Getting the Information You Need from CDW: SQL Starter Language

by Margaret Gonsoulin, PhD

VIReC
RESEARCHERS' GUIDE TO VA DATA

Thanks to:

- Richard Pham, BISL/CDW
- Colleagues at VIReC

VIReC
RESEARCHERS' GUIDE TO VA DATA

Agenda for Today

- This is follow-up to:
  - First Time Research User's Guide to CDW
  - Seeing the Data When You Can't See the Data
- Review key terms and definitions
- Orient new SQL users to CDW's data organization
- Provide new CDW users with a few basic SQL programming skills

VIReC
RESEARCHERS' GUIDE TO VA DATA

Reminder – 2 Domain Types

- Production Domains – contains tables that have been structured by database architects to support their re-joining.
- Raw Domains – contains tables that are direct extracts from the source system (e.g., VistA) that are simply being housed with no editing performed on them.
- These types of domains also vary in terms of their naming traditions.

CDW, VINCI & SQL

- CDW –Corporate Data Warehouse – a relational database that contains significant portions of the VA's electronic health record.
- VINCI – VA Informatics and Computing Infrastructure – (in part) a remote server that provides software and holds a copy of CDW.
- SQL – Structured Query Language - a programming language used to retrieve information from a relational database.

SSMS -SQL Server Management Studio

- SSMS is computer software that can be used to write and execute SQL code.
  - This talk assumes that you have access to CDW data and SSMS.
  - The examples included in this talk use oversimplified material from the CDW in order to illustrate the logic of SQL.
  - Therefore, this talk is not intended to demonstrate a viable research investigation.

By the end of this talk,

We hope that a new CDW user will:

- Have a basic understanding of SQL
- Be able to read a basic SQL query
- Be able to write a basic SQL query
  - For tables in the production domains
  - For tables in the raw domains

Poll #1: Your CDW Experience

Rate your level of experience with CDW data on a scale of 1 to 5…

1 Not worked with it at all

2

3

4

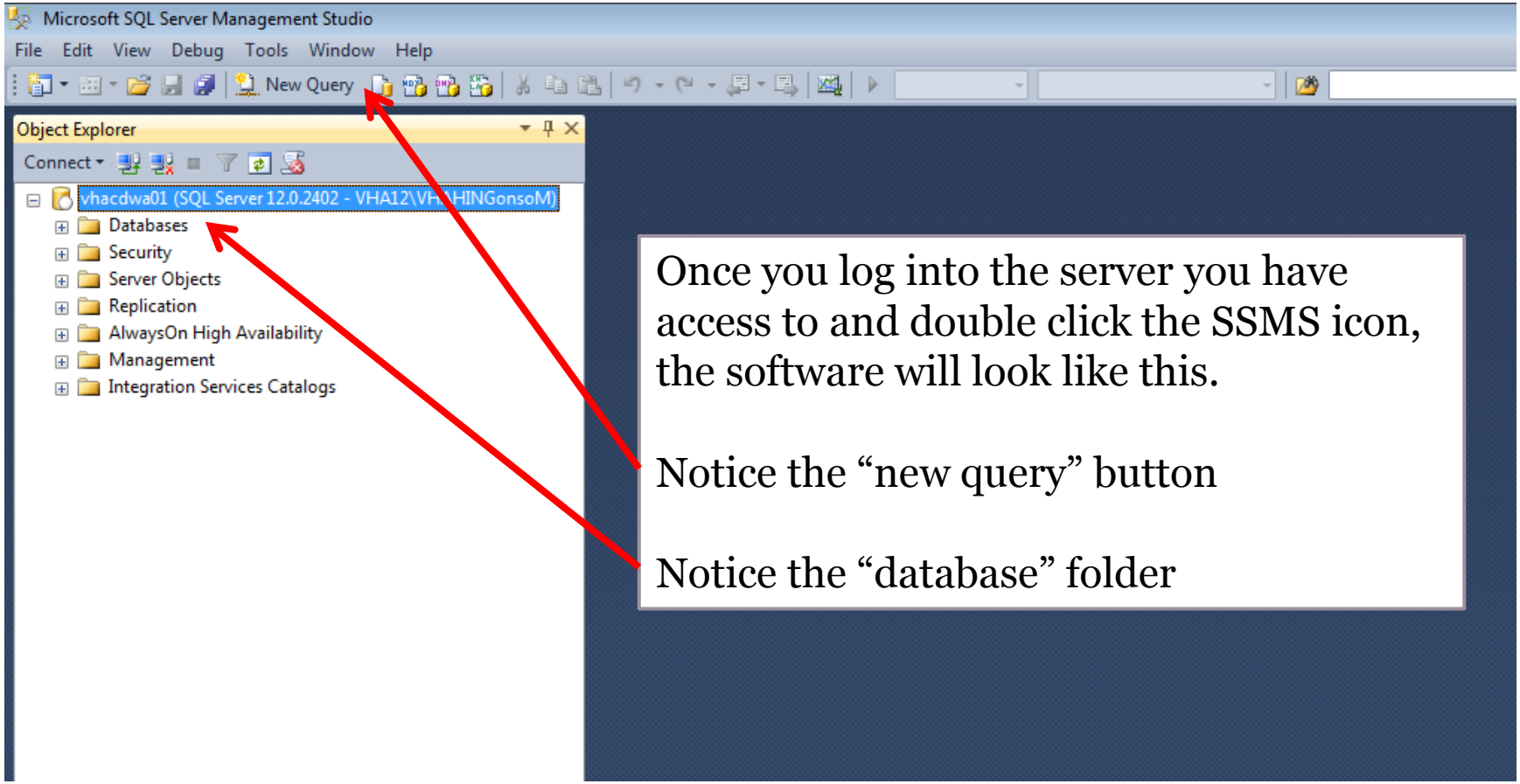
5 Very experienced with CDW

VIReC
RESEARCHERS' GUIDE TO VA DATA

In the Production Domains

The Production table called Dim.ICD9

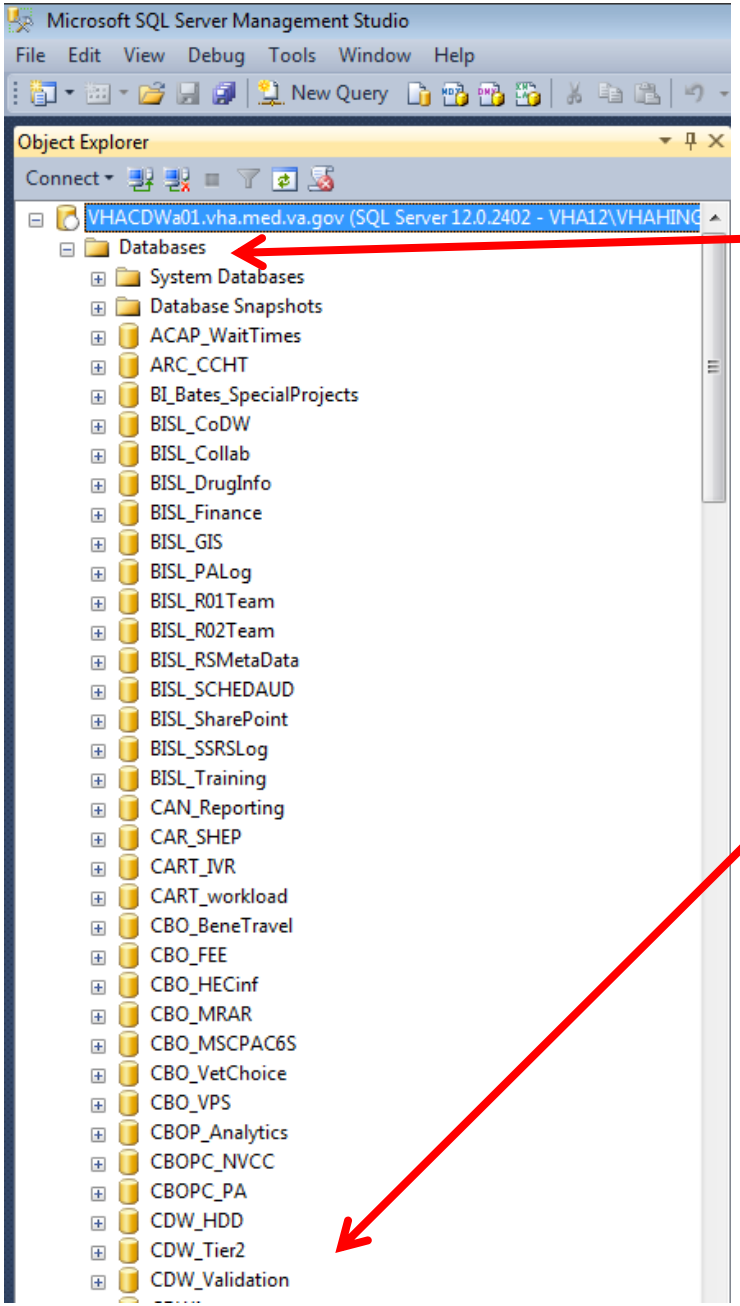
SQL Server Management Studio



Once you log into the server you have access to and double click the SSMS icon, the software will look like this.

Notice the “new query” button

Notice the “database” folder

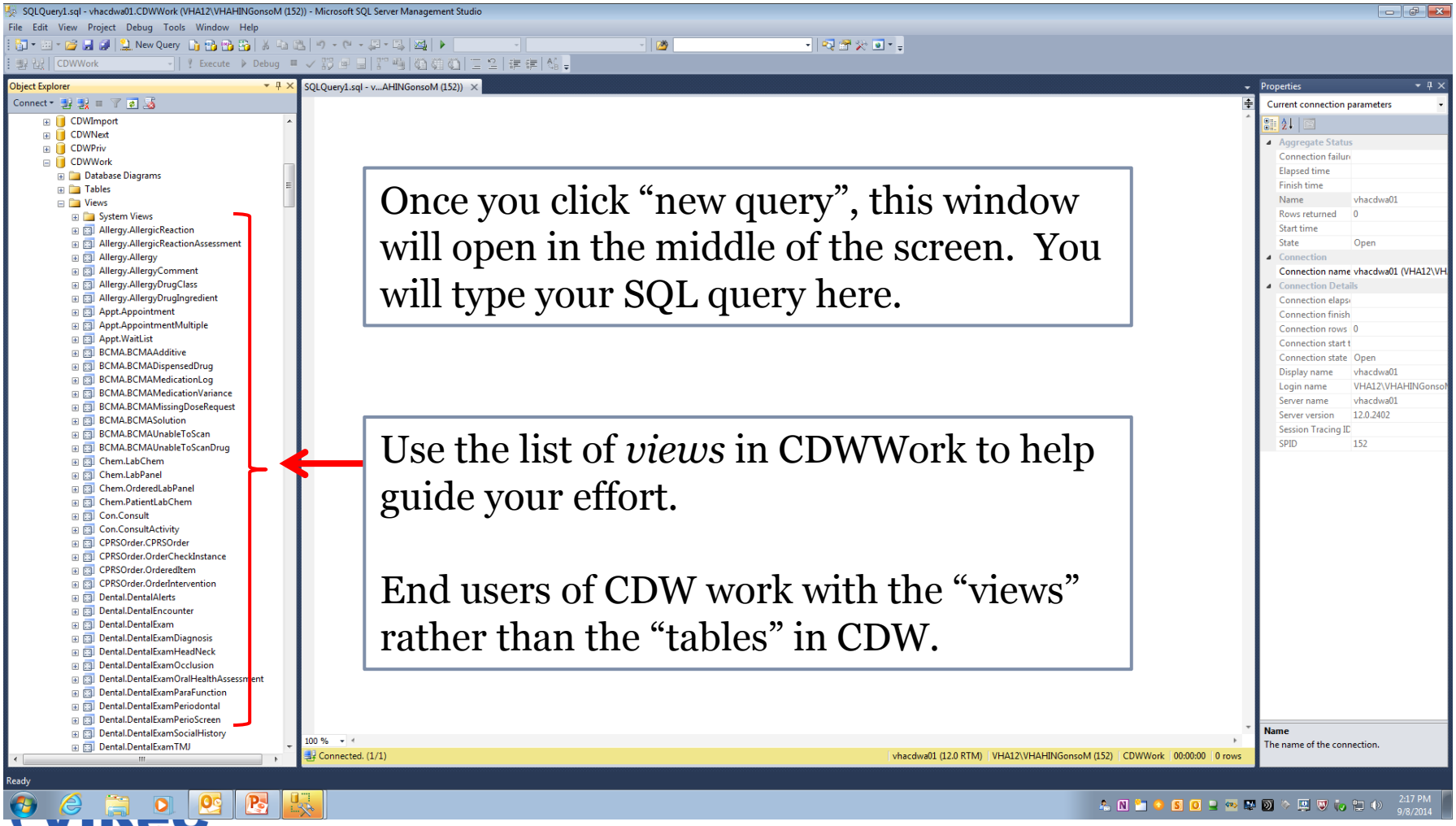


Expand “Databases”

Then, scroll down to find “CDWWork”

Note: this example uses a view of a dimension table in the database folder called CDWWork; all people with basic access to CDW will be able to see this view.

