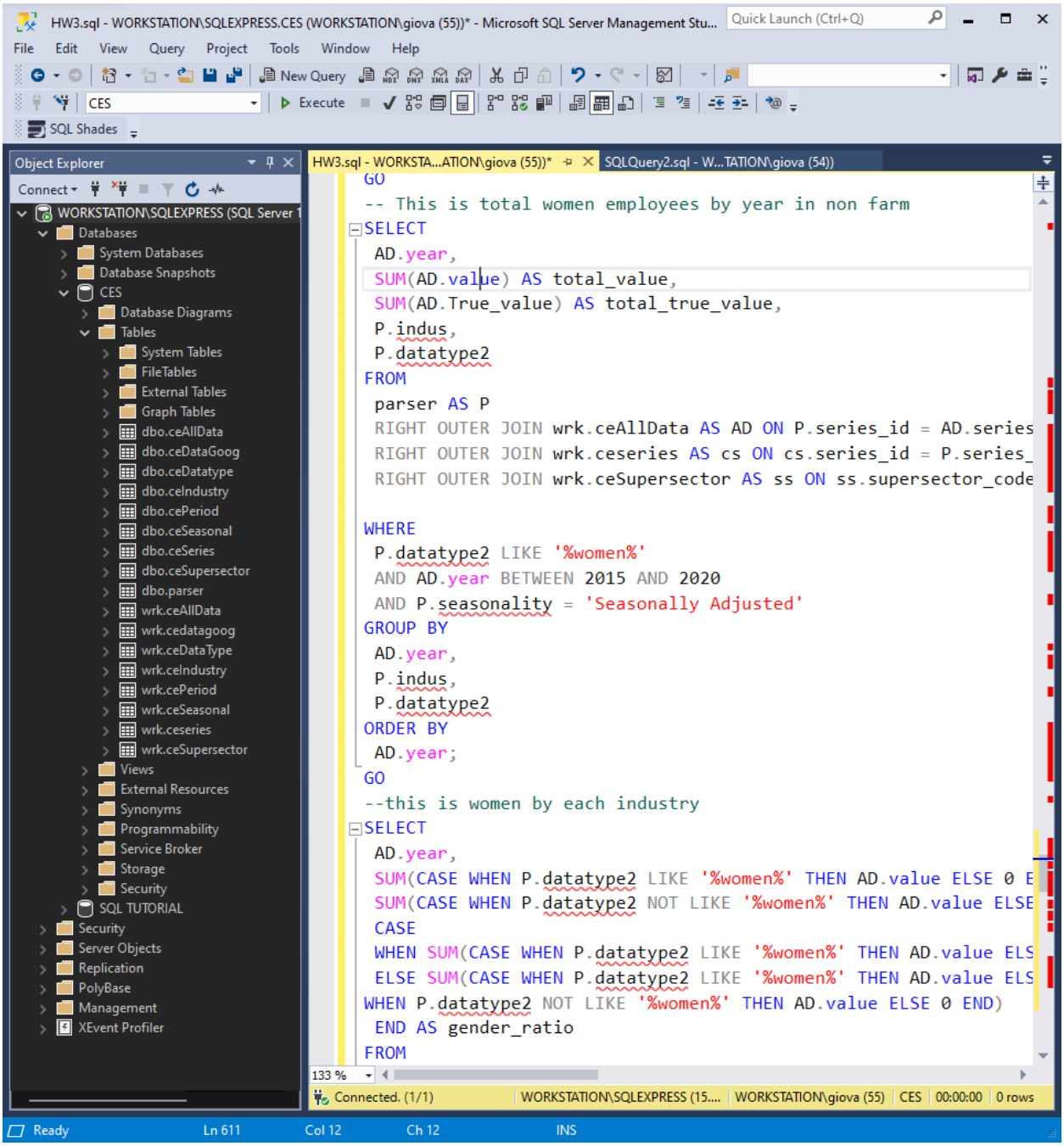


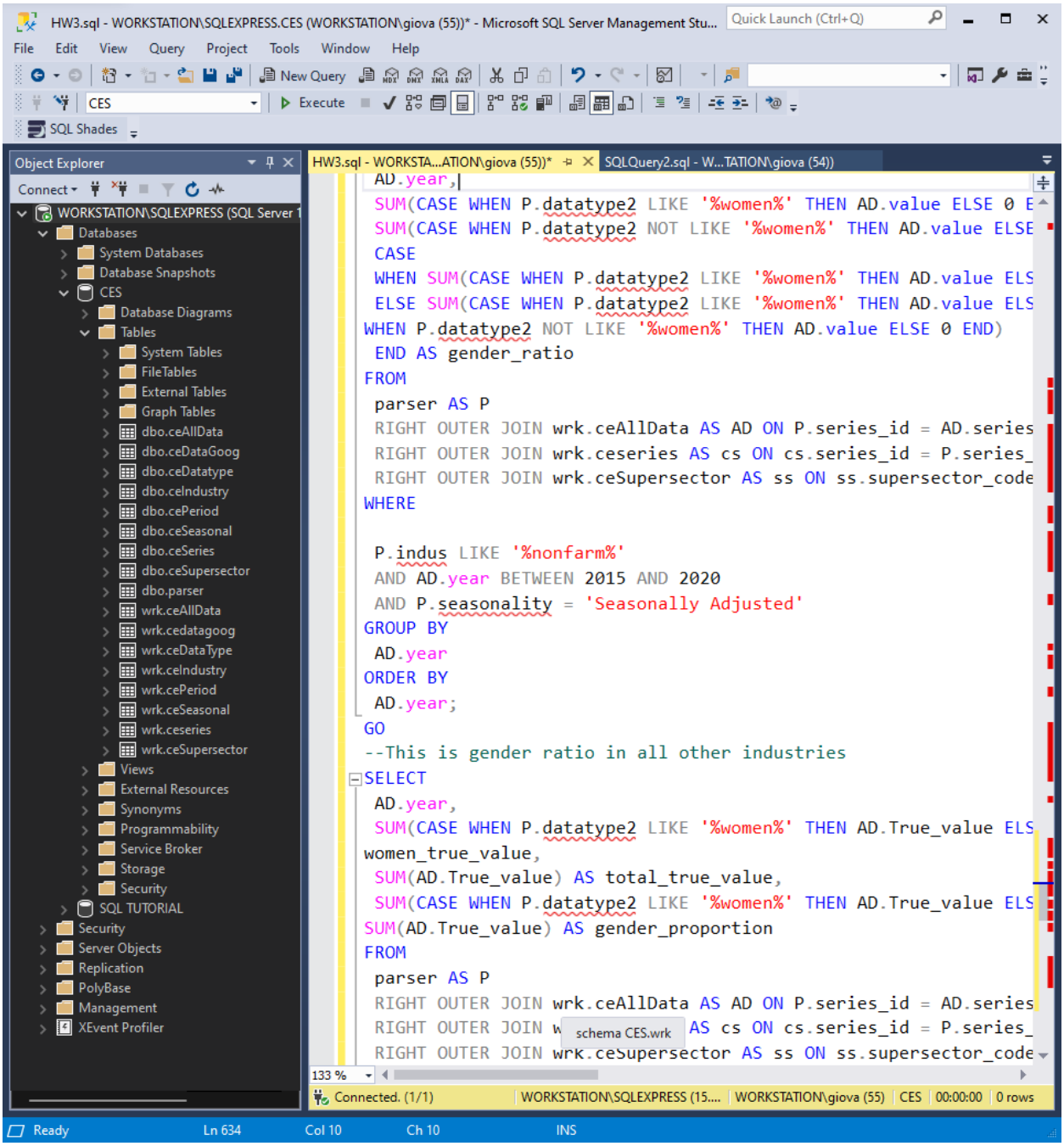
/* Question 4 Using the Total Non-Farm Employment series as base,
determine the ratio of total employment in each employment sector by

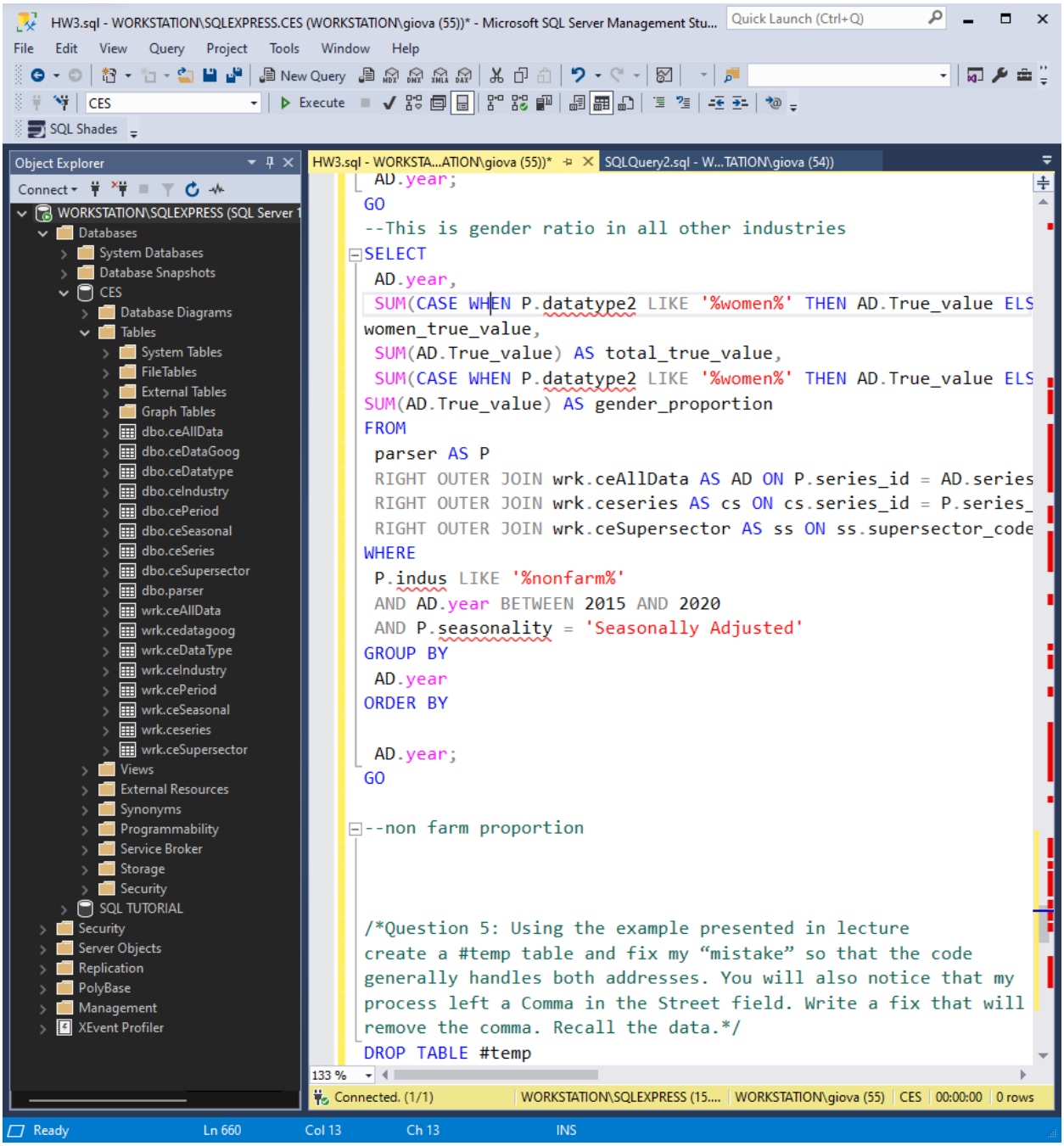
```
industry and year for 2015 to 2020.
Determine the gender breakdown of
each industry between 2015 to 2020. */
SELECT
 AD.year,
 SUM(AD.value) AS total_value,
 SUM(AD.True_value) AS total_true_value,
 P.datatype2
FROM
 parser AS P
 RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series_id
RIGHT OUTER JOIN wrk.ceseries AS cs on cs.series_id = P.series_id
RIGHT OUTER JOIN wrk.ceSupersector AS ss on ss.supersector_code = cs.supersector_code
WHERE
 P.datatype2 LIKE '%women%'
 AND P.indus LIKE '%nonfarm%'
 AND AD.year BETWEEN 2015 AND 2020
 AND P.seasonality = 'Seasonally Adjusted'
GROUP BY
 AD.year,
 P.datatype2
ORDER BY
 AD.year;
GO
-- This is total women employees by year in non farm
SELECT
 AD.year,
 SUM(AD.value) AS total_value,
 SUM(AD.True_value) AS total_true_value,
 P.indus,
 P.datatype2
FROM
 parser AS P
 RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series_id
 RIGHT OUTER JOIN wrk.ceseries AS cs ON cs.series_id = P.series_id
 RIGHT OUTER JOIN wrk.ceSupersector AS ss ON ss.supersector_code = cs.supersector_code

WHERE
 P.datatype2 LIKE '%women%'
 AND AD.year BETWEEN 2015 AND 2020
 AND P.seasonality = 'Seasonally Adjusted'
GROUP BY
