

Working Knowledge Series

SQL for Programmers

Tom Davidson

Cleindori Consulting LLC

Tom.Davidson@CleindoriConsulting.com

Ground Rules

- Please set cell phones to 'stun'
- Feel free to ask questions as we go along, I may defer to later in the session or offline.
- Not all slides in handout may be included, they are included for your reference only.
- Go to web site for current version of presentation

Cleindori Consulting LLC

- Specialties:
 - IBM i
 - Performance
 - Database Administration
 - Application design and implementation
 - Contract Programming
 - Administration
 - JDE EnterpriseOne and OneWorld CNC
 - Application integration
- www.CleindoriConsulting.com

Speaker Bio

- 26+ years on the IBM i, developer the whole time
- IBM ILE Certification
- 10+ years of performance tuning
- Now consulting, specialties include JD Edwards, performance tuning, sizing, and contract programming

Agenda

- Terminology
- Simple SQL
- Where Clause
- Useful functions
- Tips and Techniques
- Gotcha's

Terminology

- Schema – Library
- Table – File
- Row – Record
- Column – Field
- Index – Logical file
- View – Logical file (field selection, join, etc, no index)
- Primary Key – Unique key on Physical
- Foreign Key – Field (column) in database that is a Primary Key in another file

Simple SQL

- SQL has 4 basic statements
 - SELECT – Display
 - UPDATE – Change records
 - INSERT – Add records
 - DELETE – Delete records
- SQL 3 statements for tables
 - CREATE TABLE - Create
 - ALTER TABLE - Change
 - DROP TABLE - Delete

Simple SQL

- Name format for files is normally library.file instead of library/file
- Name format can be changed via JDBC Settings/Format/Naming Convention
 - *SYS = lib/file
 - *SQL = lib.file

The WHERE Clause

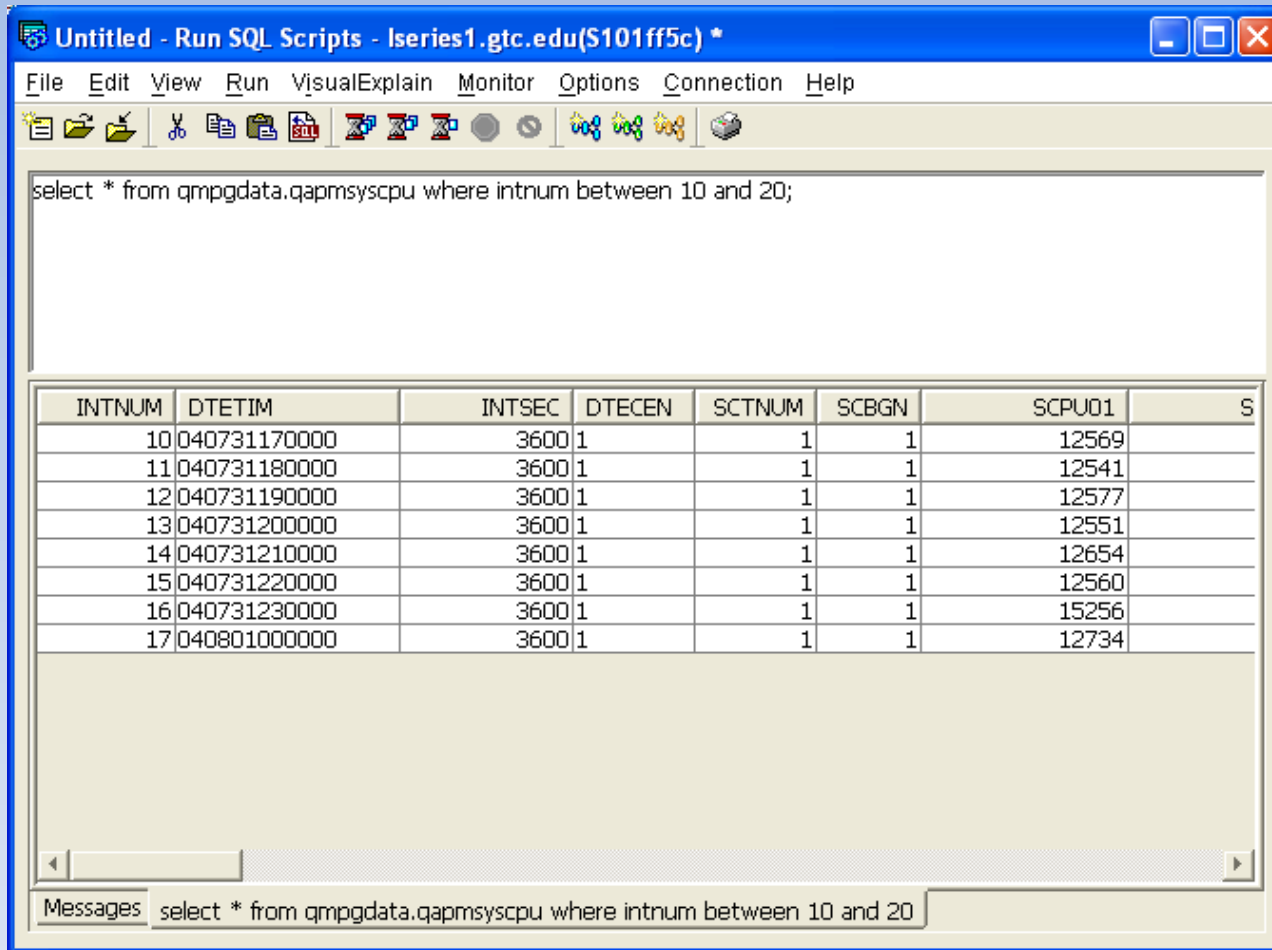
- Most useful part of SQL
 - Can be used with SELECT, DELETE, and UPDATE
 - Besides the normal =, <>, etc. There are 3 really useful keywords
 - BETWEEN – Between these values
 - IN – Allow a selection list
 - LIKE – Allows wildcards
 - Items in a where clause can be negated with the use of NOT

The WHERE Clause

Between

- Syntax: where field1 between 5 and 10
- Syntax: where field1 between "C" and "Q"
- The field type and the values must be compatible

The WHERE Clause BETWEEN



The screenshot shows a window titled "Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c) *". The menu bar includes File, Edit, View, Run, VisualExplain, Monitor, Options, Connection, and Help. The toolbar contains various icons for file operations and execution. The main text area contains the SQL query: `select * from qmpgdata.qapmsyscpu where intrnum between 10 and 20;`. Below the query, a table displays the results of the query. The table has 8 columns: INTNUM, DTETIM, INTSEC, DTECEN, SCTNUM, SCBGN, SCPU01, and S. The data rows show values for each column, with INTNUM ranging from 10 to 17. A Messages pane at the bottom shows the same query text.

INTNUM	DTETIM	INTSEC	DTECEN	SCTNUM	SCBGN	SCPU01	S
10	040731170000	3600	1	1	1	12569	
11	040731180000	3600	1	1	1	12541	
12	040731190000	3600	1	1	1	12577	
13	040731200000	3600	1	1	1	12551	
14	040731210000	3600	1	1	1	12654	
15	040731220000	3600	1	1	1	12560	
16	040731230000	3600	1	1	1	15256	
17	040801000000	3600	1	1	1	12734	

Messages select * from qmpgdata.qapmsyscpu where intrnum between 10 and 20

Useful Functions

IN

- Allows you to specify a list of values
- Syntax: where field1 in (1,3,5,7,9)
- May be used with a sub-select

The WHERE Clause IN

Untitled - Run SQL Scripts - Iseries1.gtc.edu{...}

File Edit View Run VisualExplain Monitor Options Connection Help

```
select * from qmpgdata.qapmsyscpu where intnum in (1,3,5,7,9);
```

INTNUM	DTETIM	INTSEC	DTECEN	SK
1	040731080000	1336	1	
3	040731100000	3600	1	
5	040731120000	3600	1	
7	040731140000	3600	1	
9	040731160000	3600	1	

select * from qmpgdata.qapmsyscpu where intnum in (1,3,5,7,9)

Messages

Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c) *

File Edit View Run VisualExplain Monitor Options Connection Help

```
select * from qmpgdata.qapmdisk where dsdrn in ('DD003','DD001');
```

DSDRN	DSSCAN	DSBLKR	DSBLKW
DD003	0	1776	2432
DD001	0	2456	2560
DD003	0	90712	248560
DD001	0	81000	138776
DD003	0	2424	6184
DD001	0	792	1624
DD003	0	824	2504
DD001	0	184	1904
DD003	0	552	2528
DD001	0	144	1856
DD003	0	720	2384
DD001	0	248	1808
DD003	0	216	2200
DD001	0	336	2192
DD003	0	136	2272
DD001	0	120	1760
DD003	0	1016	4020

Messages select * from qmpgdata.qapmdisk where dsdrn in ('DD003','DD001')

Useful Functions IN (subselect)

The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The menu bar includes File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, and Help. The toolbar contains various icons for file operations and execution. The main text area contains the following SQL query:

```
select customer_id, first_name, last_name, state from egl00.customer
       where state in (select state_abbrev from egl00.statetable where state_abbrev < 'F');
```

Below the query, a table displays the results of the execution:

CUSTOMER_ID	FIRST_NAME	LAST_NAME	STATE
4	Francine	Liebowitz	CA
5	Ornette	Coleman	CT
7	Martin	St. Louis	CT

At the bottom of the window, a status bar shows the text: "select customer_id, first_name, last_name, state from egl00.customer where state in (select state_abbrev from egl00.statetable where" and "Messages".

The WHERE Clause

LIKE

- Finds patterns similar to the one you enter
- Syntax: where field1 like “%Tom_D%”
- “%” matches anything
- “_” matches just one character

The WHERE Clause

LIKE

Untitled - Run SQL Scripts - lseries1.gtc.edu(...)

File Edit View Run Visual Explain Monitor Options Connector Help

```
select * from egl00.customer where first_name like 'Fr%';
```

CUSTOMER_ID	FIRST_NAME
1	Fred
4	Francine

select * from egl00.customer where first_name like 'Fr%'

Messages

Untitled - Run SQL Scripts - lseries1.gtc.edu(...)

File Edit View Run Visual Explain Monitor Options Connector Help

```
select * from egl00.customer where first_name like '%r%';
```

CUSTOMER_ID	FIRST_NAME
1	Fred
4	Francine
5	Ornette
7	Martin
8	Sergey

select * from egl00.customer where first_name like '%r%'

Messages

The WHERE Clause LIKE

Untitled - Run SQL Scripts - lseries1.gtc.edu(...)

File Edit View Run Visual Explain Monitor Options Connector Help

```
select * from egl00.customer where first_name like '_r%';
```

CUSTOMER_ID	FIRST_NAME
1	Fred
4	Francine
5	Ornette

select * from egl00.customer where first_name like '_r%'

Messages

Untitled - Run SQL Scripts - lseries1.gtc.edu(...)