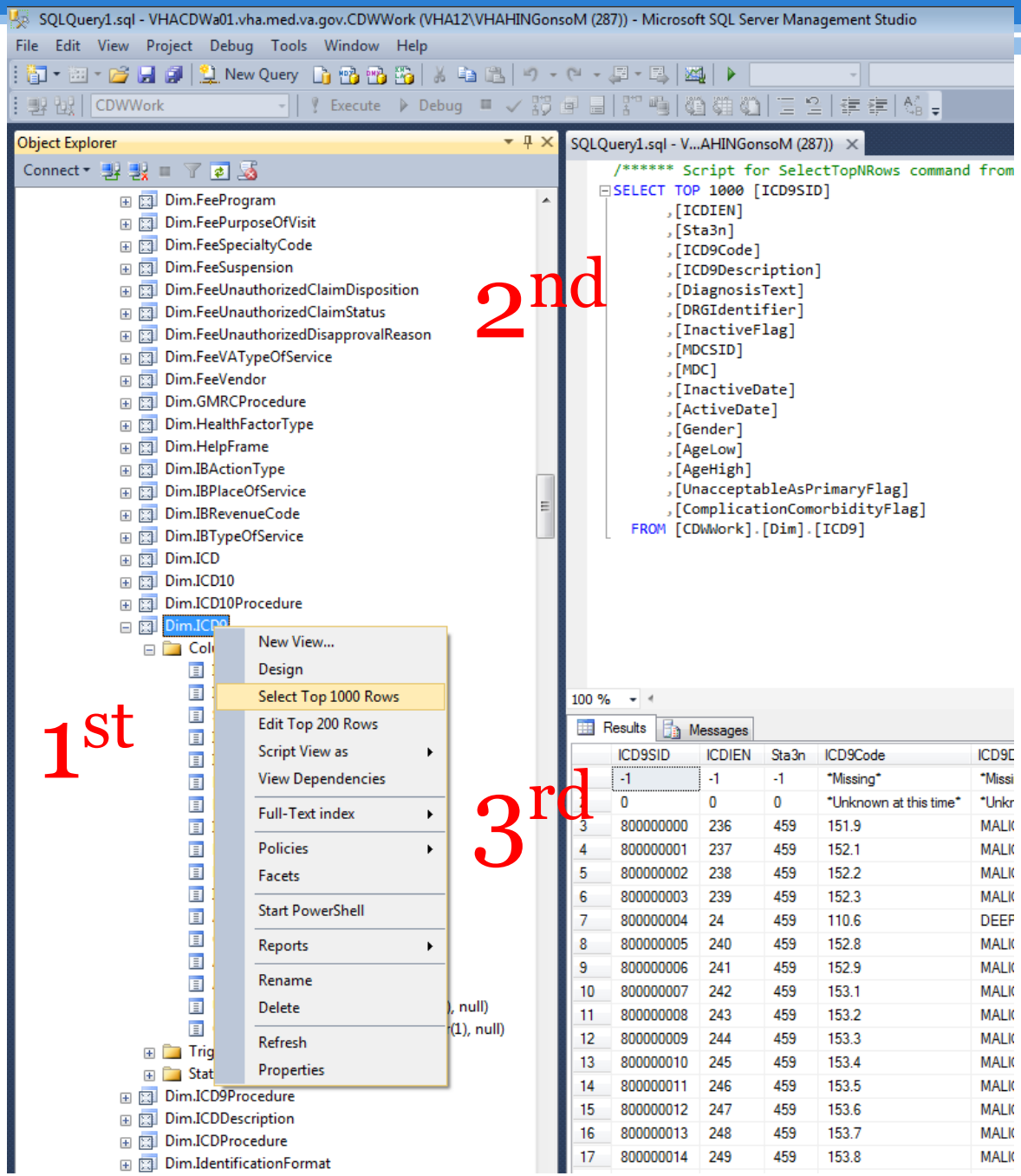
New Query



Once you click “new query”, this window will open in the middle of the screen. You will type your SQL query here.

Use the list of *views* in CDWWork to help guide your effort.

End users of CDW work with the “views” rather than the “tables” in CDW.



1st Right click the view name for this drop-down menu and then click "select top 1000 rows"

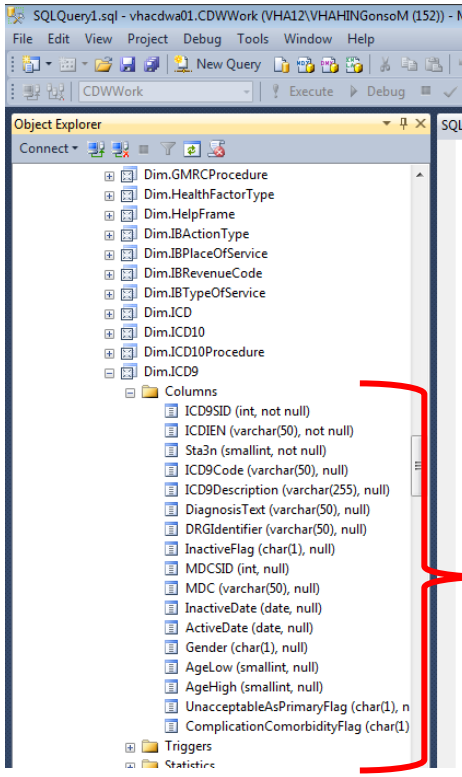
2nd The query seen in the right top screen is automatically written

3rd The results below it automatically appear.

SELECT and FROM

- SELECT allows a programmer to list the columns (variables) that they would like to see in the results of their query
  - Each column name should be followed by a comma except the last one in the list
- FROM selects that appropriate *view* from which the columns will be collected
  - The name of the view should be written in this format "database.schema.*view*"

SELECT & FROM



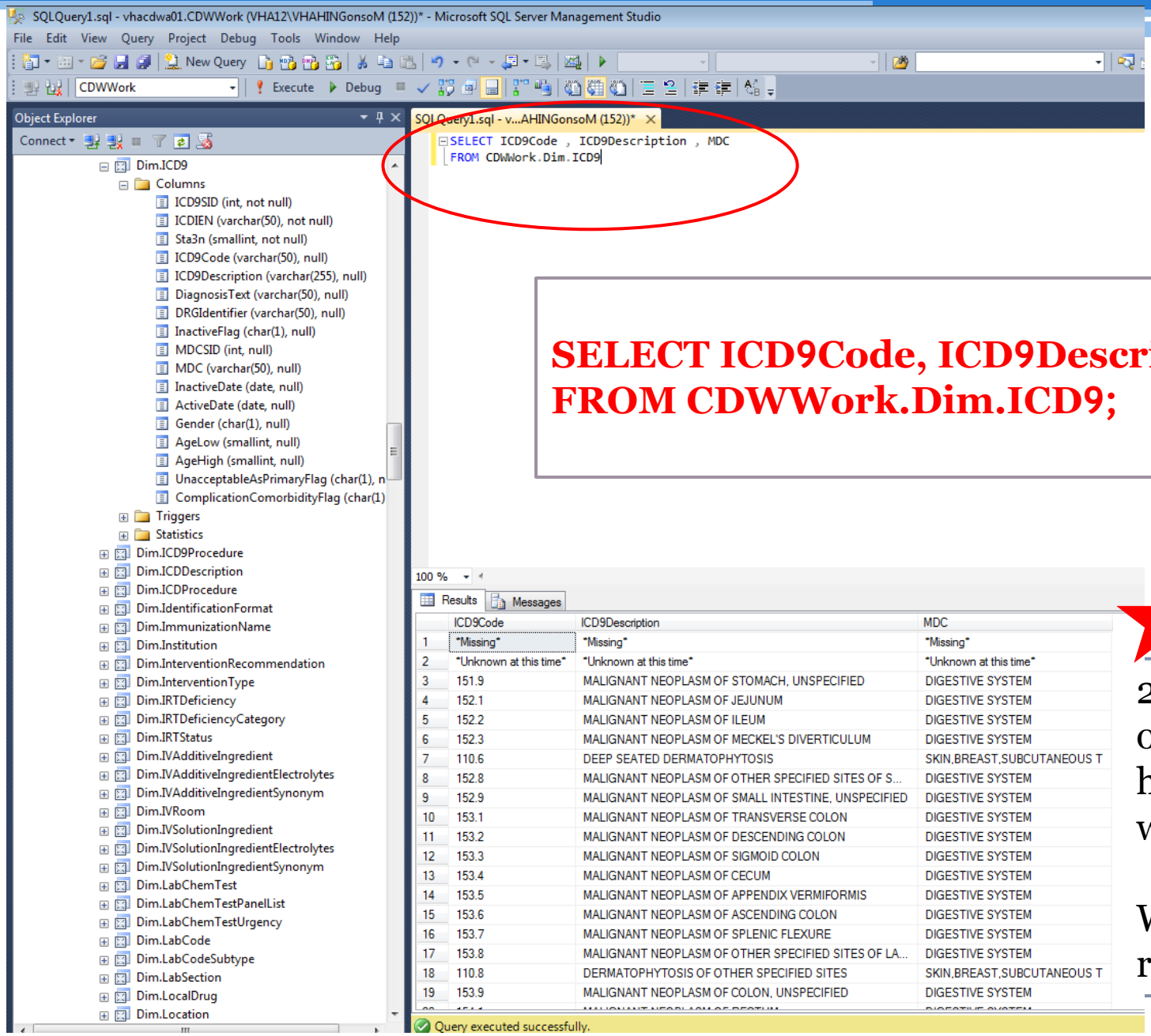
- Examining the metadata reports first is ideal.
- For this example, we will use the *view* called Dim.ICD9.
- For this example, we will use the following columns: ICD9Code, ICD9Description and MDC.