 AD.year,
 P.indus,
 P.datatype2
ORDER BY
 AD.year;
GO
--this is women by each industry
SELECT
 AD.year,
 SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELSE 0 END) AS total_value_women,
 SUM(CASE WHEN P.datatype2 NOT LIKE '%women%' THEN AD.value ELSE 0 END) AS
total_value_all,
```

```
 CASE
 WHEN SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELSE 0 END) = 0 THEN 0
 ELSE SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELSE 0 END) / SUM(CASE
WHEN P.datatype2 NOT LIKE '%women%' THEN AD.value ELSE 0 END)
 END AS gender_ratio
FROM
 parser AS P
 RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series_id
 RIGHT OUTER JOIN wrk.ceseries AS cs ON cs.series_id = P.series_id
 RIGHT OUTER JOIN wrk.ceSupersector AS ss ON ss.supersector_code = cs.supersector_code
WHERE

 P.indus LIKE '%nonfarm%'
 AND AD.year BETWEEN 2015 AND 2020
 AND P.seasonality = 'Seasonally Adjusted'
GROUP BY
 AD.year
ORDER BY
 AD.year;
GO
--This is gender ratio in all other industries
SELECT
 AD.year,
 SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.True_value ELSE 0 END) AS
women_true_value,
 SUM(AD.True_value) AS total_true_value,
 SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.True_value ELSE 0 END) /
SUM(AD.True_value) AS gender_proportion
FROM
 parser AS P
 RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series_id
 RIGHT OUTER JOIN wrk.ceseries AS cs ON cs.series_id = P.series_id