File Edit View Run Visual Explain Monitor Options Connector Help

```
select * from egl00.customer where first_name like '%e';
```

CUSTOMER_ID	FIRST_NAME
-------------	------------

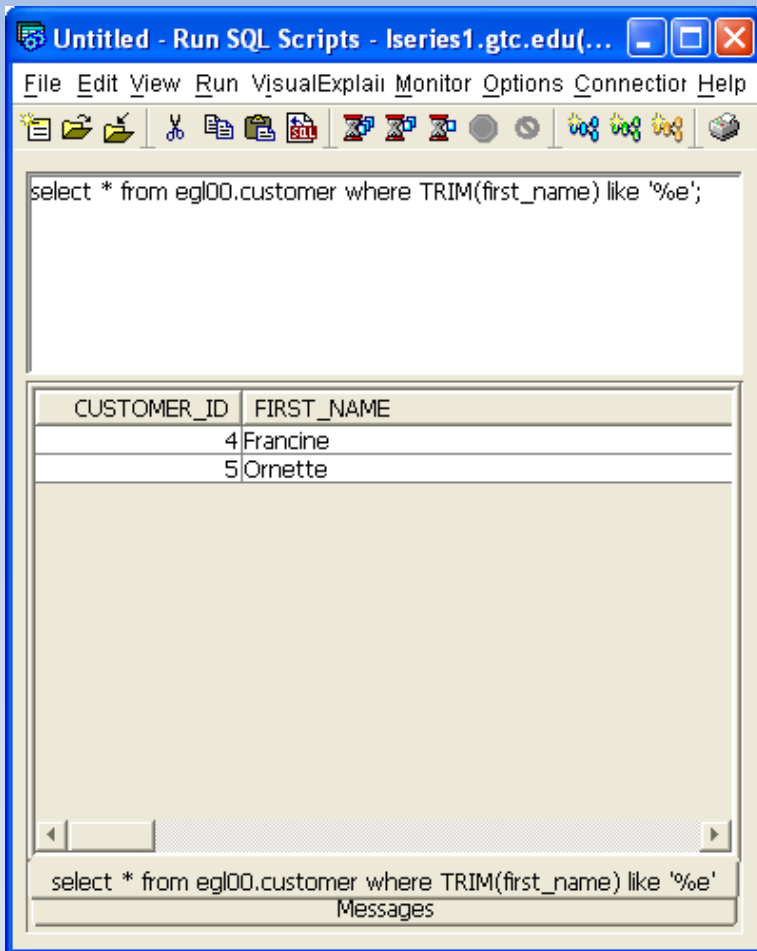
select * from egl00.customer where first_name like '%e'

Messages

Unexpected result. Why?

The WHERE Clause LIKE

Field has trailing blanks. This will do what we want



The screenshot shows a window titled "Untitled - Run SQL Scripts - Iseries1.gtc.edu(...)". The menu bar includes File, Edit, View, Run, Visual Explain, Monitor, Options, Connector, and Help. The toolbar contains various icons for file operations and execution. The main text area contains the SQL query: `select * from egl00.customer where TRIM(first_name) like '%e';`. Below the query, a table displays the results:

CUSTOMER_ID	FIRST_NAME
4	Francine
5	Ornette

At the bottom of the window, there is a status bar with the text: `select * from egl00.customer where TRIM(first_name) like '%e'` and "Messages".

Useful Functions

Useful Functions

- Two groups
 - Aggregate – summarizes data
 - Scaler – Uses only 1 record
- Restrictions:
 - If you use a BIF, you must list the fields you want
 - Aggregate functions require you to use either all the records or the 'GROUP BY' syntax

Useful Functions

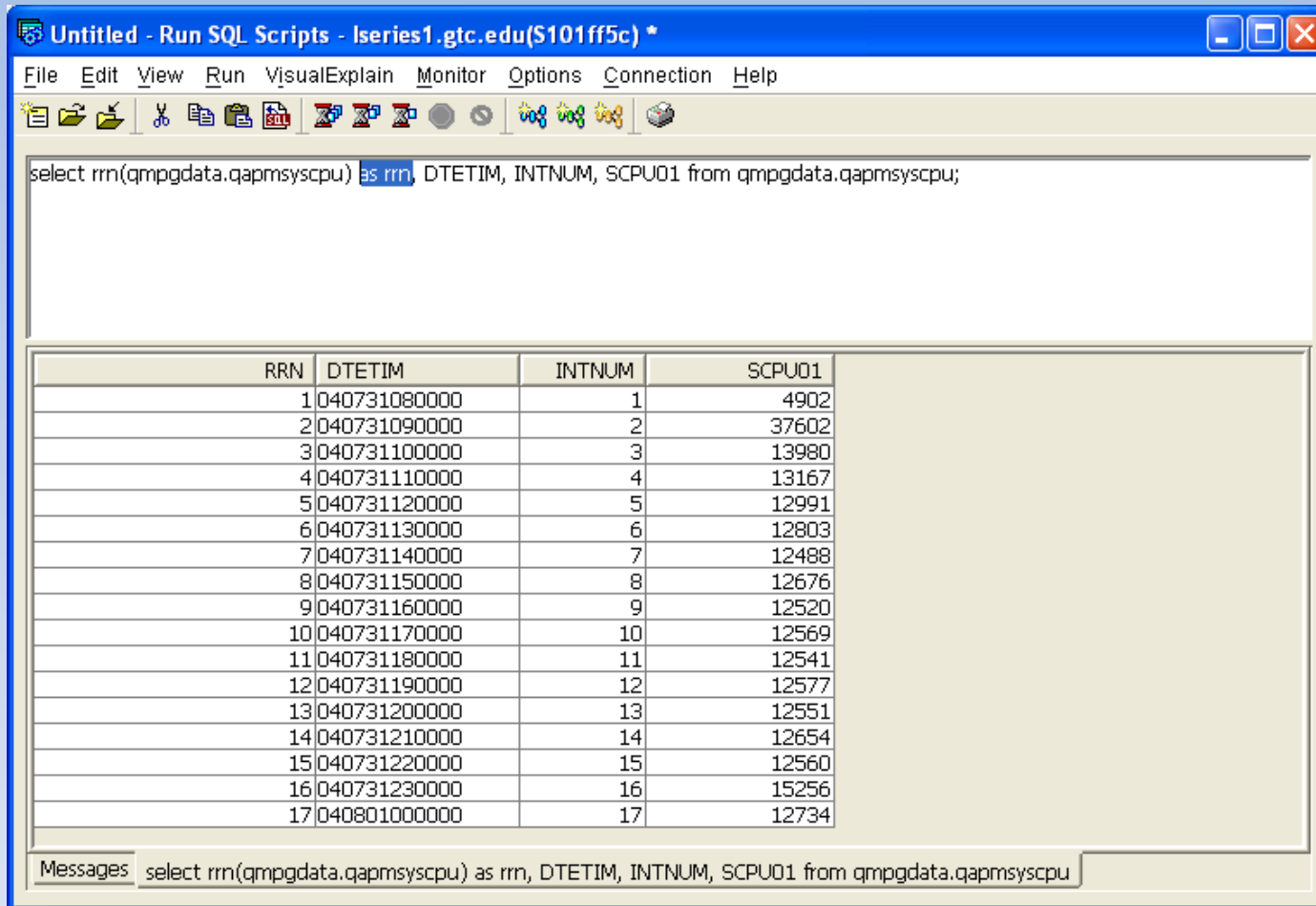
RRN

- Retrieves RRN of a record (IBM i only)
- Allows you to get a unique identifier even on files that do not have a unique key
- Sometimes the only way to select a single record

- Syntax: `rrn(library.filename)`

Useful Functions

RRN



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window contains a menu bar with "File", "Edit", "View", "Run", "Visual Explain", "Monitor", "Options", "Connection", and "Help". Below the menu bar is a toolbar with various icons. The main text area contains the following SQL query:

```
select rrn(qmpgdata.qapmsyscpu) as rrn, DTETIM, INTNUM, SCPU01 from qmpgdata.qapmsyscpu;
```

Below the query, a table displays the results of the query. The table has four columns: RRN, DTETIM, INTNUM, and SCPU01. The data is as follows:

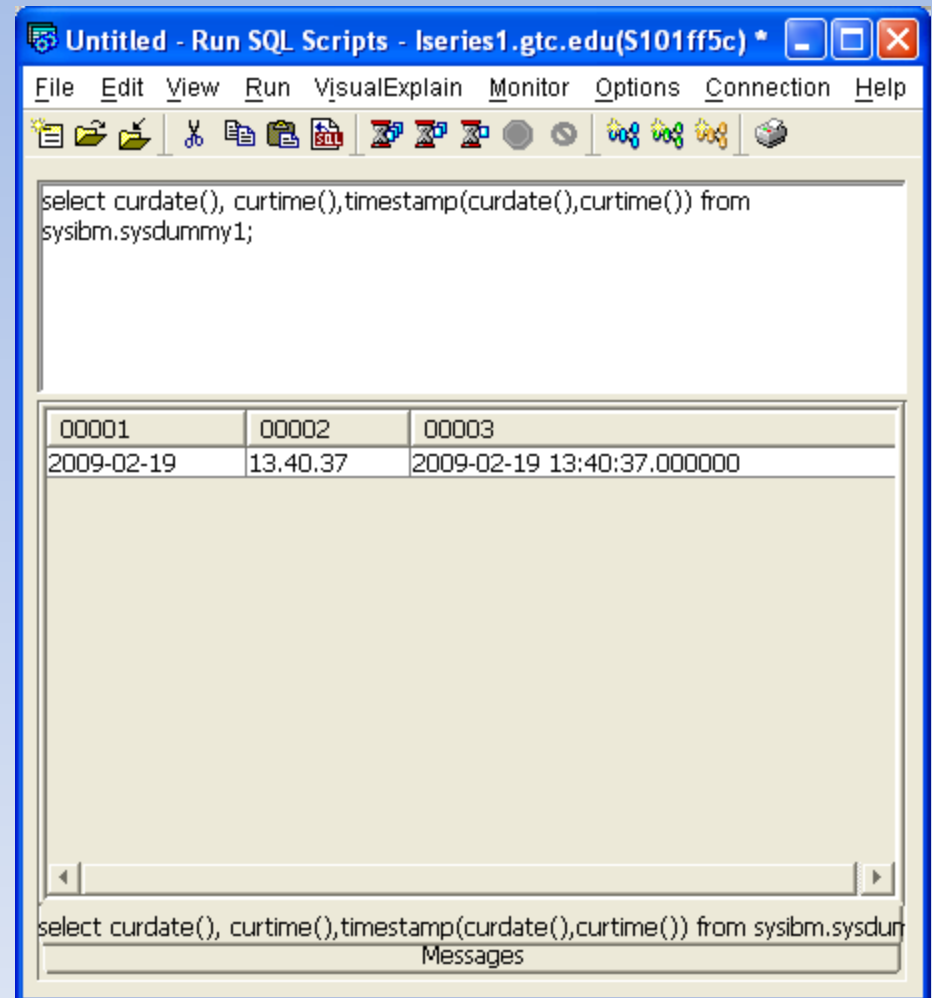
RRN	DTETIM	INTNUM	SCPU01
1	040731080000	1	4902
2	040731090000	2	37602
3	040731100000	3	13980
4	040731110000	4	13167
5	040731120000	5	12991
6	040731130000	6	12803
7	040731140000	7	12488
8	040731150000	8	12676
9	040731160000	9	12520
10	040731170000	10	12569
11	040731180000	11	12541
12	040731190000	12	12577
13	040731200000	13	12551
14	040731210000	14	12654
15	040731220000	15	12560
16	040731230000	16	15256
17	040801000000	17	12734