We will use this form:

SELECT column1, column2, column3
FROM Database.Schema.View;

VIReC RESEARCHERS' GUIDE TO VA DATA

*MDC stands for Major Diagnostic Category
and ICD stands for International Classification of Diseases



	ICD9Code	ICD9Description	MDC
1	"Missing"	"Missing"	"Missing"
2	"Unknown at this time"	"Unknown at this time"	"Unknown at this time"
3	151.9	MALIGNANT NEOPLASM OF STOMACH, UNSPECIFIED	DIGESTIVE SYSTEM
4	152.1	MALIGNANT NEOPLASM OF JEJUNUM	DIGESTIVE SYSTEM
5	152.2	MALIGNANT NEOPLASM OF ILEUM	DIGESTIVE SYSTEM
6	152.3	MALIGNANT NEOPLASM OF MECKEL'S DIVERTICULUM	DIGESTIVE SYSTEM
7	110.6	DEEP SEATED DERMATOPHYTOSIS	SKIN,BREAST,SUBCUTANEOUS T
8	152.8	MALIGNANT NEOPLASM OF OTHER SPECIFIED SITES OF S...	DIGESTIVE SYSTEM
9	152.9	MALIGNANT NEOPLASM OF SMALL INTESTINE, UNSPECIFIED	DIGESTIVE SYSTEM
10	153.1	MALIGNANT NEOPLASM OF TRANSVERSE COLON	DIGESTIVE SYSTEM
11	153.2	MALIGNANT NEOPLASM OF DESCENDING COLON	DIGESTIVE SYSTEM
12	153.3	MALIGNANT NEOPLASM OF SIGMOID COLON	DIGESTIVE SYSTEM
13	153.4	MALIGNANT NEOPLASM OF CECUM	DIGESTIVE SYSTEM
14	153.5	MALIGNANT NEOPLASM OF APPENDIX VERMIFORMIS	DIGESTIVE SYSTEM
15	153.6	MALIGNANT NEOPLASM OF ASCENDING COLON	DIGESTIVE SYSTEM
16	153.7	MALIGNANT NEOPLASM OF SPLENIC FLEXURE	DIGESTIVE SYSTEM
17	153.8	MALIGNANT NEOPLASM OF OTHER SPECIFIED SITES OF LA...	DIGESTIVE SYSTEM
18	110.8	DERMATOPHYTOSIS OF OTHER SPECIFIED SITES	SKIN,BREAST,SUBCUTANEOUS T
19	153.9	MALIGNANT NEOPLASM OF COLON, UNSPECIFIED	DIGESTIVE SYSTEM

SELECT ICD9Code, ICD9Description, MDC
FROM CDWWork.Dim.ICD9;

2,025,871 results or rows appear here in a new window.

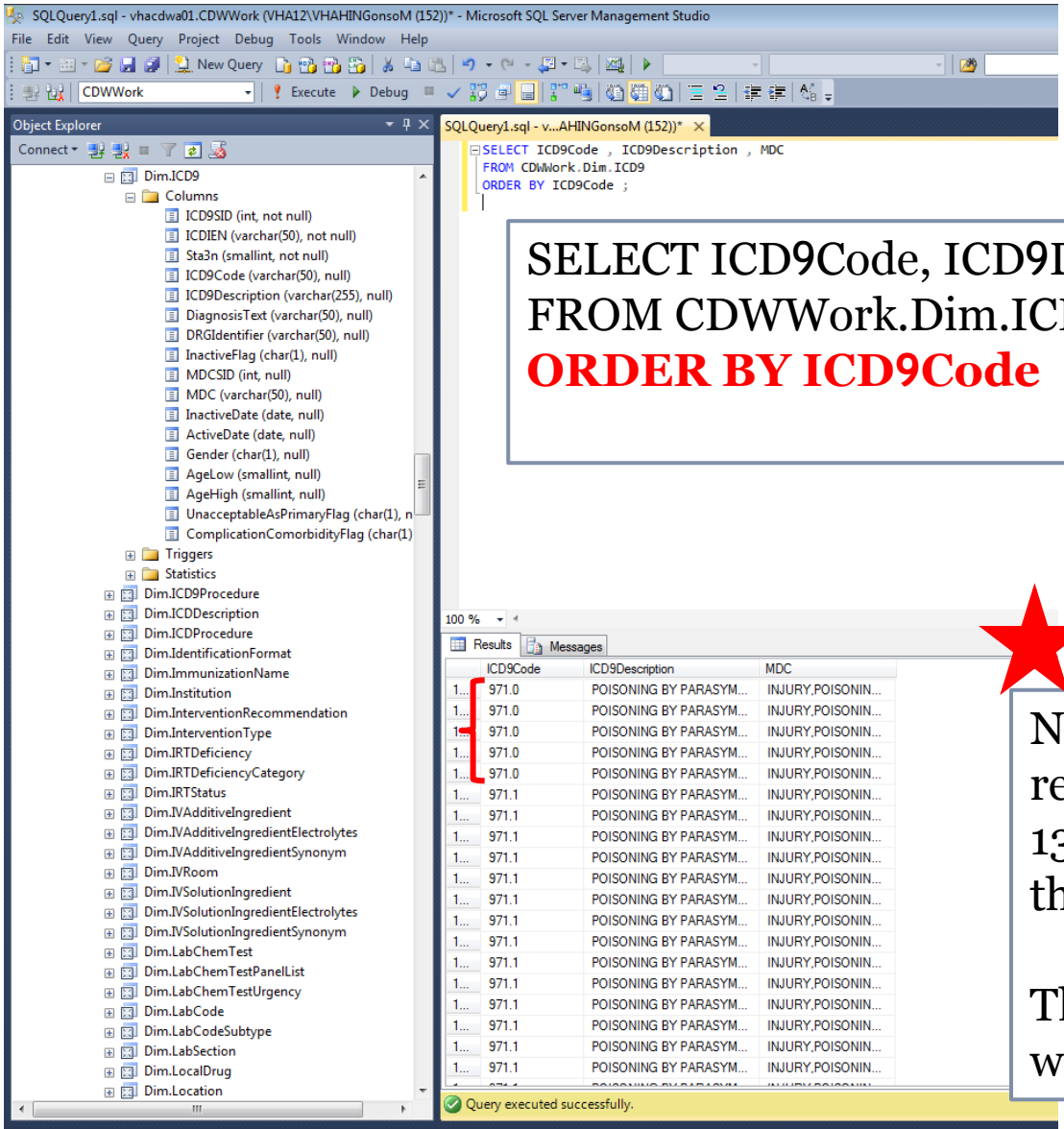
We can ask for the results in order.

ORDER BY

- The ORDER BY command allows the programmer to see the results in alpha-numeric order based on the column that is chosen.
- ORBER BY must be the last command in a query.

```
SELECT column1, column2, column3
FROM Database.Schema.View
ORDER BY column1 ;
```

ORDER BY



SELECT ICD9Code, ICD9Description, MDC
FROM CDWWork.Dim.ICD9
ORDER BY ICD9Code ;

Notice that the same ICD codes repeat in the data because the 130 VistA systems have each sent their own list.

The repetition can be eliminated with a new command.

GROUP BY

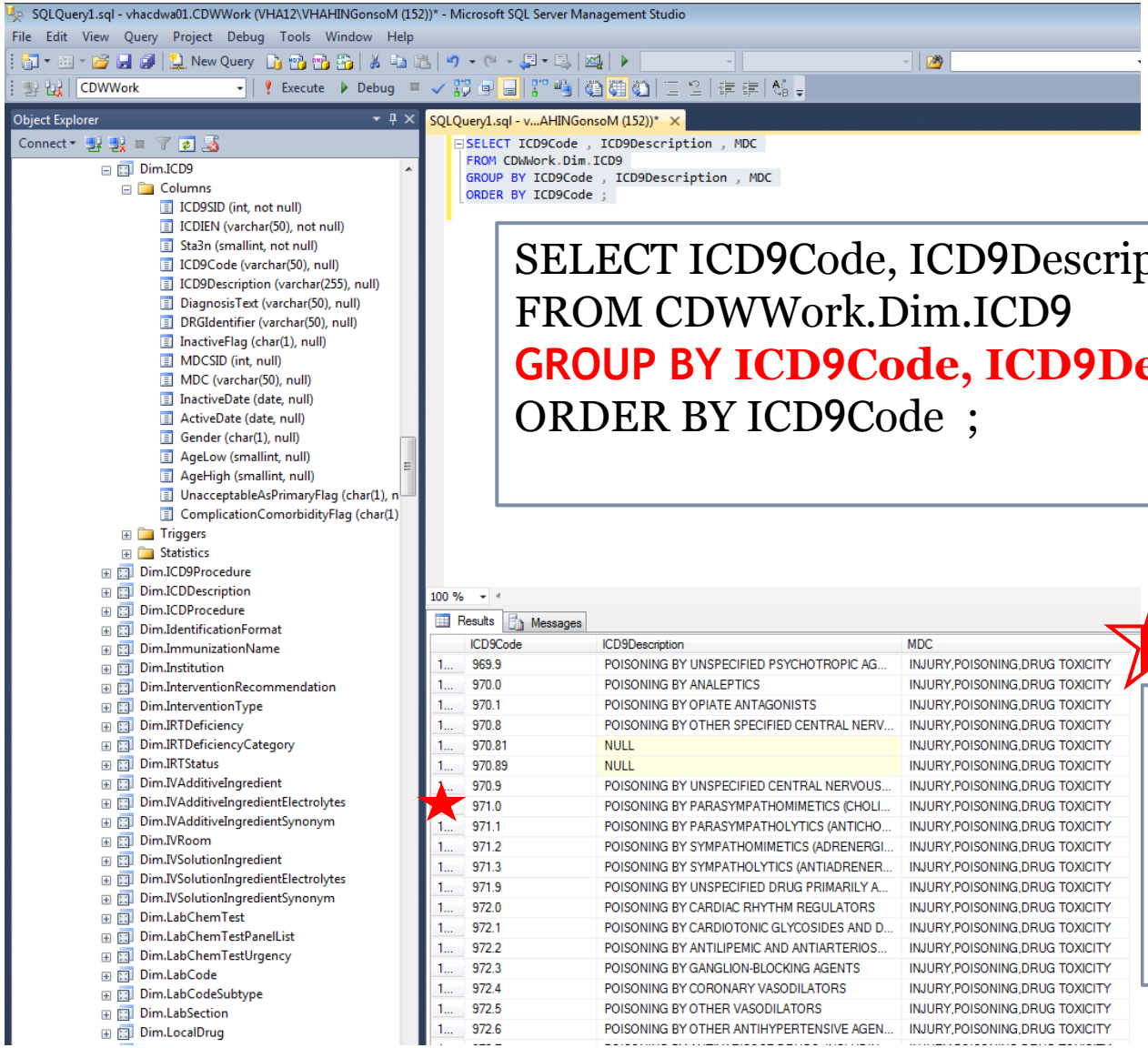
- The GROUP BY command collapses identical values into a single entry rather than allowing repetitive rows to exist in the output.
- Remembering that ORDER BY must be the last command in a query, GROUP BY should precede it.
- Every column that is not an aggregated* measure must be included in the GROUP BY clause.

```
SELECT column1, column2, column3
FROM Database.Schema.Table
GROUP BY column1, column2 , column3
ORDER BY column1 ;
```

VIReC RESEARCHERS' GUIDE TO VA DATA

*Sums, averages, minimum values… would be examples of aggregate measures

GROUP BY




```
SELECT ICD9Code, ICD9Description, MDC
FROM CDWWork.Dim.ICD9
GROUP BY ICD9Code, ICD9Description, MDC
ORDER BY ICD9Code  ;
```

Now there are 16,645 rows. We can see the ICD code of 971.0 only has a single row.