 RIGHT OUTER JOIN wrk.ceSupersector AS ss ON ss.supersector_code = cs.supersector_code
WHERE
 P.indus LIKE '%nonfarm%'
 AND AD.year BETWEEN 2015 AND 2020
 AND P.seasonality = 'Seasonally Adjusted'
GROUP BY
 AD.year
ORDER BY

 AD.year;
GO

--non farm proportion
```


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
/* Question 4 Using the Total Non-Farm Employment series as base
determine the ratio of total employment in each employment secto
industry and year for 2015 to 2020. Determine the gender breakd
each industry between 2015 to 2020. */
SELECT
AD.year,
SUM(AD.value) AS total_value,
SUM(AD.True_value) AS total_true_value,
P.datatype2
FROM
parser AS P
RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series
RIGHT OUTER JOIN wrk.ceseries AS cs on cs.series_id = P.series_i
RIGHT OUTER JOIN wrk.ceSupersector AS ss on ss.supersector_code
WHERE
P.datatype2 LIKE '%women%'
AND P.indus LIKE '%nonfarm%'
AND AD.year BETWEEN 2015 AND 2020
AND P.seasonality = 'Seasonally Adjusted'
GROUP BY
AD.year,
P.datatype2
ORDER BY
AD.year;
GO
-- This is total women employees by year in non farm
SELECT
AD.year,
SUM(AD.value) AS total_value,
SUM(AD.True_value) AS total_true_value,
P.indus,
P.datatype2
FROM
133 %
Connected. (1/1)
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 587
Col 1
Ch 1
INS

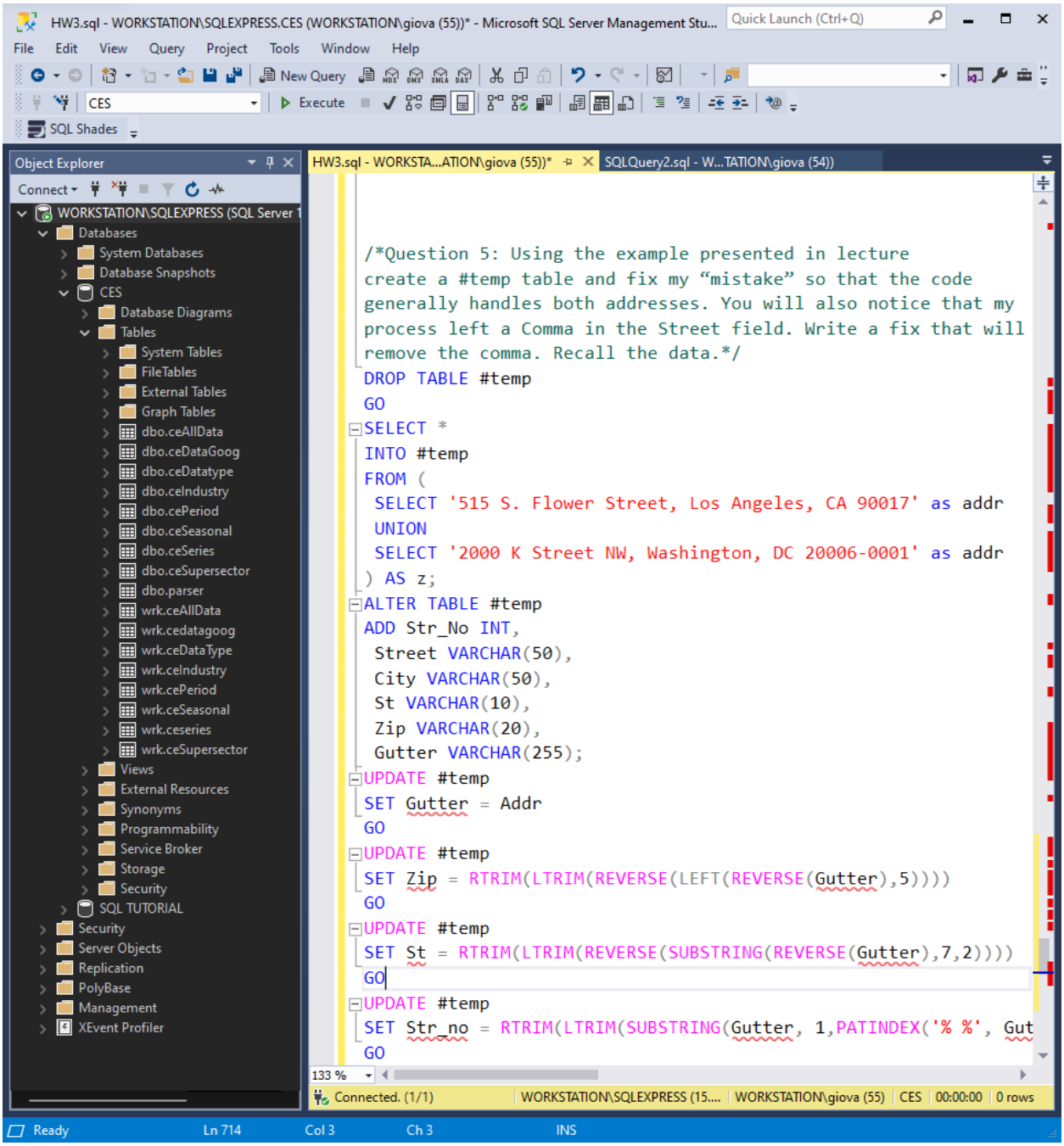

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
GO
-- This is total women employees by year in non farm