At the bottom of the window, a "Messages" pane shows the same SQL query: "select rrn(qmpgdata.qapmsyscpu) as rrn, DTETIM, INTNUM, SCPU01 from qmpgdata.qapmsyscpu;"

Useful Functions

CURDATE/CURTIME/TIMESTAMP

- Gets current date and time
- Can be used for INSERTs and UPDATEs as timestamps



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window contains a menu bar with "File", "Edit", "View", "Run", "Visual Explain", "Monitor", "Options", "Connection", and "Help". Below the menu bar is a toolbar with various icons. The main area of the window displays a SQL query: `select curdate(), curtime(),timestamp(curdate(),curtime()) from sysibm.sysdummy1;`. Below the query, the results are displayed in a table with three columns. The first column contains the value "00001", the second column contains "00002", and the third column contains "00003". The second row of the table shows the results of the query: "2009-02-19", "13.40.37", and "2009-02-19 13:40:37.000000". At the bottom of the window, there is a "Messages" pane with the text "select curdate(), curtime(),timestamp(curdate(),curtime()) from sysibm.sysdur".

00001	00002	00003
2009-02-19	13.40.37	2009-02-19 13:40:37.000000

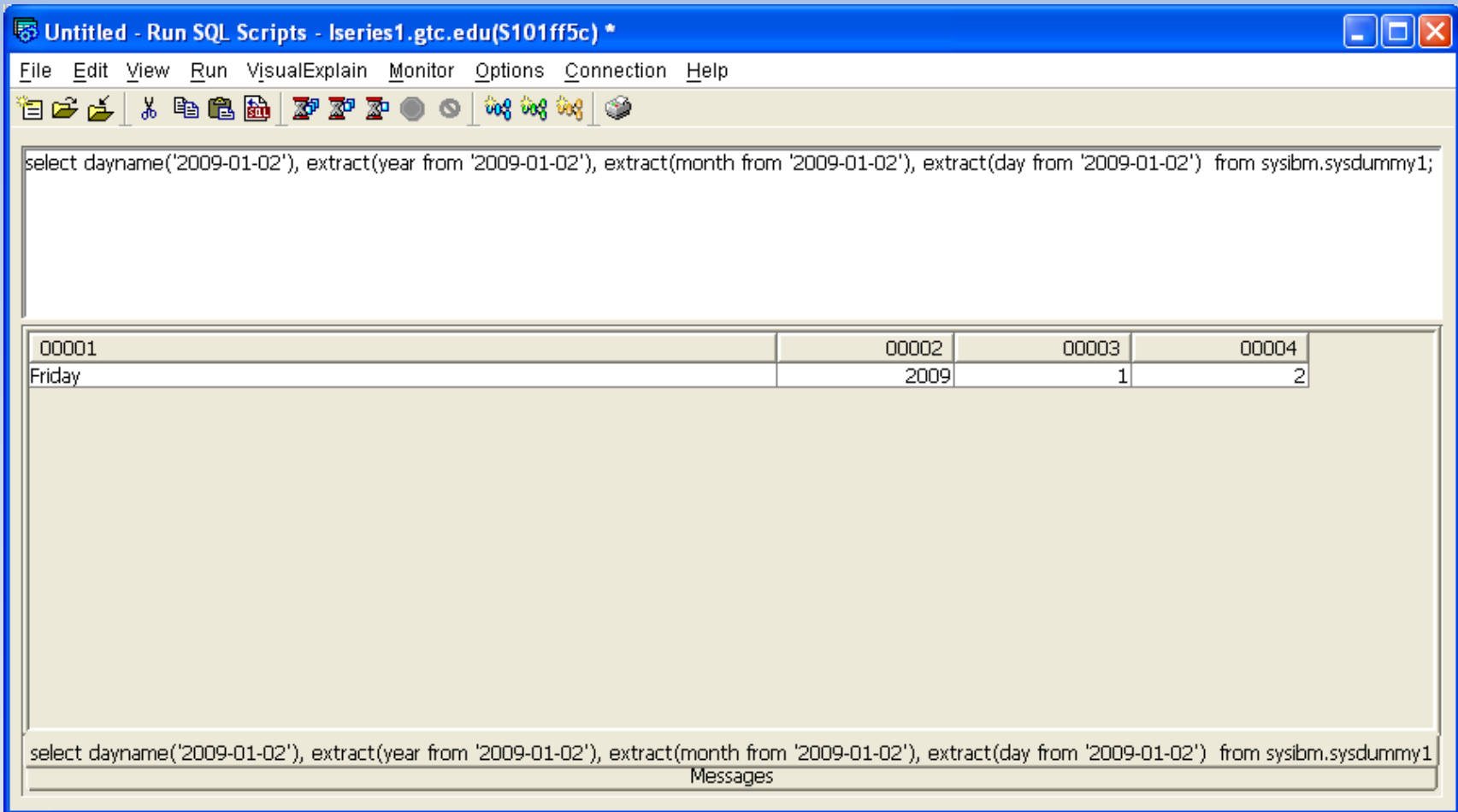
Useful Functions

Date functions

- **DAYNAME** – Gets day of week
 - Syntax: `DAYNAME(date value)`
- **EXTRACT** – Gets portion of date
 - Syntax: `EXTRACT(YEAR FROM datevalue)`
 - Syntax: `EXTRACT(MONTH FROM datevalue)`
 - Syntax: `EXTRACT(DAY FROM datevalue)`

Useful Functions

Date functions



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window contains a menu bar with "File", "Edit", "View", "Run", "VisualExplain", "Monitor", "Options", "Connection", and "Help". Below the menu bar is a toolbar with various icons. The main area of the window displays a SQL query: `select dayname('2009-01-02'), extract(year from '2009-01-02'), extract(month from '2009-01-02'), extract(day from '2009-01-02') from sysibm.sysdummy1;`. Below the query, the results are displayed in a table with four columns. The first column contains the day name "Friday", and the other three columns contain the year "2009", the month "1", and the day "2". At the bottom of the window, the same SQL query is repeated, followed by the word "Messages".

```
select dayname('2009-01-02'), extract(year from '2009-01-02'), extract(month from '2009-01-02'), extract(day from '2009-01-02') from sysibm.sysdummy1;
```

00001	00002	00003	00004
Friday	2009	1	2

```
select dayname('2009-01-02'), extract(year from '2009-01-02'), extract(month from '2009-01-02'), extract(day from '2009-01-02') from sysibm.sysdummy1
```

Messages

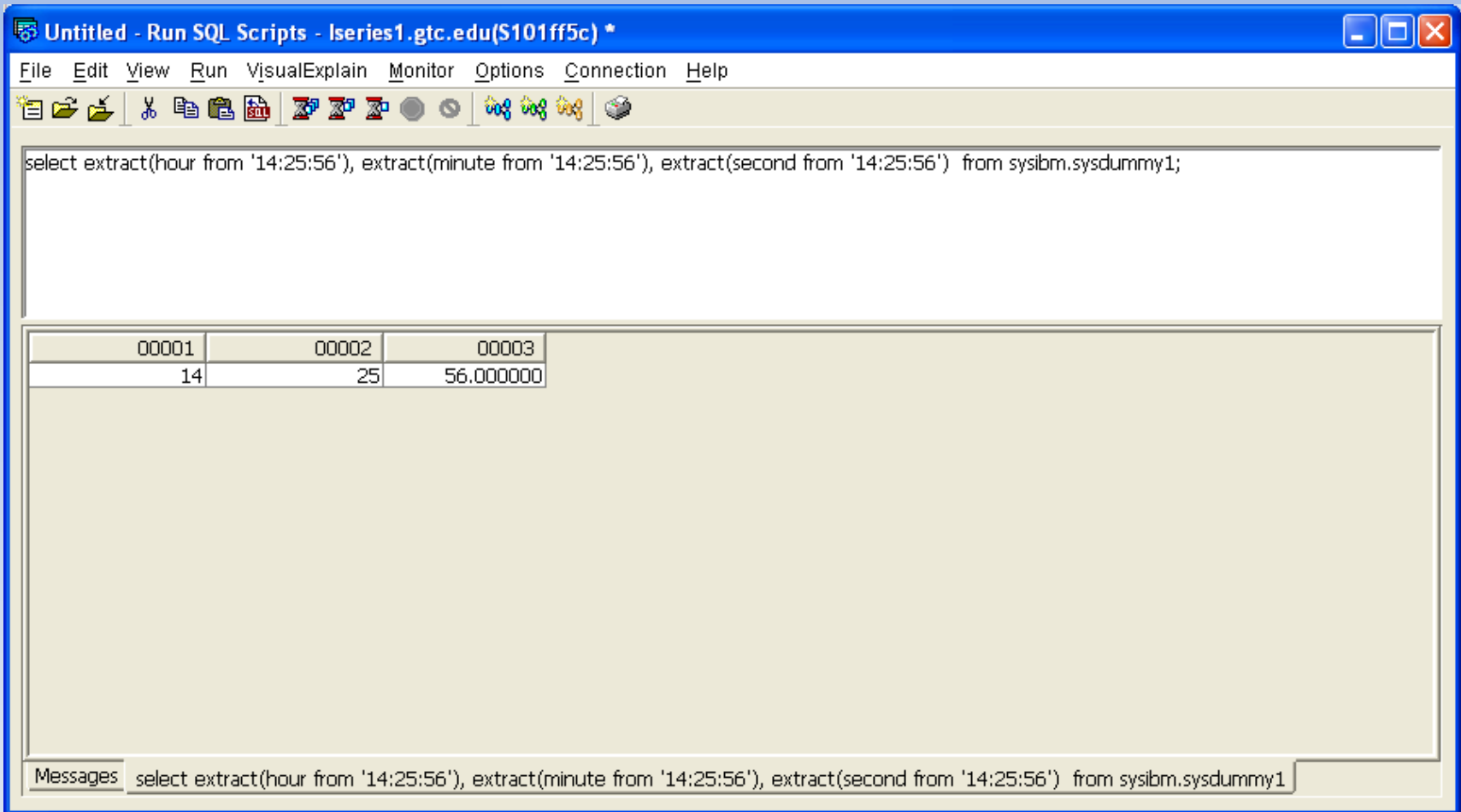
Useful Functions

Time Functions

- **EXTRACT** – Gets portion of date
 - Syntax: `EXTRACT(HOUR FROM datevalue)`
 - Syntax: `EXTRACT(MINUTE FROM datevalue)`
 - Syntax: `EXTRACT(SECOND FROM datevalue)`