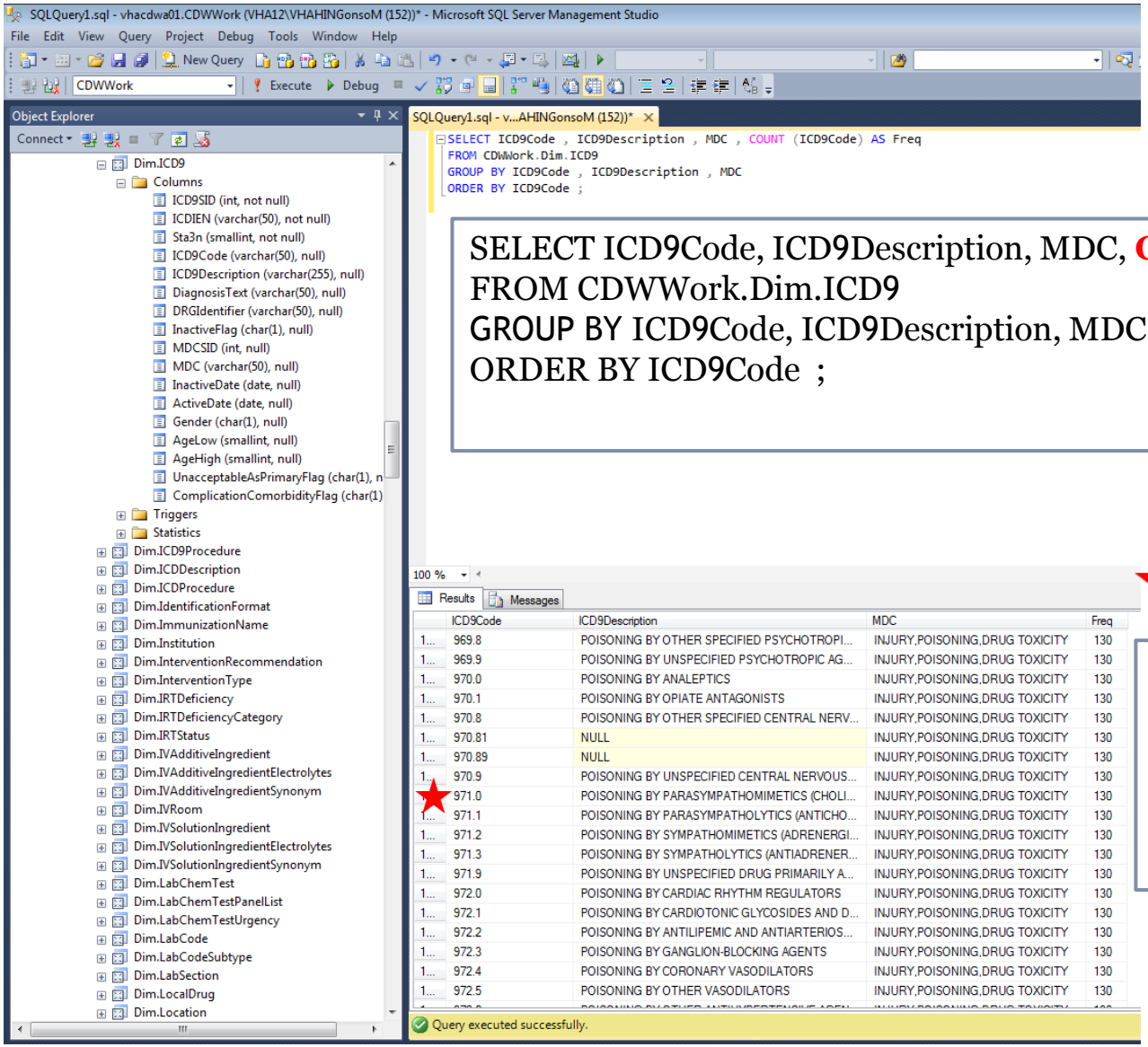
We can ask to see how many times each value repeats.

COUNT function

- The COUNT function can be added to the SELECT phrase in the query to create a new column.
- Because the new column is a sum (an aggregate measure), the programmer is obligated to include the GROUP BY clause in the query for all non-aggregate columns.
- The AS allows for the new column to be given a name by the programmer.

```
SELECT column1, column2, column3, COUNT (column1) AS Freq
FROM Database.Schema.View
GROUP BY column1, column2 , column3
ORDER BY column1 ;
```

COUNT

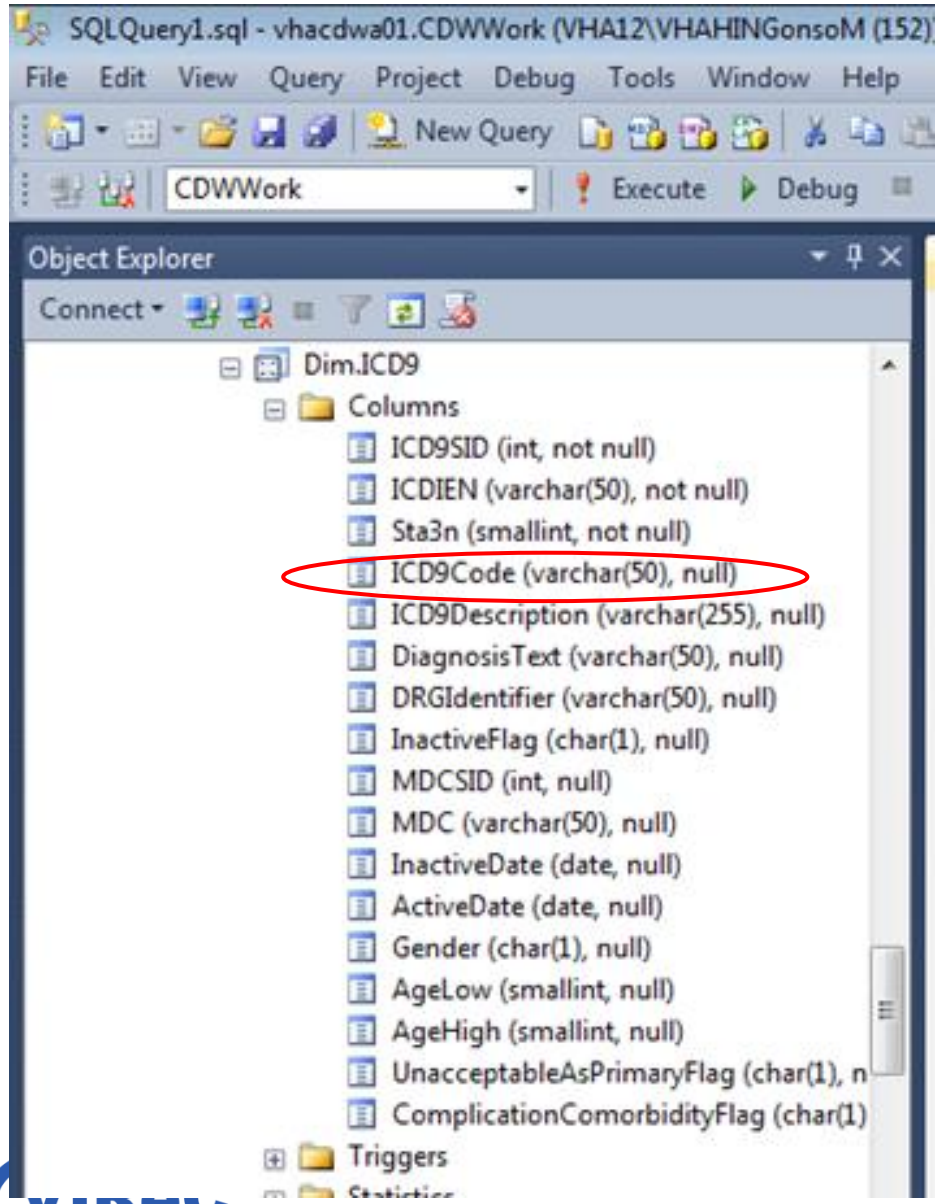


SELECT ICD9Code, ICD9Description, MDC, **COUNT (ICD9Code) AS Freq**
FROM CDWWork.Dim.ICD9
GROUP BY ICD9Code, ICD9Description, MDC
ORDER BY ICD9Code ;

Now we have the equivalent of a frequency table. The ICD code of 971.0 appears 130 times in the view Dim.ICD9, once for each VistA system.

WHERE statements

- The command WHERE allows a user to limit their search to include only a chosen subset of the data.
- Let's select the ICD9 code (309.81), indicating post-traumatic stress disorder.
- Although 309.81 looks like a number, you will need to know whether ICD9Code is a character or a numeric entry in CDW before you can write a WHERE statement.



VARCHAR = a string of variable length

(50) = the maximum number of characters

Other examples seen here:

Char = string of fixed length

Int = integer

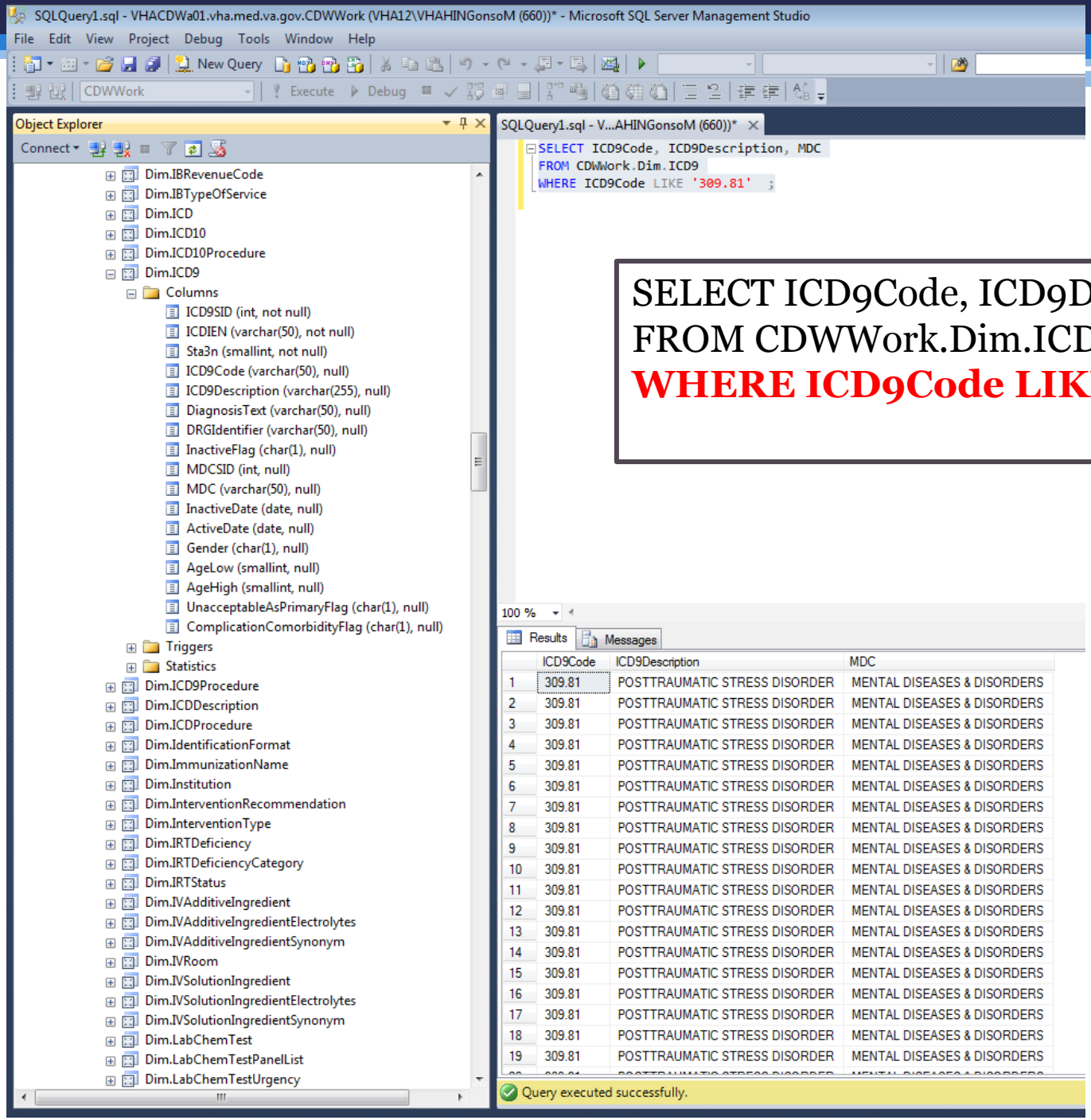
Smallint = integer

Date = a date

LIKE vs. =

- LIKE is typically used for string variables
- = is typically used for numeric variables
- So, because ICD9Code is a string, we will use LIKE in the WHERE phrase
- Always use single quotes around your characters
- Numeric values would not require quotes

```
SELECT column1, column2, column3
FROM Database.Schema.View
WHERE column1 LIKE '------'   ;  --note the value will go in the quotes
```




```
SELECT ICD9Code, ICD9Description, MDC
FROM CDWWork.Dim.ICD9
WHERE ICD9Code LIKE '309.81' ;
```

Note: In an actual research investigation, one might find it necessary to address varied entries with a wildcard such as '309%81' or similar logic.