SELECT
AD.year,
SUM(AD.value) AS total_value,
SUM(AD.True_value) AS total_true_value,
P.indus,
P.datatype2
FROM
parser AS P
RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series
RIGHT OUTER JOIN wrk.ceseries AS cs ON cs.series_id = P.series_
RIGHT OUTER JOIN wrk.ceSupersector AS ss ON ss.supersector_code
WHERE
P.datatype2 LIKE '%women%'
AND AD.year BETWEEN 2015 AND 2020
AND P.seasonality = 'Seasonally Adjusted'
GROUP BY
AD.year,
P.indus,
P.datatype2
ORDER BY
AD.year;
GO
--this is women by each industry
SELECT
AD.year,
SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELSE 0 E
SUM(CASE WHEN P.datatype2 NOT LIKE '%women%' THEN AD.value ELSE
CASE
WHEN SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELS
ELSE SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELS
WHEN P.datatype2 NOT LIKE '%women%' THEN AD.value ELSE 0 END)
END AS gender_ratio
FROM
133 %
Connected. (1/1)
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 611
Col 12
Ch 12
INS

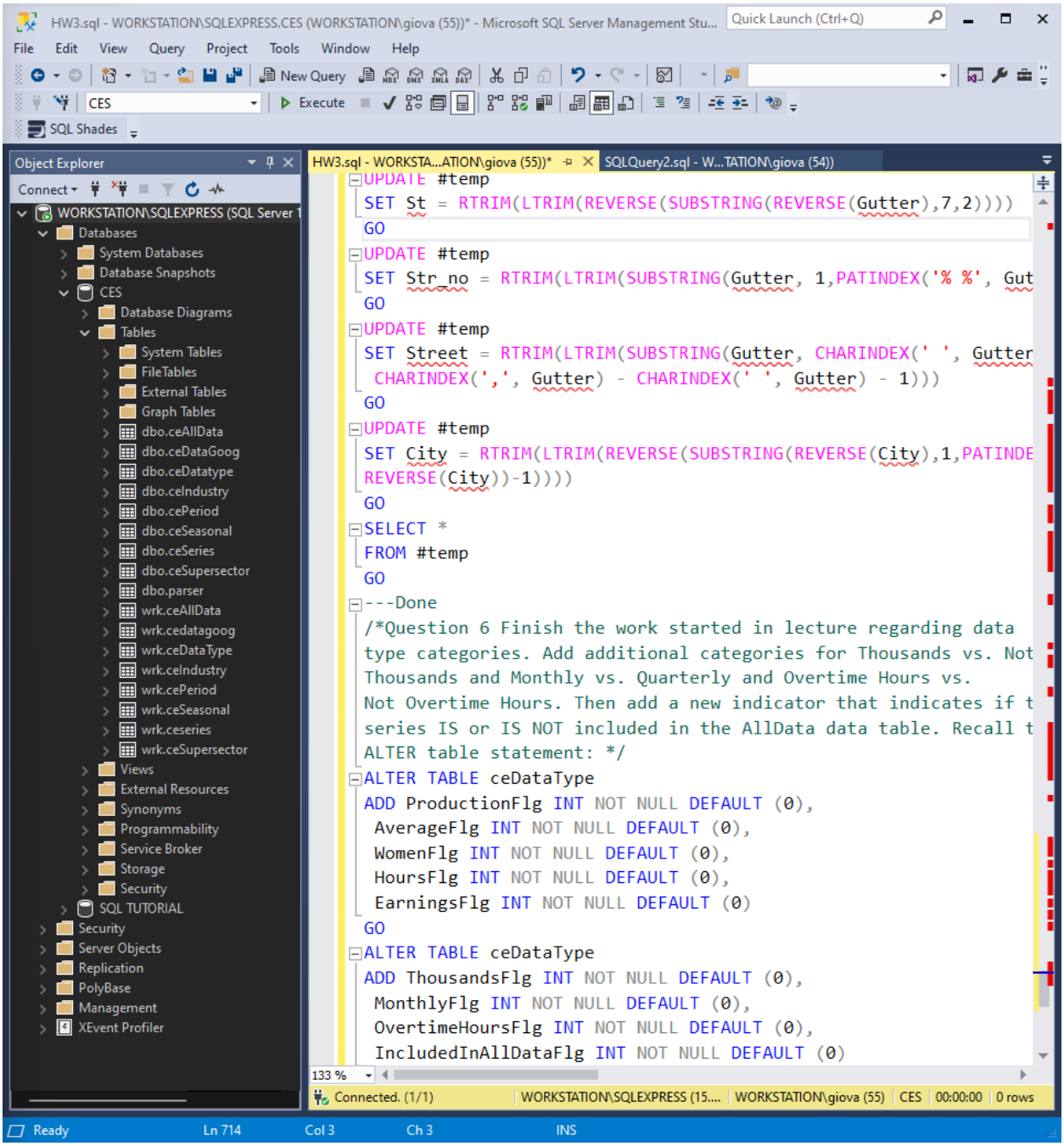

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
AD.year,
SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELSE 0 E
SUM(CASE WHEN P.datatype2 NOT LIKE '%women%' THEN AD.value ELSE
CASE
WHEN SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELS
ELSE SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.value ELS
WHEN P.datatype2 NOT LIKE '%women%' THEN AD.value ELSE 0 END)