Useful Functions

Time functions



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window has a menu bar with "File", "Edit", "View", "Run", "VisualExplain", "Monitor", "Options", "Connection", and "Help". Below the menu bar is a toolbar with various icons. The main area contains a SQL query: `select extract(hour from '14:25:56'), extract(minute from '14:25:56'), extract(second from '14:25:56') from sysibm.sysdummy1;`. Below the query, a table displays the results of the query. The table has three columns with headers "00001", "00002", and "00003". The first row of data contains the values "14", "25", and "56.000000". At the bottom of the window, a "Messages" pane shows the same SQL query: `select extract(hour from '14:25:56'), extract(minute from '14:25:56'), extract(second from '14:25:56') from sysibm.sysdummy1`.

```
select extract(hour from '14:25:56'), extract(minute from '14:25:56'), extract(second from '14:25:56') from sysibm.sysdummy1;
```

00001	00002	00003
14	25	56.000000

Messages select extract(hour from '14:25:56'), extract(minute from '14:25:56'), extract(second from '14:25:56') from sysibm.sysdummy1

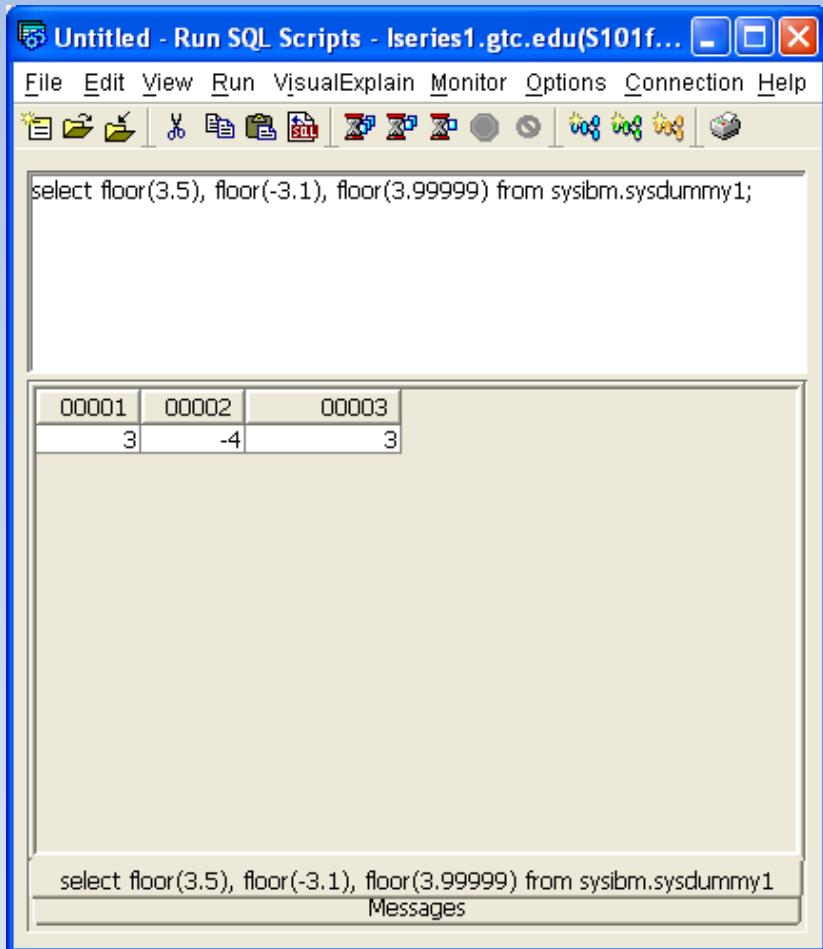
Useful Functions

Rounding

- FLOOR – Round DOWN to nearest integer
 - Syntax: FLOOR(numericvalue)
- CEILING – Round UP to nearest integer
 - Syntax: CEILING(numericvalue)
- ROUND – Round normally
 - Syntax: ROUND(numericvalue,decpos)

Useful Functions

Rounding



Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101f...)

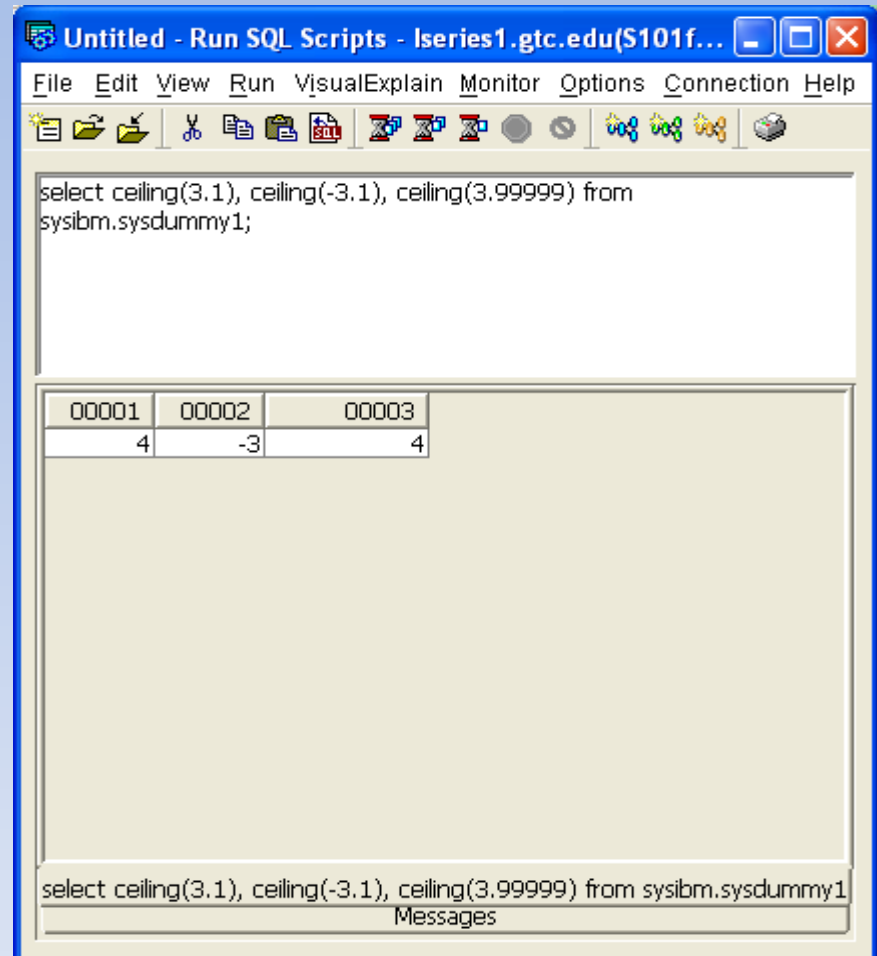
File Edit View Run VisualExplain Monitor Options Connection Help

```
select floor(3.5), floor(-3.1), floor(3.99999) from sysibm.sysdummy1;
```

00001	00002	00003
3	-4	3

select floor(3.5), floor(-3.1), floor(3.99999) from sysibm.sysdummy1
Messages

Detailed description: This screenshot shows a window titled 'Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101f...)'. The menu bar includes 'File', 'Edit', 'View', 'Run', 'VisualExplain', 'Monitor', 'Options', 'Connection', and 'Help'. Below the menu is a toolbar with various icons. The main text area contains the SQL query: 'select floor(3.5), floor(-3.1), floor(3.99999) from sysibm.sysdummy1;'. Below the text area is a table with three columns and one row of data. The columns are labeled '00001', '00002', and '00003'. The values in the row are 3, -4, and 3. At the bottom of the window, there is a status bar with the text 'select floor(3.5), floor(-3.1), floor(3.99999) from sysibm.sysdummy1' and 'Messages'.



Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101f...)

File Edit View Run VisualExplain Monitor Options Connection Help

```
select ceiling(3.1), ceiling(-3.1), ceiling(3.99999) from sysibm.sysdummy1;
```