Poll #2: About You

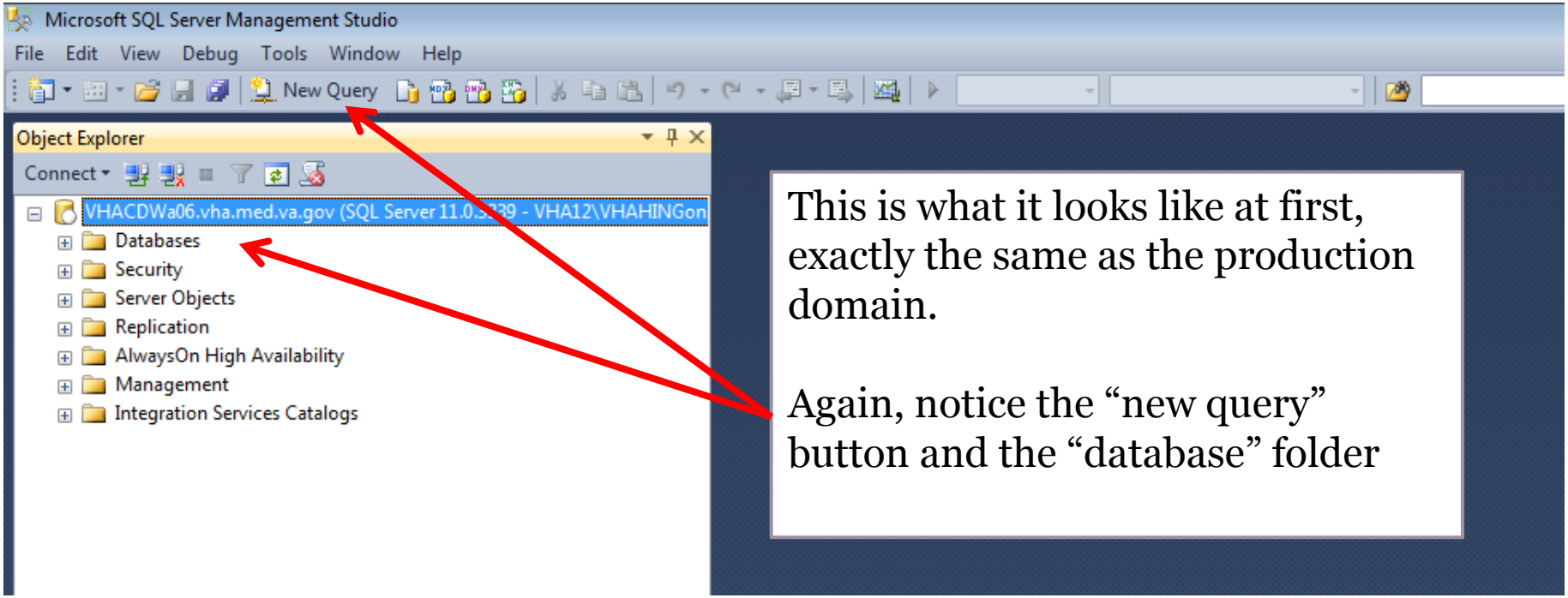
Which of the following best describes your role in the VA?

- ❑ Research Investigator / PI
- ❑ Data Manager / Analyst
- ❑ Project Coordinator
- ❑ Operations
- ❑ Other

In the Raw Domains

We will repeat the process with Dim.ICD9_Diagnosis_80

SQL Server Management Studio



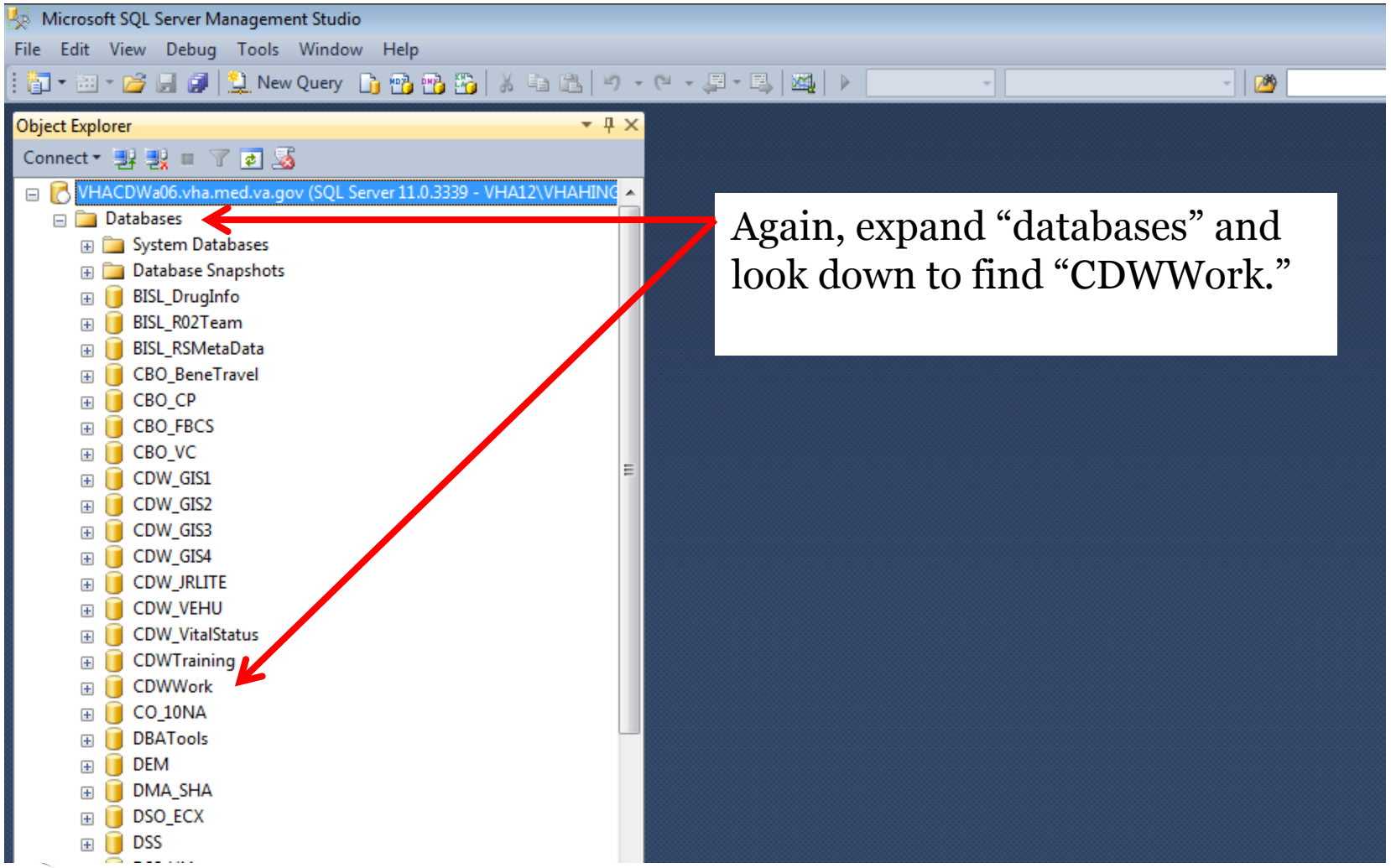
This is what it looks like at first, exactly the same as the production domain.

Again, notice the "new query" button and the "database" folder

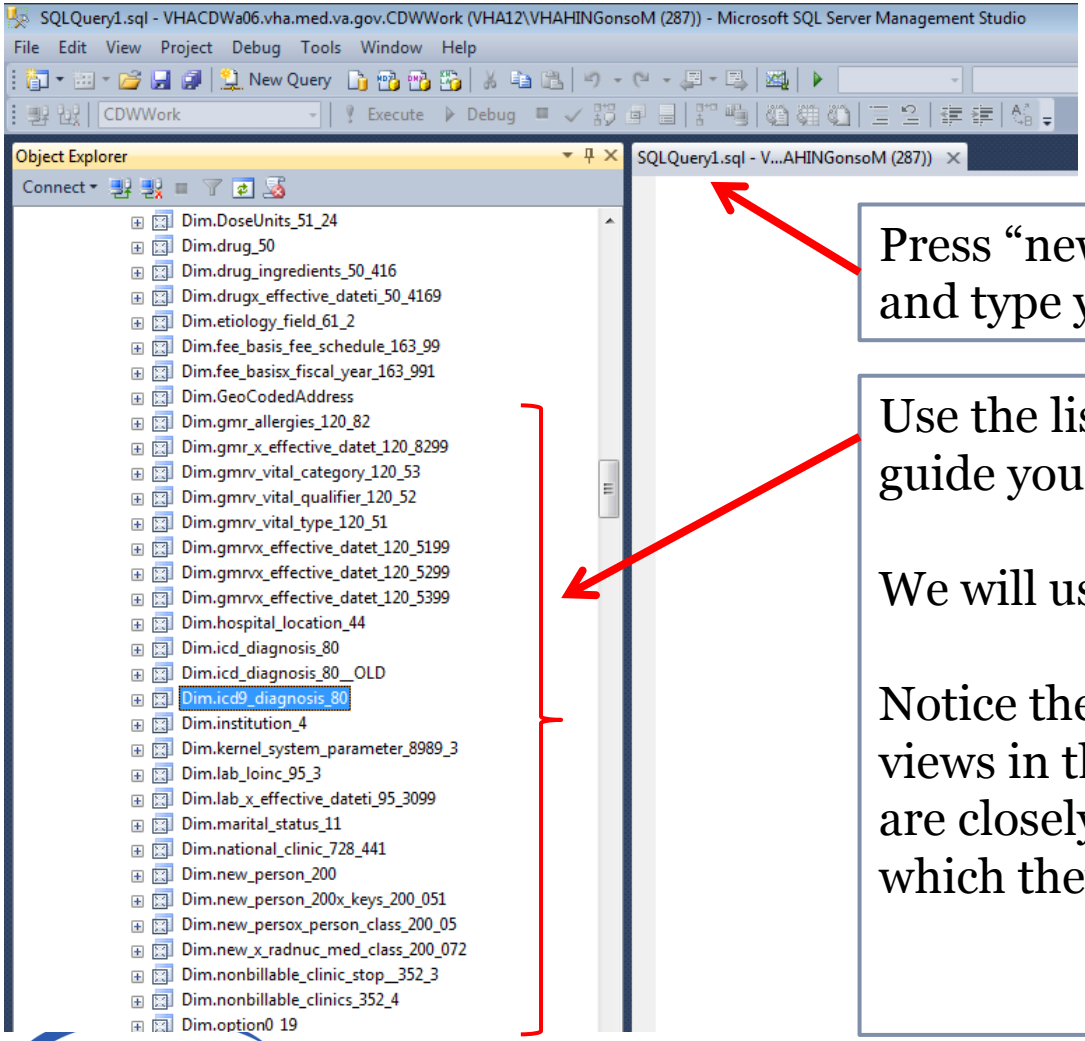
The only difference from the previous example is that I am now working off of a different server that holds the raw domains.