END AS gender_ratio
FROM
parser AS P
RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series
RIGHT OUTER JOIN wrk.ceseries AS cs ON cs.series_id = P.series_
RIGHT OUTER JOIN wrk.ceSupersector AS ss ON ss.supersector_code
WHERE
P.indus LIKE '%nonfarm%'
AND AD.year BETWEEN 2015 AND 2020
AND P.seasonality = 'Seasonally Adjusted'
GROUP BY
AD.year
ORDER BY
AD.year;
GO
--This is gender ratio in all other industries
SELECT
AD.year,
SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.True_value ELS
women_true_value,
SUM(AD.True_value) AS total_true_value,
SUM(CASE WHEN P.datatype2 LIKE '%women%' THEN AD.True_value ELS
SUM(AD.True_value) AS gender_proportion
FROM
parser AS P
RIGHT OUTER JOIN wrk.ceAllData AS AD ON P.series_id = AD.series
RIGHT OUTER JOIN w
schema CES.wrk
AS cs ON cs.series_id = P.series_
RIGHT OUTER JOIN wrk.ceSupersector AS ss ON ss.supersector_code
133 %
Connected. (1/1)
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 634
Col 10
Ch 10
INS




```
/*Question 5: Using the example presented
in lecture
create a #temp table and fix my “mistake”
so that the code
```

```
generally handles both addresses. You will also notice that my
process left a Comma in the Street field. Write a fix that will
remove the comma. Recall the data.*/

DROP TABLE #temp
GO
SELECT *
INTO #temp
FROM (
 SELECT '515 S. Flower Street, Los Angeles, CA 90017' as addr
 UNION
 SELECT '2000 K Street NW, Washington, DC 20006-0001' as addr