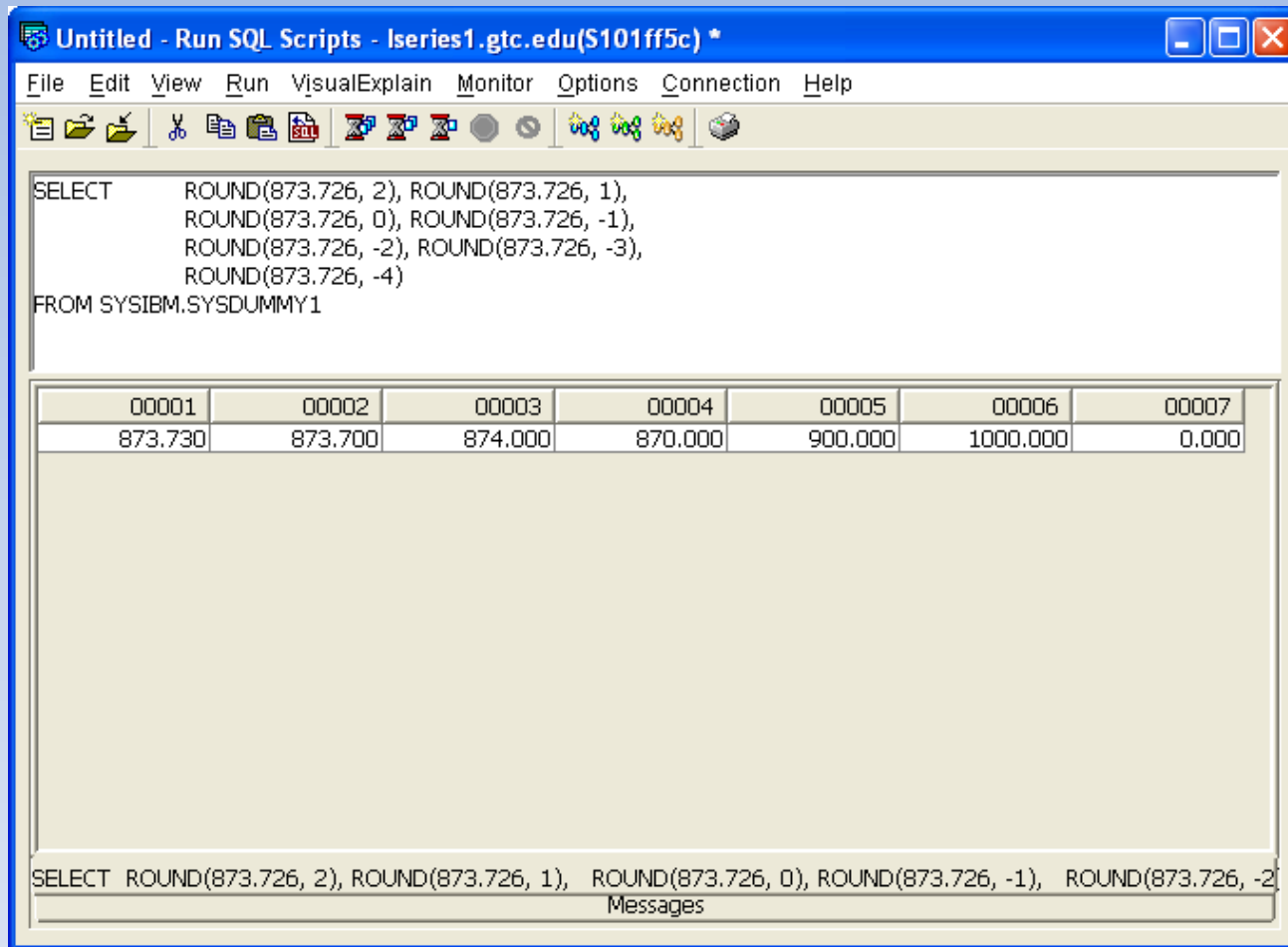
00001	00002	00003
4	-3	4

select ceiling(3.1), ceiling(-3.1), ceiling(3.99999) from sysibm.sysdummy1
Messages

Detailed description: This screenshot shows a window titled 'Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101f...)'. The menu bar includes 'File', 'Edit', 'View', 'Run', 'VisualExplain', 'Monitor', 'Options', 'Connection', and 'Help'. Below the menu is a toolbar with various icons. The main text area contains the SQL query: 'select ceiling(3.1), ceiling(-3.1), ceiling(3.99999) from sysibm.sysdummy1;'. Below the text area is a table with three columns and one row of data. The columns are labeled '00001', '00002', and '00003'. The values in the row are 4, -3, and 4. At the bottom of the window, there is a status bar with the text 'select ceiling(3.1), ceiling(-3.1), ceiling(3.99999) from sysibm.sysdummy1' and 'Messages'.

Useful Functions

Rounding



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(\$101ff5c) *". The menu bar includes File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, and Help. The toolbar contains various icons for file operations and execution. The main area displays the following SQL query:

```
SELECT ROUND(873.726, 2), ROUND(873.726, 1),  
       ROUND(873.726, 0), ROUND(873.726, -1),  
       ROUND(873.726, -2), ROUND(873.726, -3),  
       ROUND(873.726, -4)  
FROM SYSIBM.SYSDUMMY1
```

The results are displayed in a table with 7 columns and 1 row:

00001	00002	00003	00004	00005	00006	00007
873.730	873.700	874.000	870.000	900.000	1000.000	0.000

At the bottom of the window, the following SQL query is visible:

```
SELECT ROUND(873.726, 2), ROUND(873.726, 1), ROUND(873.726, 0), ROUND(873.726, -1), ROUND(873.726, -2)
```

Below the query, the text "Messages" is displayed.

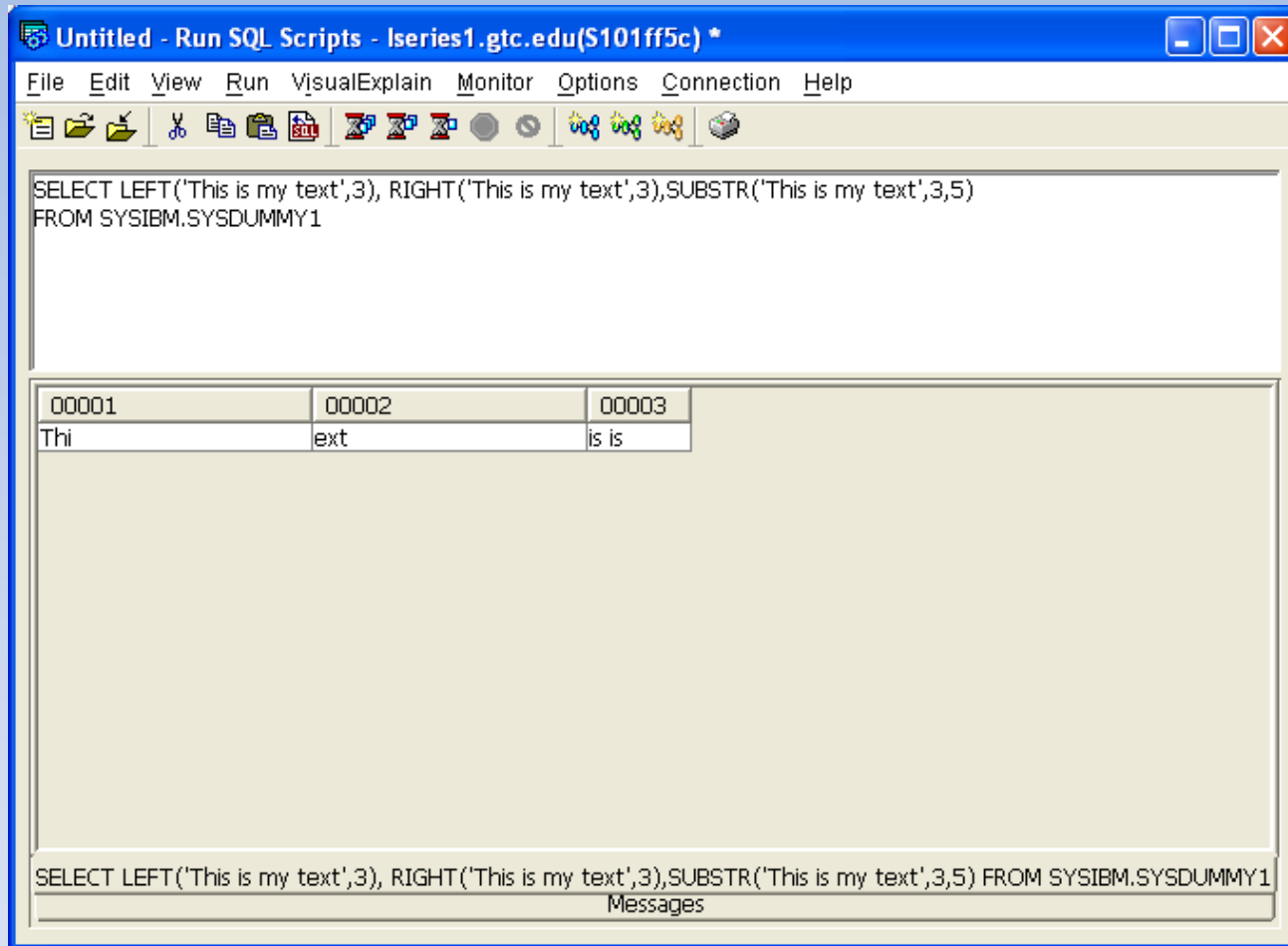
Useful Functions

Text Manipulation

- LEFT – Retrieves leftmost x characters
 - Syntax: LEFT(textvalue, numchars)
- RIGHT – Retrieves rightmost x characters
 - Syntax: RIGHT(textvalue, numchars)
- SUBSTRING – Takes a string out of the middle
 - Syntax: SUBSTRING(textvalue, start, *length*)
 - Length is optional

Useful Functions

Text Manipulation



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The menu bar includes File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, and Help. The toolbar contains various icons for file operations and execution. The main text area contains the following SQL query:

```
SELECT LEFT('This is my text',3), RIGHT('This is my text',3),SUBSTR('This is my text',3,5)
FROM SYSIBM.SYSDUMMY1
```

The results are displayed in a table with three columns:

00001	00002	00003
Thi	ext	is is

At the bottom of the window, the same query is repeated, followed by the text "Messages".

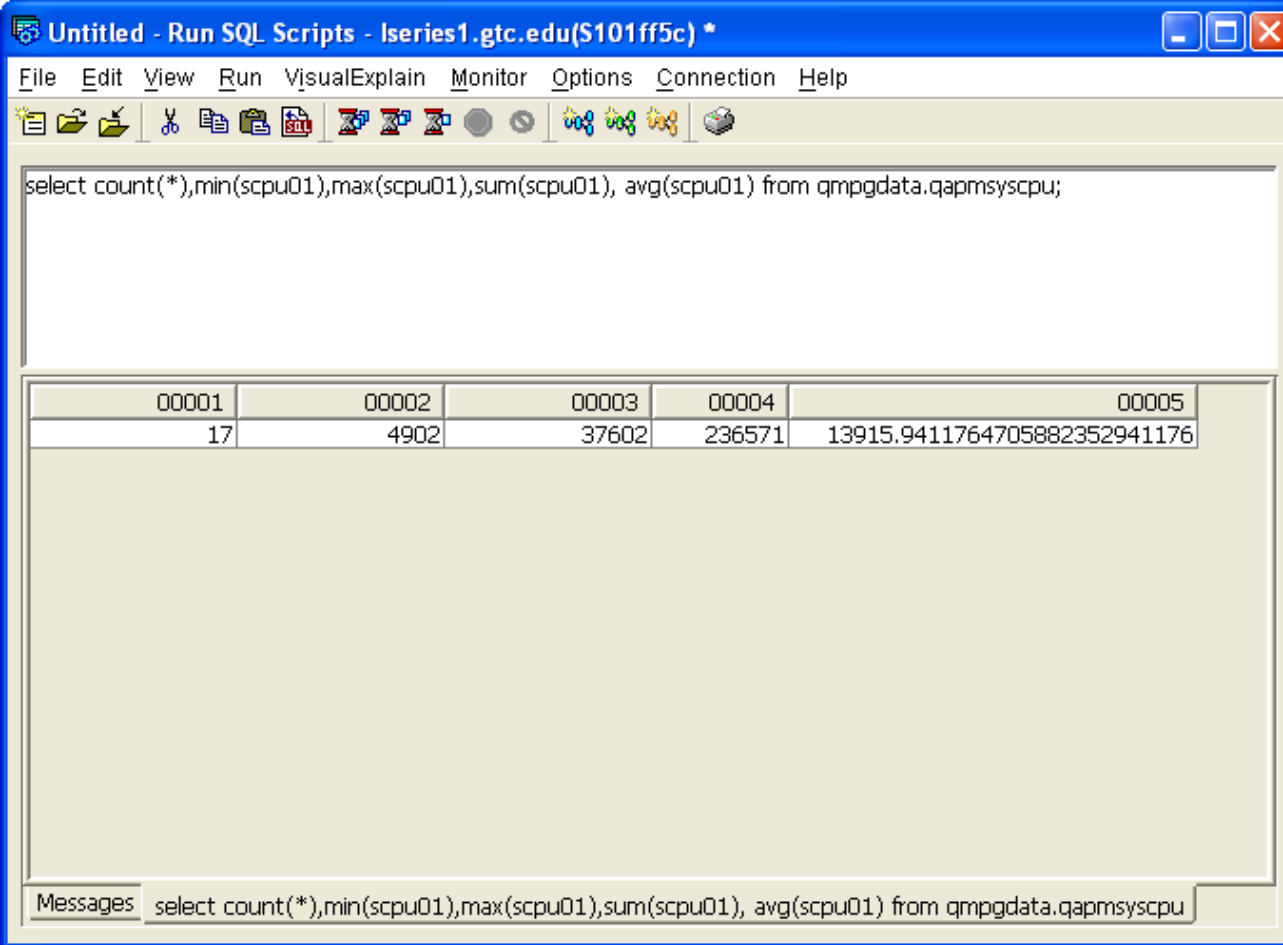
Useful Functions

Aggregate functions

- COUNT – Count number of records returned
 - Syntax: Count(*)
- SUM – Add up the column
 - Syntax: SUM(columnname)
- MIN – Get minimum value in column
 - Syntax: MIN(columnname)
- MAX – Get maximum value in column
 - Syntax: MAX(columnname)
- AVG – Get the average of the column
 - Syntax: AVG(columnname)