VIReC
RESEARCHERS' GUIDE TO VA DATA

Expand Databases and Scroll to CDWWork



New Query

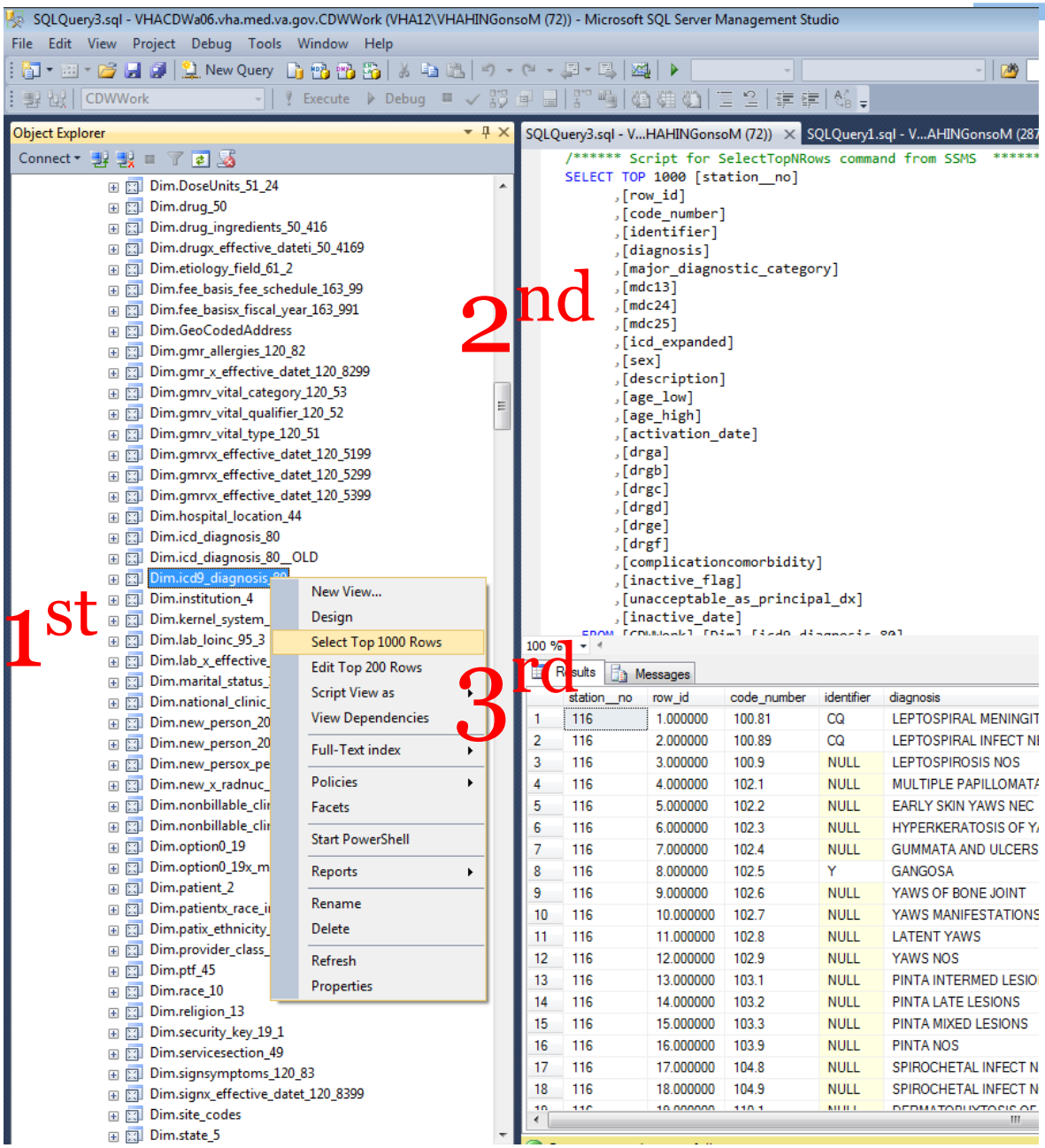


Press “new query” to open this window, and type your SQL query here.

Use the list of *views* in CDWWork to help guide your effort.

We will use Dim.icd9_diagnosis_80

Notice the difference in the naming styles; views in the raw domains have names that are closely related to the VistA’s files from which they originate.



Again,

1st Right click the view name for this drop-down menu and click “select top 1000 rows”

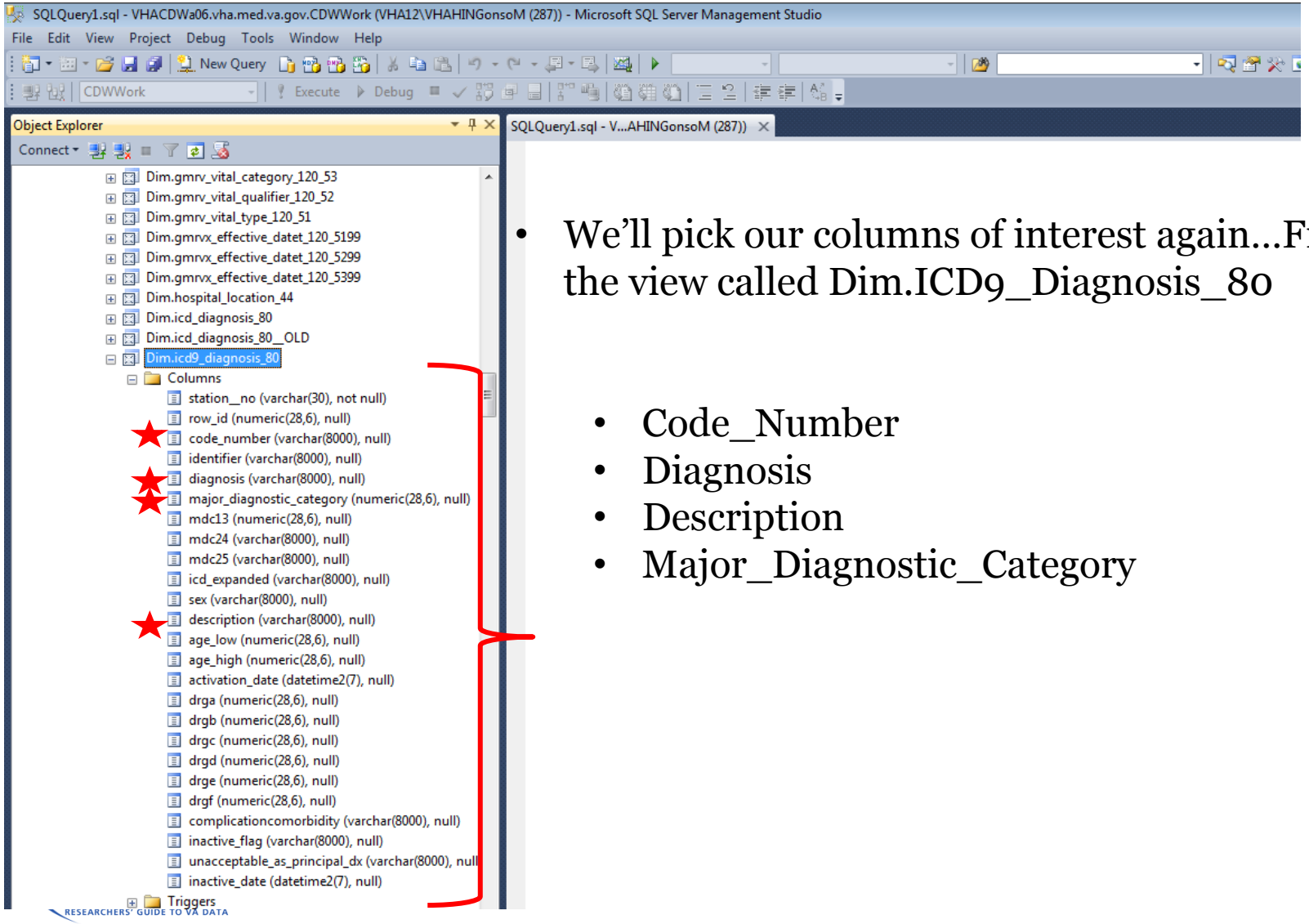
2nd, the query in the circle is automatically written

3rd the results automatically appear in the bottom window

SELECT and FROM

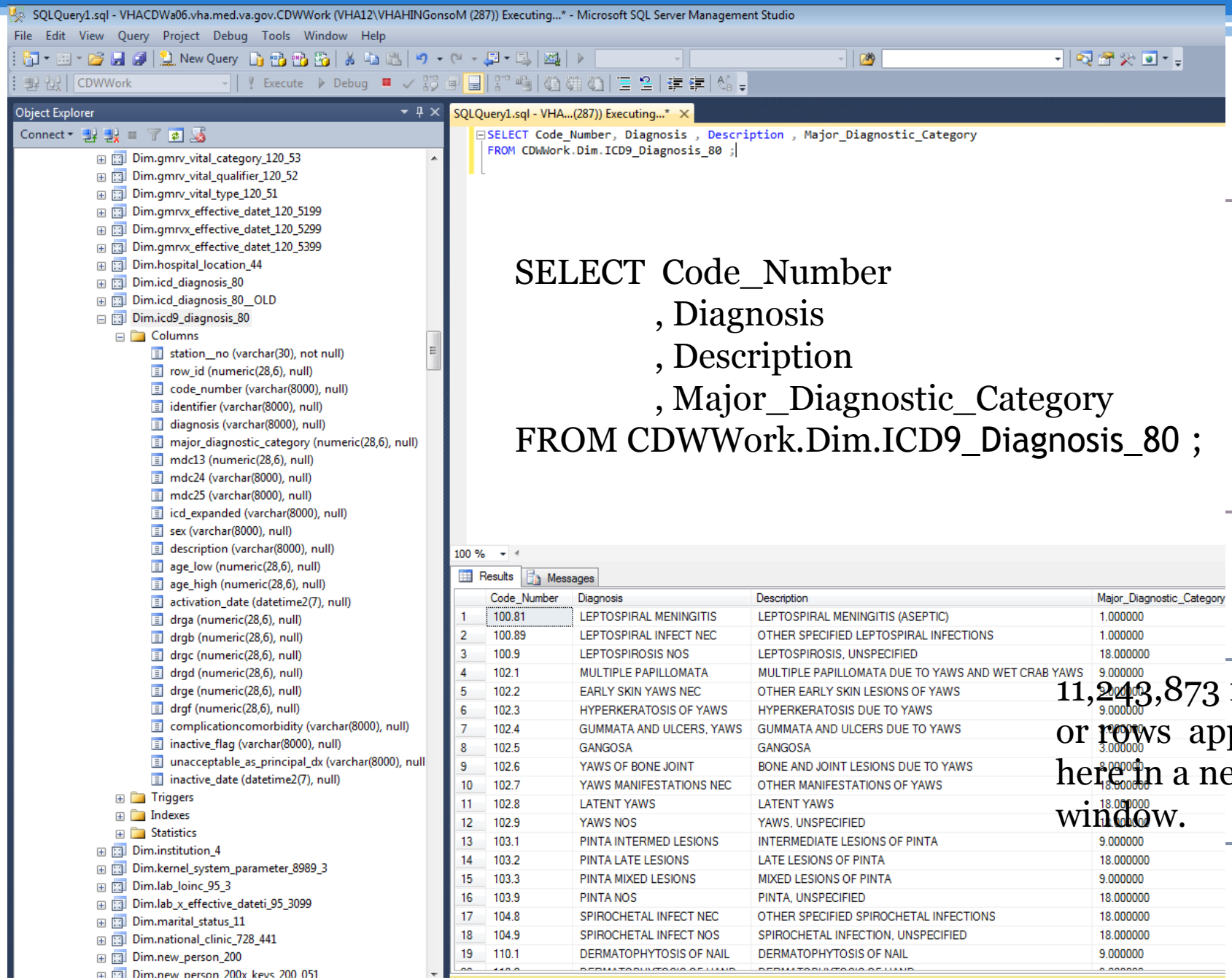
- SELECT your columns (a.k.a, variables)
  - SELECT column1, column2, column3
- FROM your *view* of choice
  - From database.schema.view_name
- Don't forget to end with a semi-colon