) AS z;
ALTER TABLE #temp
ADD Str_No INT,
 Street VARCHAR(50),
 City VARCHAR(50),
 St VARCHAR(10),
 Zip VARCHAR(20),
 Gutter VARCHAR(255);
UPDATE #temp
SET Gutter = Addr
GO
UPDATE #temp
SET Zip = RTRIM(LTRIM(REVERSE(LEFT(REVERSE(Gutter),5))))
GO
UPDATE #temp
SET St = RTRIM(LTRIM(REVERSE(SUBSTRING(REVERSE(Gutter),7,2))))
GO
UPDATE #temp
SET Str_no = RTRIM(LTRIM(SUBSTRING(Gutter, 1,PATINDEX('% %', Gutter))))
GO
UPDATE #temp
SET Street = RTRIM(LTRIM(SUBSTRING(Gutter, CHARINDEX(' ', Gutter) + 1,
 CHARINDEX(',', Gutter) - CHARINDEX(' ', Gutter) - 1)))
GO
UPDATE #temp
SET City = RTRIM(LTRIM(REVERSE(SUBSTRING(REVERSE(City),1,PATINDEX('%,%',
REVERSE(City))-1))))
GO
SELECT *
FROM #temp
GO
---Done
```

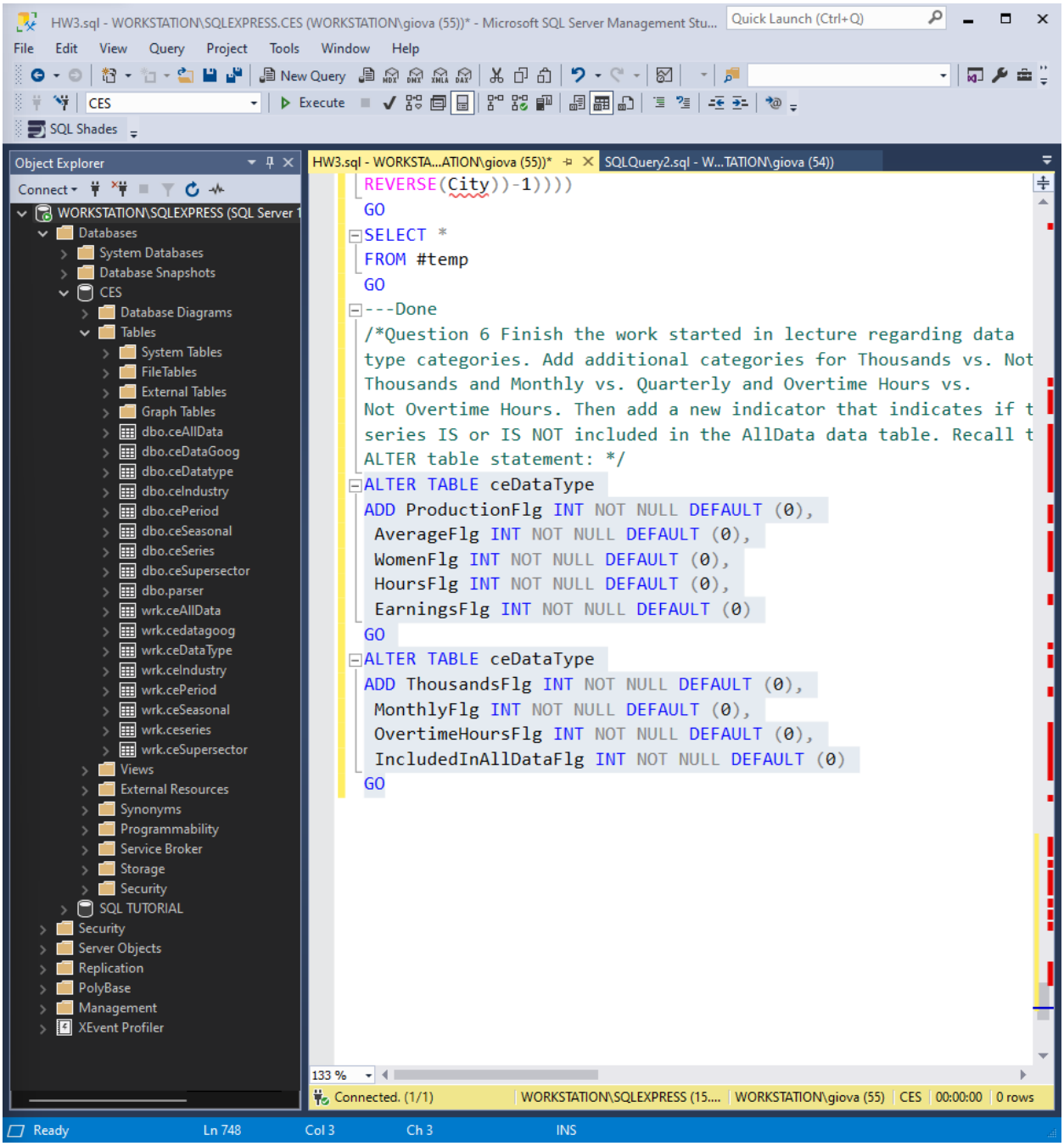

HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
/*Question 5: Using the example presented in lecture
create a #temp table and fix my “mistake” so that the code
generally handles both addresses. You will also notice that my
process left a Comma in the Street field. Write a fix that will
remove the comma. Recall the data.*/
DROP TABLE #temp
GO
SELECT *
INTO #temp
FROM (
SELECT '515 S. Flower Street, Los Angeles, CA 90017' as addr
UNION
SELECT '2000 K Street NW, Washington, DC 20006-0001' as addr
) AS z;
ALTER TABLE #temp
ADD Str_No INT,
Street VARCHAR(50),
City VARCHAR(50),
St VARCHAR(10),
Zip VARCHAR(20),
Gutter VARCHAR(255);
UPDATE #temp