Useful Functions

Aggregate Functions



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window contains a menu bar with "File", "Edit", "View", "Run", "Visual Explain", "Monitor", "Options", "Connection", and "Help". Below the menu bar is a toolbar with various icons. The main area of the window displays the following SQL query:

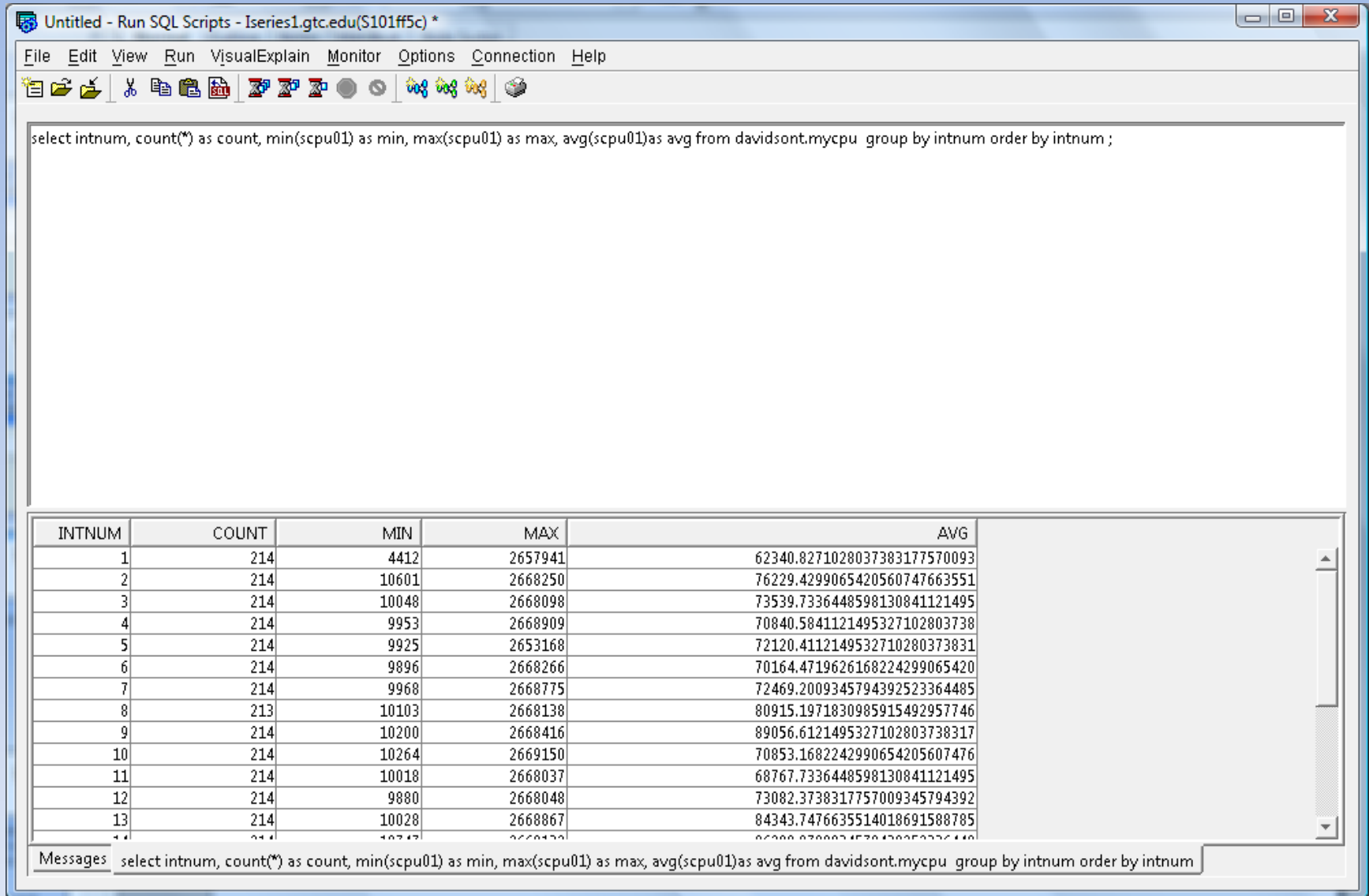
```
select count(*),min(scpu01),max(scpu01),sum(scpu01), avg(scpu01) from qmpgdata.qapmsyscpu;
```

The results of the query are displayed in a table with five columns:

00001	00002	00003	00004	00005
17	4902	37602	236571	13915.9411764705882352941176

At the bottom of the window, there is a "Messages" pane containing the same SQL query: `select count(*),min(scpu01),max(scpu01),sum(scpu01), avg(scpu01) from qmpgdata.qapmsyscpu;`

Useful Functions



The screenshot shows a window titled "Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c) *". The window contains a menu bar with "File", "Edit", "View", "Run", "Visual Explain", "Monitor", "Options", "Connection", and "Help". Below the menu bar is a toolbar with various icons. The main area contains a SQL query: `select intnum, count(*) as count, min(scpu01) as min, max(scpu01) as max, avg(scpu01) as avg from dauidson.mycpu group by intnum order by intnum ;`. Below the query is a table with the following data:

INTNUM	COUNT	MIN	MAX	AVG
1	214	4412	2657941	62340.8271028037383177570093
2	214	10601	2668250	76229.4299065420560747663551
3	214	10048	2668098	73539.7336448598130841121495
4	214	9953	2668909	70840.5841121495327102803738
5	214	9925	2653168	72120.4112149532710280373831
6	214	9896	2668266	70164.4719626168224299065420
7	214	9968	2668775	72469.2009345794392523364485
8	213	10103	2668138	80915.1971830985915492957746
9	214	10200	2668416	89056.6121495327102803738317
10	214	10264	2669150	70853.1682242990654205607476
11	214	10018	2668037	68767.7336448598130841121495
12	214	9880	2668048	73082.3738317757009345794392
13	214	10028	2668867	84343.7476635514018691588785
14	214	10077	2668478	80000.00000000000000000000

At the bottom of the window, there is a "Messages" pane containing the same SQL query: `select intnum, count(*) as count, min(scpu01) as min, max(scpu01) as max, avg(scpu01) as avg from dauidson.mycpu group by intnum order by intnum ;`

Tips and Techniques

Tips and Techniques

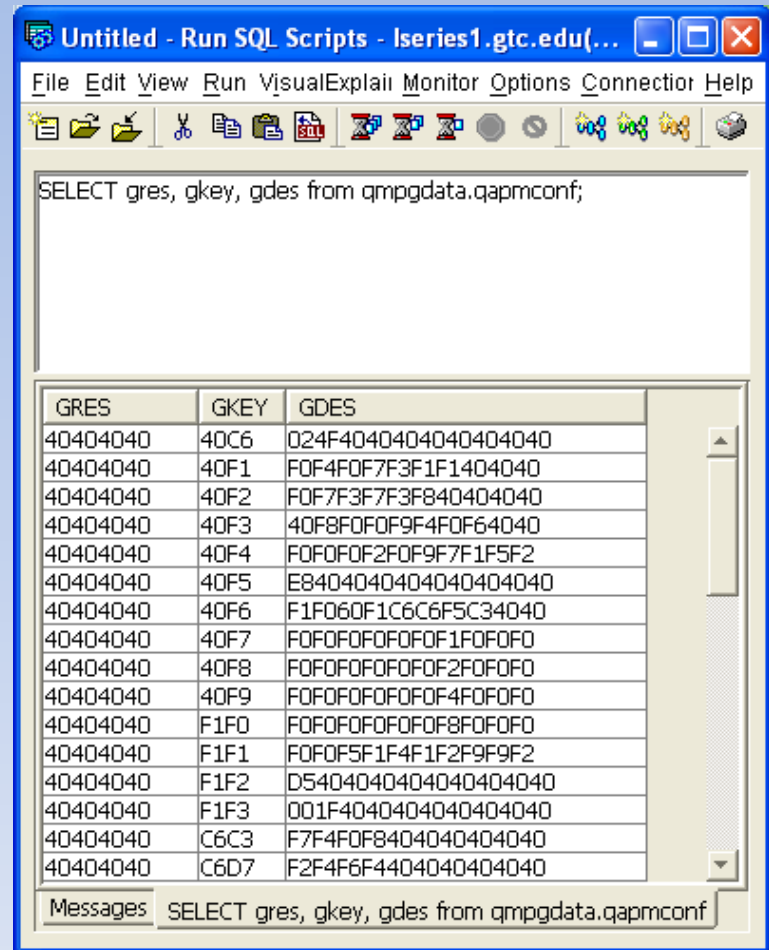
Before doing a delete do a select

- Do a select first to determine what records you will be deleting.
- Once you have the right selection, copy & paste the where clause to your delete.