SELECT & FROM



- We'll pick our columns of interest again…From the view called Dim.ICD9_Diagnosis_80

  - Code_Number
  - Diagnosis
  - Description
  - Major_Diagnostic_Category



SELECT Code_Number
, Diagnosis
, Description
, Major_Diagnostic_Category
FROM CDWWork.Dim.ICD9_Diagnosis_80 ;

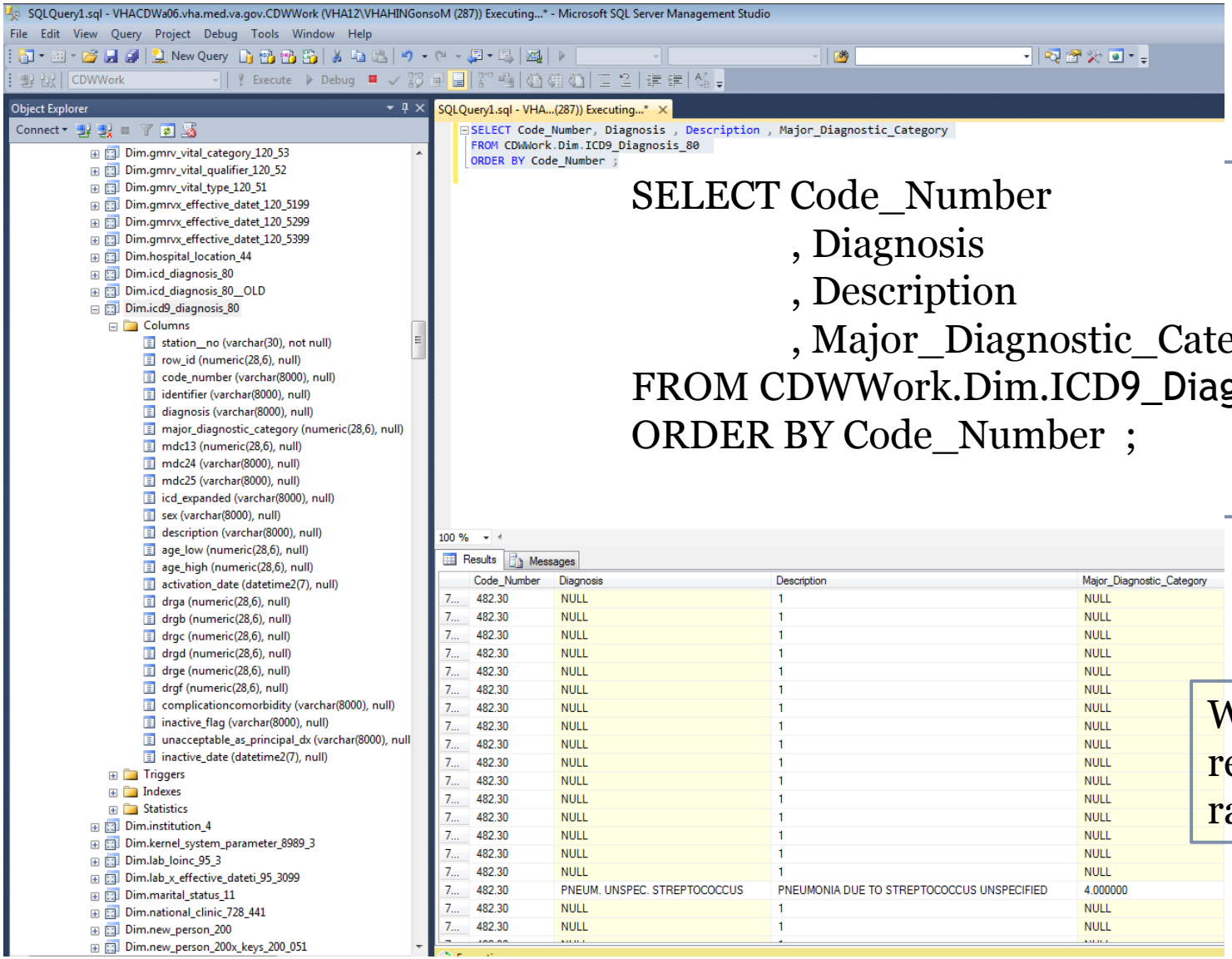
11,243,873 results or rows appear here in a new window.

ORDER BY

- The ORDER BY command will allow us to request the results in order by our chosen column(s).
- Remember that ORDER BY must be the last command in any query.

```
SELECT column1, column2, column3
FROM Database.Schema.View
ORDER BY column1 ;
```

ORDER BY




```
SELECT Code_Number
     , Diagnosis
     , Description
     , Major_Diagnostic_Category
FROM CDWWork.Dim.ICD9_Diagnosis_80
ORDER BY Code_Number  ;
```

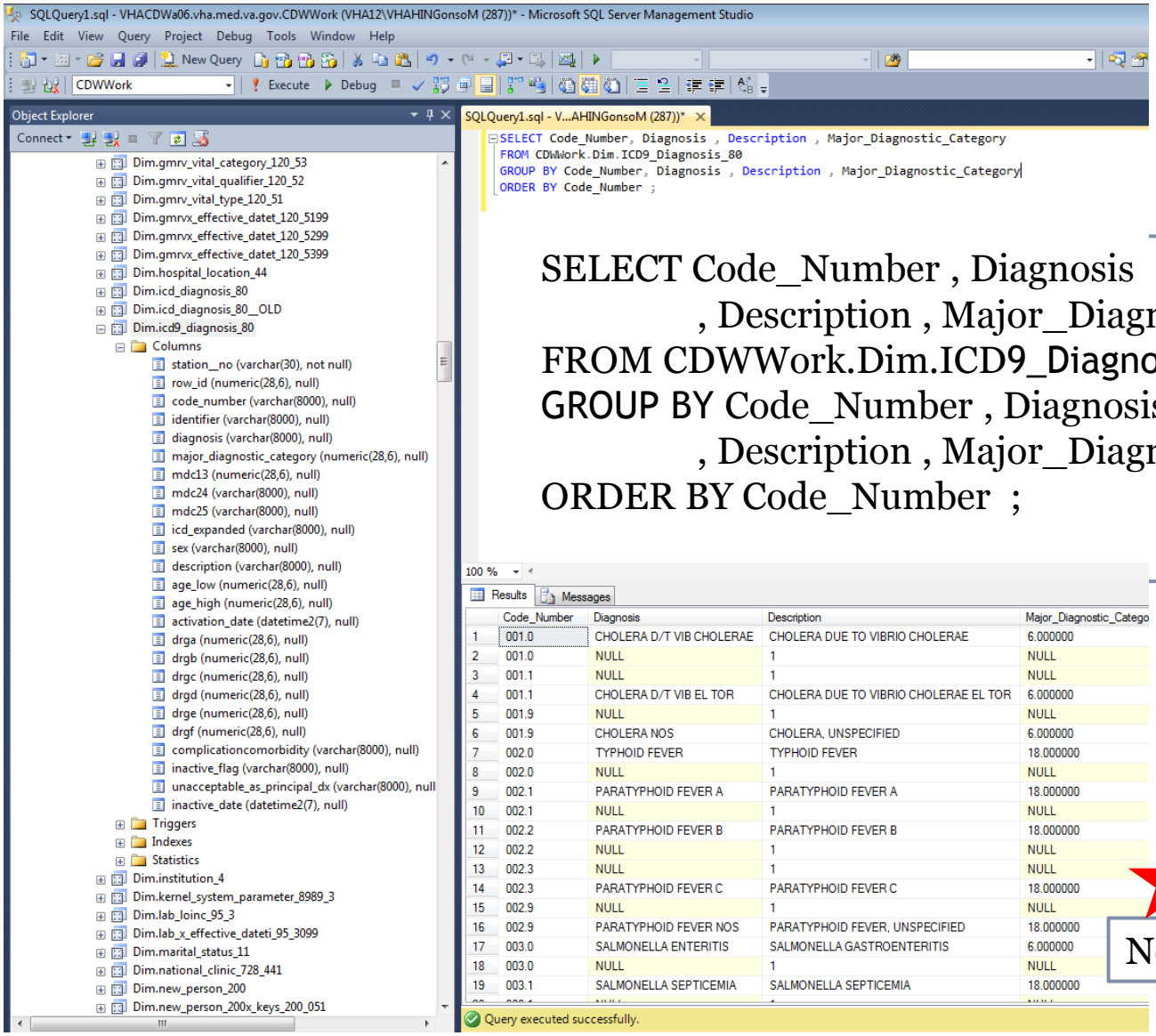
We see the same repetition in the raw domain.

GROUP BY

- Again, we will collapse the repeated entries using the GROUP BY command.
- Remember:
  - ORDER BY must be the last command
  - Every column that is not an aggregate measure must be included in the GROUP BY clause

```
SELECT column1, column2, column3
FROM Database.Schema.View
GROUP BY column1, column2 , column3
ORDER BY column1 ;
```

GROUP BY




```
SELECT Code_Number , Diagnosis
      , Description , Major_Diagnostic_Code
FROM CDWWork.Dim.ICD9_Diagnosis_80
GROUP BY Code_Number , Diagnosis
      , Description , Major_Diagnostic_Code
ORDER BY Code_Number  ;
```