SET Gutter = Addr
GO
UPDATE #temp
SET Zip = RTRIM(LTRIM(REVERSE(LEFT(REVERSE(Gutter),5))))
GO
UPDATE #temp
SET St = RTRIM(LTRIM(REVERSE(SUBSTRING(REVERSE(Gutter),7,2))))
GO
UPDATE #temp
SET Str_no = RTRIM(LTRIM(SUBSTRING(Gutter, 1,PATINDEX('% %', Gut
GO
133 %
Connected. (1/1)
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 714
Col 3
Ch 3
INS




```
/*Question 6 Finish the work started in
lecture regarding data
type categories. Add additional categories
for Thousands vs. Not
Thousands and Monthly vs. Quarterly and
Overtime Hours vs.
```

Not Overtime Hours. Then add a new indicator that indicates if the series IS or IS NOT included in the AllData data table. Recall the ALTER table statement: */

```
ALTER TABLE ceDataType
ADD ProductionFlg INT NOT NULL DEFAULT (0),
 AverageFlg INT NOT NULL DEFAULT (0),
 WomenFlg INT NOT NULL DEFAULT (0),
 HoursFlg INT NOT NULL DEFAULT (0),
 EarningsFlg INT NOT NULL DEFAULT (0)
GO
ALTER TABLE ceDataType
ADD ThousandsFlg INT NOT NULL DEFAULT (0),
 MonthlyFlg INT NOT NULL DEFAULT (0),
 OvertimeHoursFlg INT NOT NULL DEFAULT (0),
 IncludedInAllDataFlg INT NOT NULL DEFAULT (0)
GO
```


HW3.sql - WORKSTATION\SQLEXPRESS.CES (WORKSTATION\giova (55))* - Microsoft SQL Server Management Stu...
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
CES
Execute
SQL Shades
Object Explorer
Connect
WORKSTATION\SQLEXPRESS (SQL Server 1
Databases
System Databases
Database Snapshots
CES