Tips and Techniques

Dealing with 65535 Data

- By default SQL returns the hex values of 65535 data
- Solution: Change your connection settings



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(...)" with a menu bar (File, Edit, View, Run, VisualExplain, Monitor, Options, Connector, Help) and a toolbar. The main area contains a SQL query: `SELECT gres, gkey, gdes from qmpgdata.qapmconf;`. Below the query is a table with the following data:

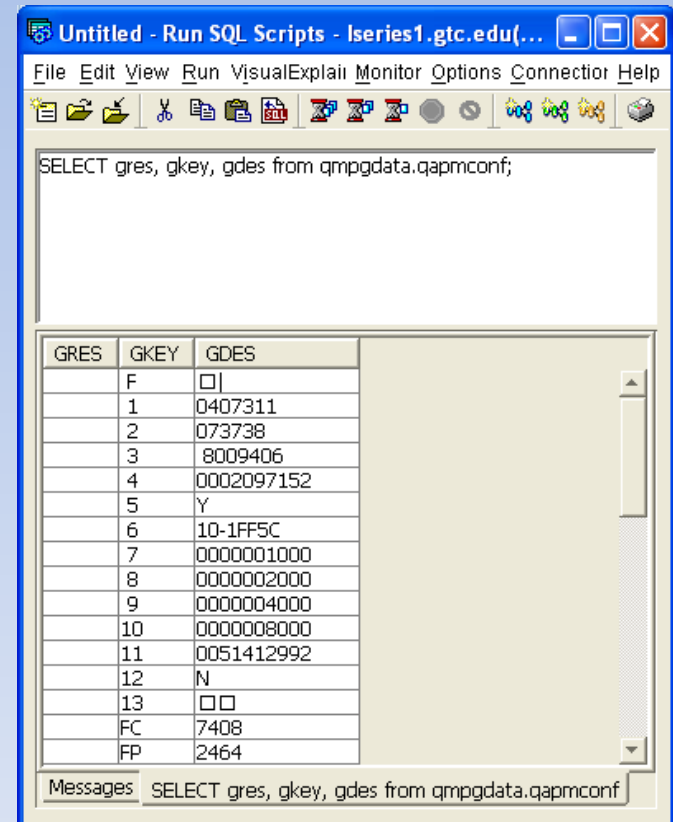
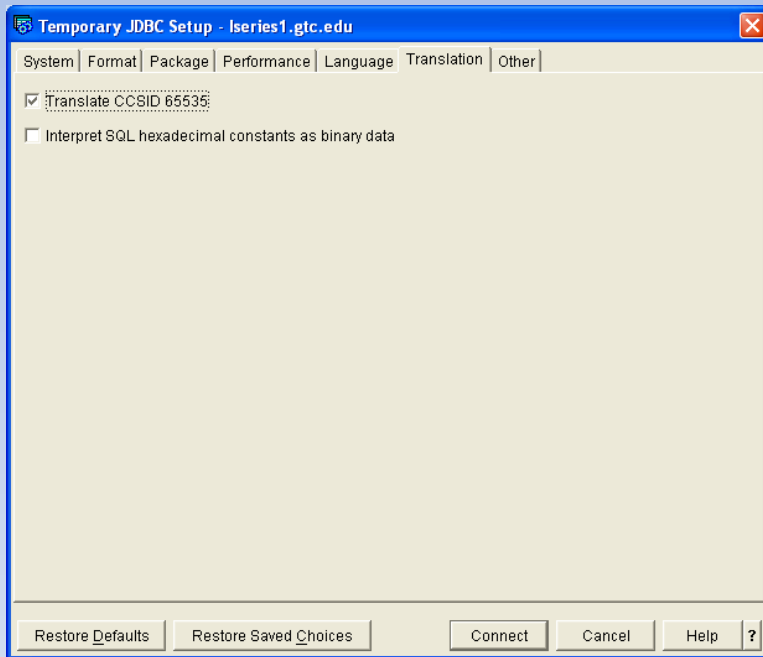
GRES	GKEY	GDES
40404040	40C6	024F4040404040404040
40404040	40F1	F0F4F0F7F3F1F1404040
40404040	40F2	F0F7F3F7F3F840404040
40404040	40F3	40F8F0F0F9F4F0F64040
40404040	40F4	F0F0F0F2F0F9F7F1F5F2
40404040	40F5	E8404040404040404040
40404040	40F6	F1F060F1C6C6F5C34040
40404040	40F7	F0F0F0F0F0F0F1F0F0F0
40404040	40F8	F0F0F0F0F0F0F2F0F0F0
40404040	40F9	F0F0F0F0F0F0F4F0F0F0
40404040	F1F0	F0F0F0F0F0F0F8F0F0F0
40404040	F1F1	F0F0F5F1F4F1F2F9F9F2
40404040	F1F2	D5404040404040404040
40404040	F1F3	001F4040404040404040
40404040	C6C3	F7F4F0F8404040404040
40404040	C6D7	F2F4F6F4404040404040

At the bottom, a "Messages" pane shows the same query: `SELECT gres, gkey, gdes from qmpgdata.qapmconf;`

Tips and Techniques

Dealing with 65535 Data

- Take Connection/Use Temporary JDBC Settings



Tips and Techniques

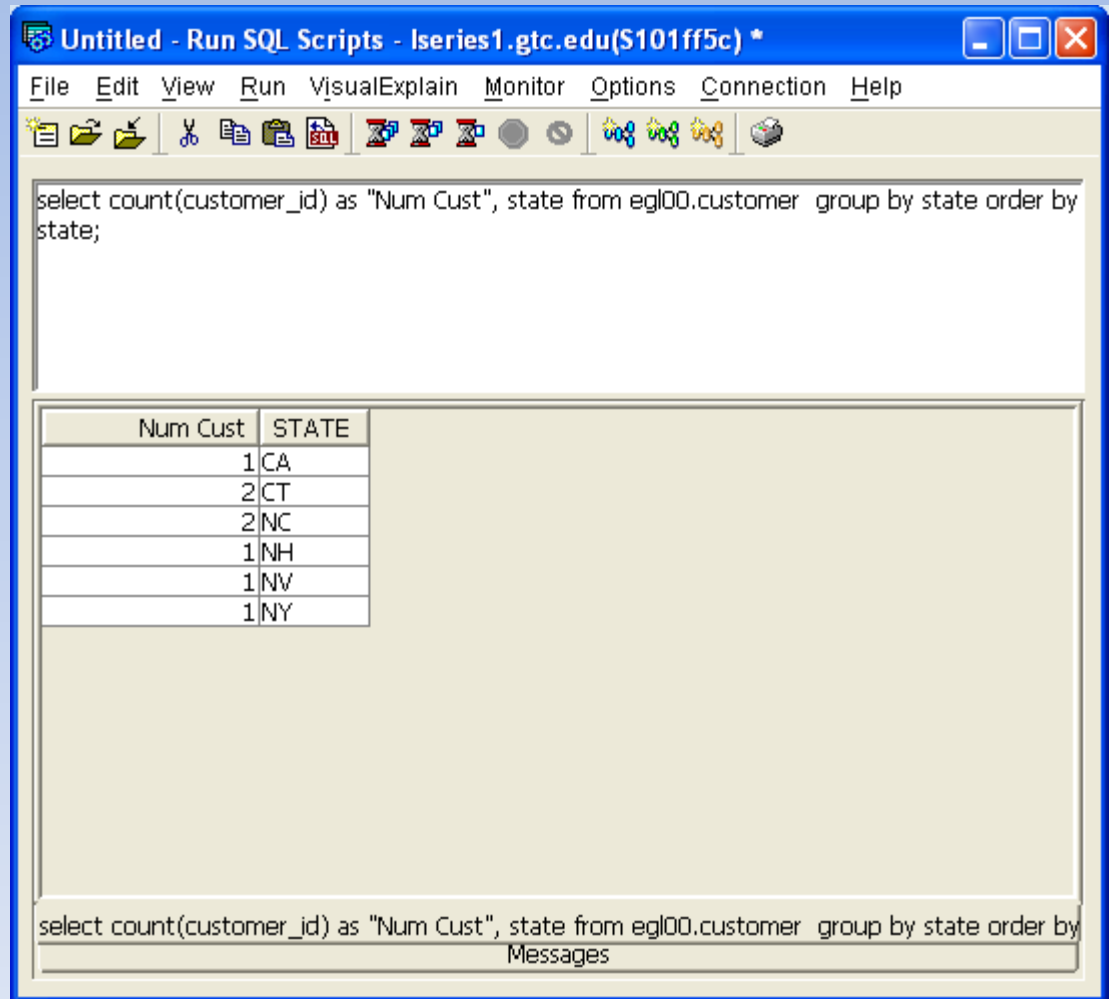
Changing code fields into text for readability

- Situation: You have a field that uses a 'code' to represent values, such as states. You want to show the actual state name in your SQL output

Tips and Techniques

Changing code fields into text for readability

- The file:



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window has a menu bar with "File", "Edit", "View", "Run", "Visual Explain", "Monitor", "Options", "Connection", and "Help". Below the menu bar is a toolbar with various icons. The main area contains a SQL query: `select count(customer_id) as "Num Cust", state from egl00.customer group by state order by state;`. Below the query is a table with two columns: "Num Cust" and "STATE". The table contains the following data:

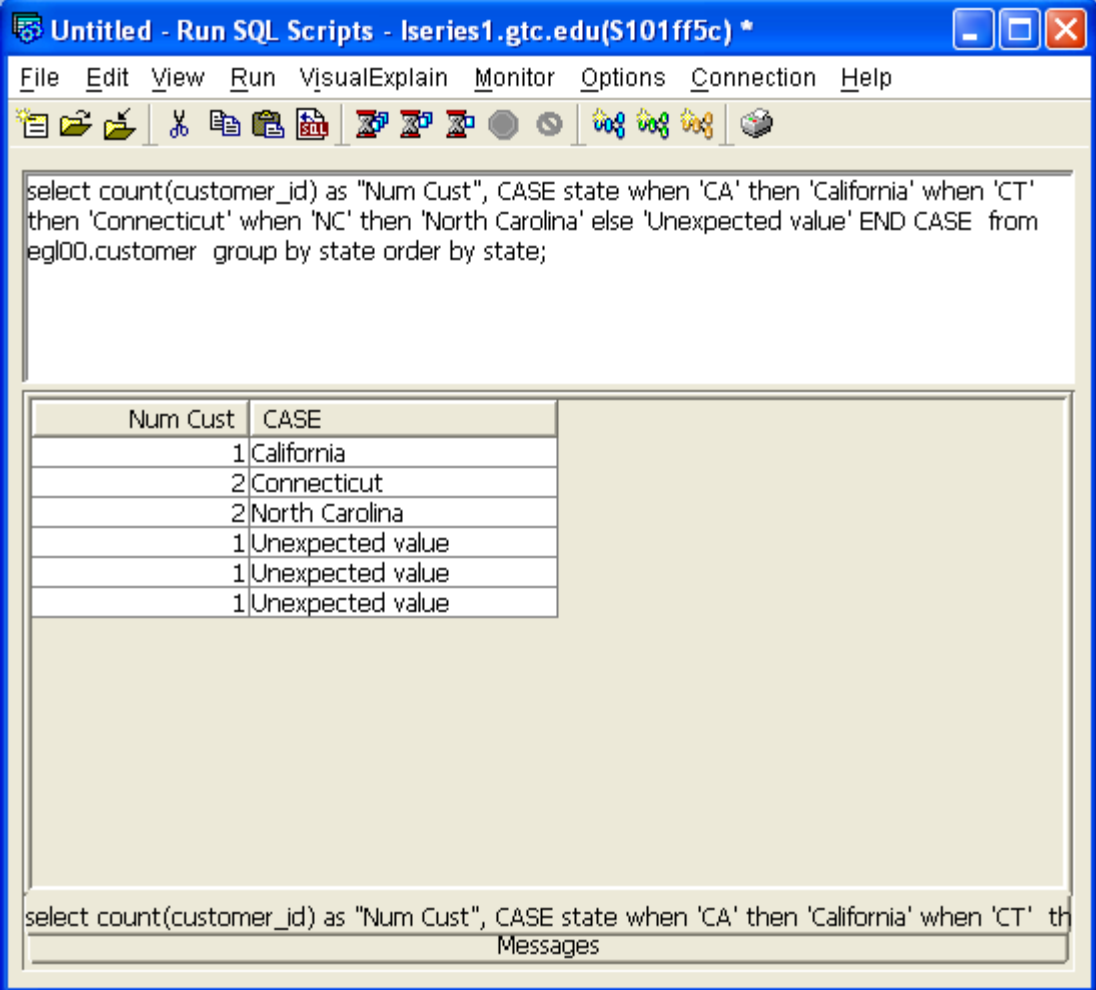
Num Cust	STATE
1	CA
2	CT
2	NC
1	NH
1	NV
1	NY

At the bottom of the window, there is a status bar with the text "select count(customer_id) as "Num Cust", state from egl00.customer group by state order by Messages".