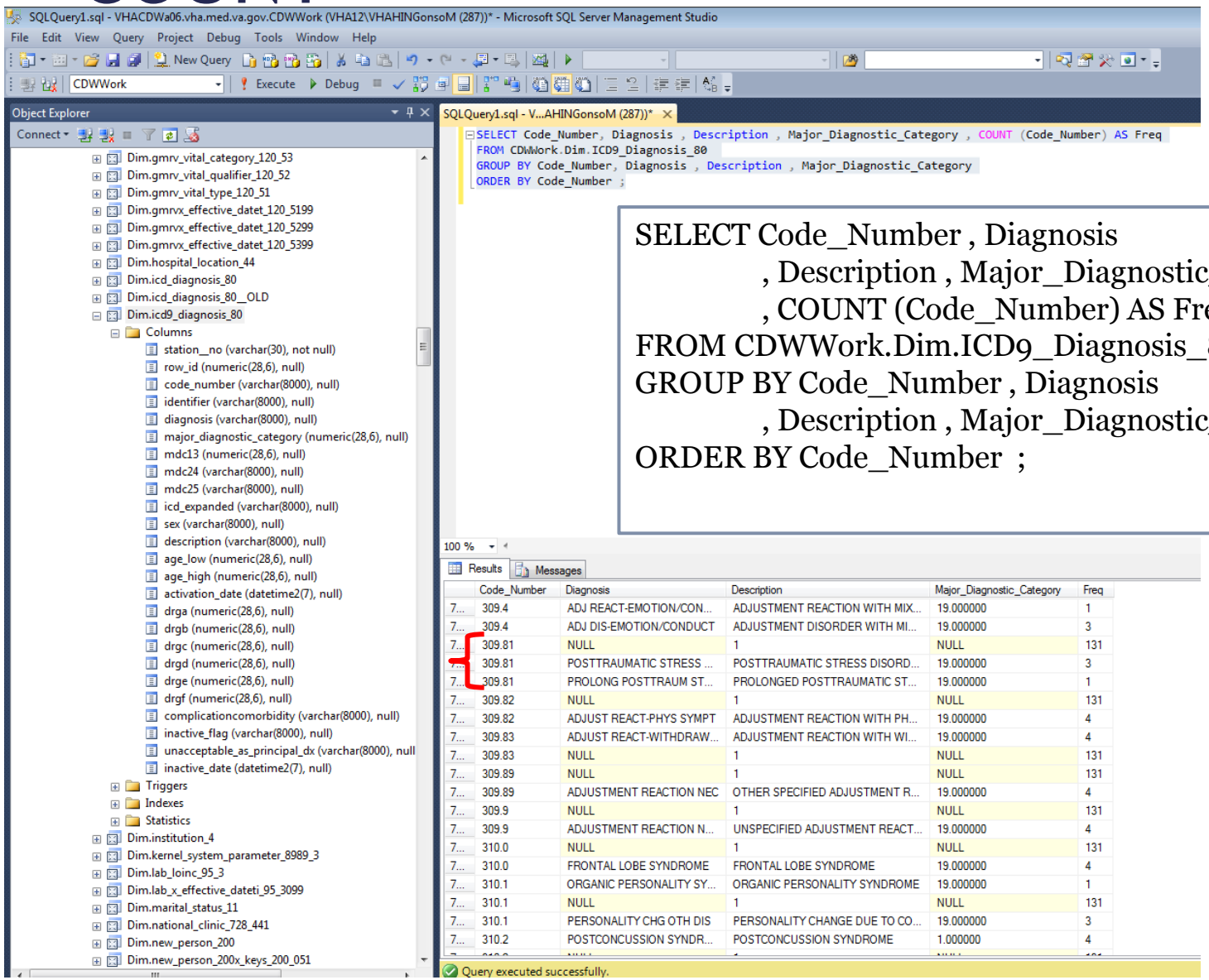
Now there are 101,438 rows.

COUNT function

- Again, we can use the COUNT function to create a frequency table.
- Using AS we can assign the name “freq” to the count column.
- Remember to use the GROUP BY clause for all non-aggregate columns.

```
SELECT column1, column2, column3, COUNT (column1) AS Freq
FROM Database.Schema.View
GROUP BY column1, column2 , column3
ORDER BY column1 ;
```

COUNT

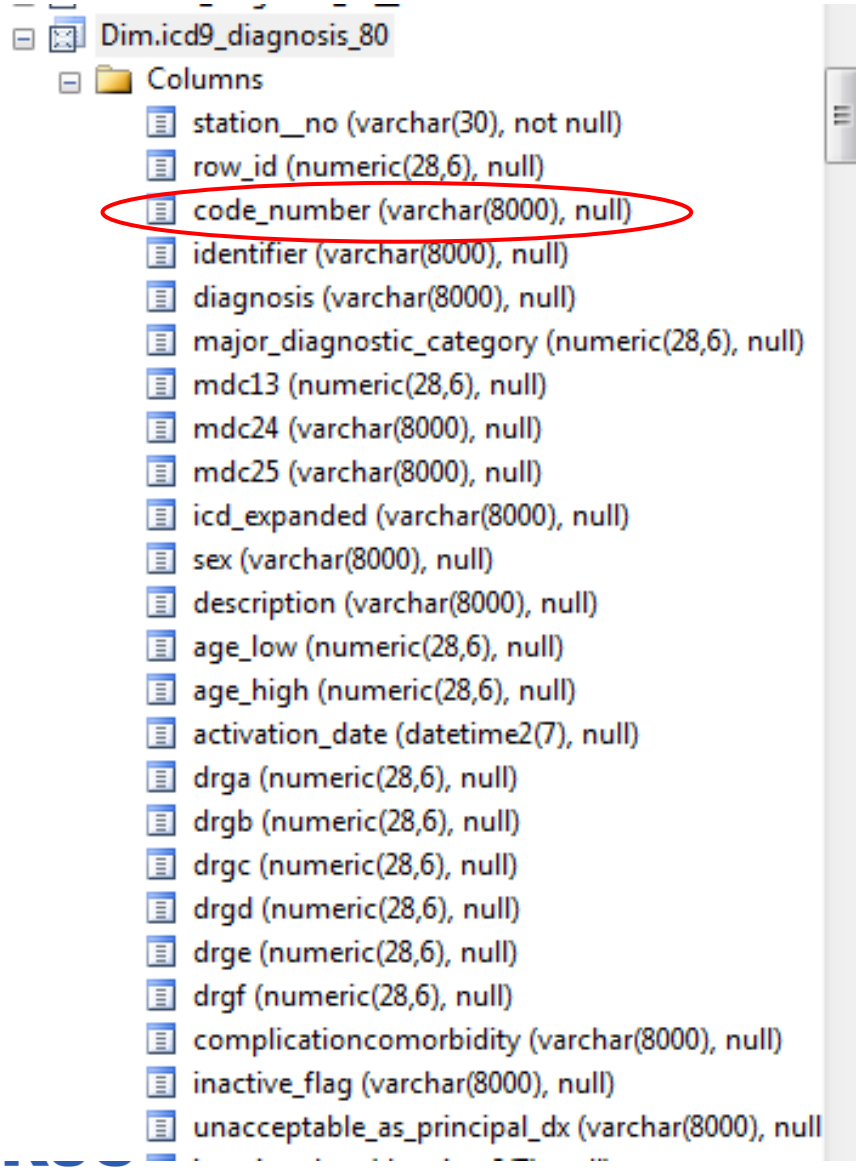



```
SELECT Code_Number , Diagnosis
            , Description , Major_Diagnostic_Code
            , COUNT (Code_Number) AS Freq
FROM CDWWork.Dim.ICD9_Diagnosis_80
GROUP BY Code_Number , Diagnosis
            , Description , Major_Diagnostic_Code
ORDER BY Code_Number  ;
```

Now we have the equivalent of a frequency table.

WHERE statements

- We'll use the WHERE again to choose a subset of the data.
- We'll select the Code_Number (309.81) for PTSD again.
- Again, we'll check to see whether Code_Number is a character or a numeric before writing our WHERE statement.



VARCHAR = a string of variable length

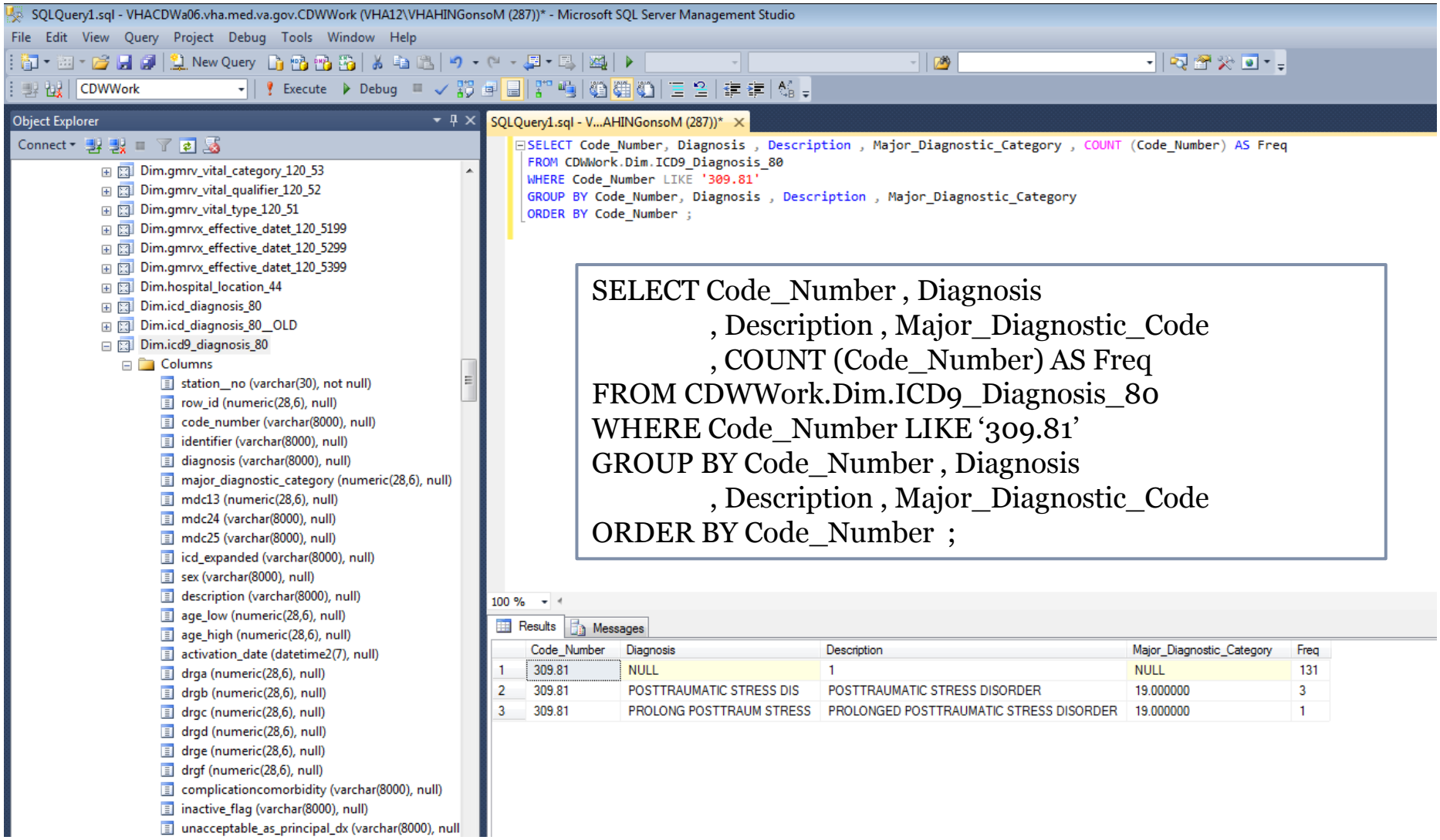
(8000) = the maximum number of characters

Therefore, we will tend use the LIKE function rather than an equal sign in our WHERE clause.

Pulling it all together in one query

- SELECT, COUNT, AS, WHERE, LIKE, GROUP BY and ORDER BY
- Remember the rules!

```
SELECT column1, column2, column3, COUNT (columnX) AS NewName
FROM Database.Schema.View
WHERE column1 LIKE '------'     --note the value will go in the quotes
GROUP BY column1, column2...    -- include all non-aggregate columns
ORDER BY column1...             -- name column(s) to order by
;                               -- don't forget to end with a ;
```




```
SELECT Code_Number , Diagnosis
            , Description , Major_Diagnostic_Code
            , COUNT (Code_Number) AS Freq
FROM CDWWork.Dim.ICD9_Diagnosis_80
WHERE Code_Number LIKE '309.81'
GROUP BY Code_Number , Diagnosis
            , Description , Major_Diagnostic_Code
ORDER BY Code_Number  ;
```

Summary/Conclusion

- These basic commands (SELECT, FROM, WHERE, GROUP BY and ORDER BY) form the core of most SQL queries.
- Using this basic structure, you should be able to explore the content of any view in CDW.
- The goal of the next cyberseminar will be to show you how to combine columns from multiple views; in other words, it will step the new user through a JOIN in SQL.

VIReC
RESEARCHERS' GUIDE TO VA DATA

Contact Information

Margaret Gonsoulin, PhD

VIReC@va.gov

708-202-2413

VIReC
RESEARCHERS' GUIDE TO VA DATA

Questions?