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.ceAllData
dbo.ceDataGoog
dbo.ceDatatype
dbo.ceIndustry
dbo.cePeriod
dbo.ceSeasonal
dbo.ceSeries
dbo.ceSupersector
dbo.parser
wrk.ceAllData
wrk.cedatagoog
wrk.ceDataType
wrk.ceIndustry
wrk.cePeriod
wrk.ceSeasonal
wrk.ceseries
wrk.ceSupersector
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
SQL TUTORIAL
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler
HW3.sql - WORKSTA...ATION\giova (55))*
SQLQuery2.sql - W...TATION\giova (54))
REVERSE(City))-1))))
GO
SELECT *
FROM #temp
GO
---Done
/*Question 6 Finish the work started in lecture regarding data
type categories. Add additional categories for Thousands vs. Not
Thousands and Monthly vs. Quarterly and Overtime Hours vs.
Not Overtime Hours. Then add a new indicator that indicates if t
series IS or IS NOT included in the AllData data table. Recall t
ALTER table statement: */
ALTER TABLE ceDataType
ADD ProductionFlg INT NOT NULL DEFAULT (0),
AverageFlg INT NOT NULL DEFAULT (0),
WomenFlg INT NOT NULL DEFAULT (0),
HoursFlg INT NOT NULL DEFAULT (0),
EarningsFlg INT NOT NULL DEFAULT (0)
GO
ALTER TABLE ceDataType
ADD ThousandsFlg INT NOT NULL DEFAULT (0),
MonthlyFlg INT NOT NULL DEFAULT (0),
OvertimeHoursFlg INT NOT NULL DEFAULT (0),
IncludedInAllDataFlg INT NOT NULL DEFAULT (0)
GO
133 %
Connected. (1/1)
WORKSTATION\SQLEXPRESS (15....
WORKSTATION\giova (55)
CES
00:00:00
0 rows
Ready
Ln 748
Col 3
Ch 3
INS