Tips and Techniques

Changing code fields into text for readability

- Solution:
The SQL CASE statement



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window contains a menu bar (File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, Help) and a toolbar with various icons. The main area displays the following SQL query:

```
select count(customer_id) as "Num Cust", CASE state when 'CA' then 'California' when 'CT' then 'Connecticut' when 'NC' then 'North Carolina' else 'Unexpected value' END CASE from egl00.customer group by state order by state;
```

Below the query, a table displays the results of the execution:

Num Cust	CASE
1	California
2	Connecticut
2	North Carolina
1	Unexpected value
1	Unexpected value
1	Unexpected value

At the bottom of the window, the same SQL query is repeated, followed by the text "Messages".

Tips and Techniques

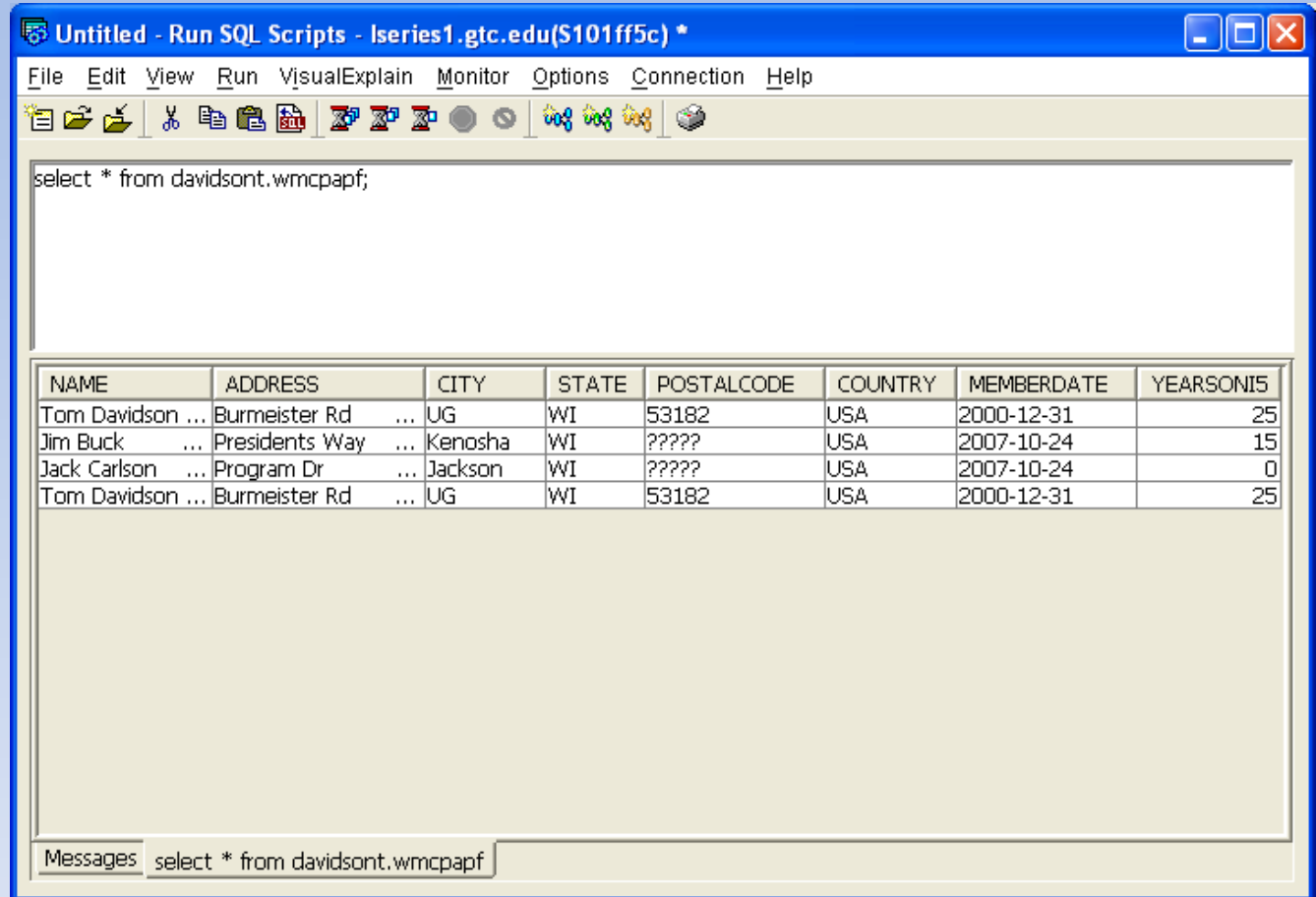
Deleting duplicate records

- Situation: You have accidentally placed duplicate records into your database. You can't use DELETE because the records are truly duplicates and you still need one to remain

Tips and Techniques

Deleting duplicate records

- The file



The screenshot shows a window titled "Untitled - Run SQL Scripts - lseries1.gtc.edu(S101ff5c) *". The window contains a menu bar (File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, Help) and a toolbar with various icons. The main area displays the SQL query: `select * from davidsont.wmcpapf;`. Below the query, a table of results is shown with the following data:

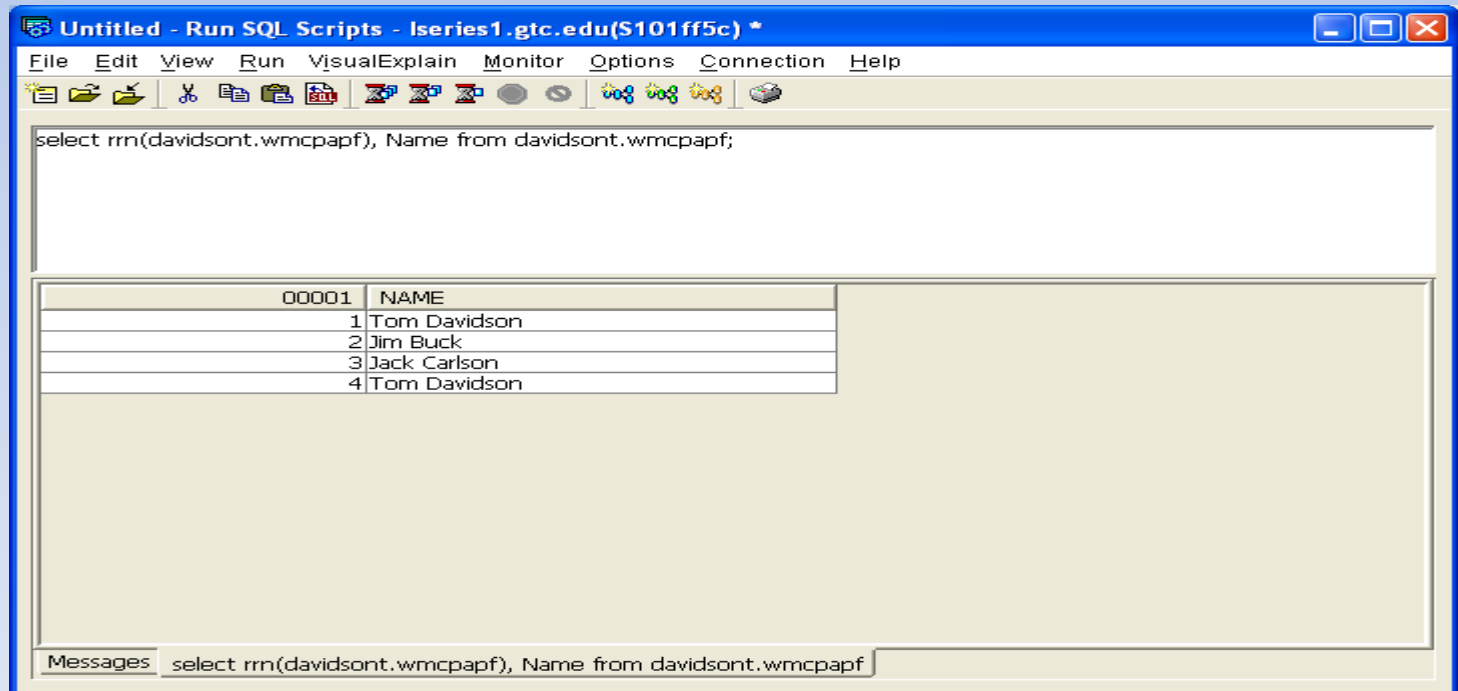
NAME	ADDRESS	CITY	STATE	POSTALCODE	COUNTRY	MEMBERDATE	YEARSONI5
Tom Davidson ...	Burmeister Rd ...	UG	WI	53182	USA	2000-12-31	25
Jim Buck ...	Presidents Way ...	Kenosha	WI	?????	USA	2007-10-24	15
Jack Carlson ...	Program Dr ...	Jackson	WI	?????	USA	2007-10-24	0
Tom Davidson ...	Burmeister Rd ...	UG	WI	53182	USA	2000-12-31	25

At the bottom of the window, a "Messages" pane shows the executed query: `select * from davidsont.wmcpapf`.

Tips and Techniques

Deleting duplicate records

- Solution: You will need to use the RRN function to tell records apart, then delete by rrn



The screenshot shows a window titled "Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c)". The window contains a menu bar (File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, Help) and a toolbar with various icons. The main area displays the following SQL query:

```
select rrn(davidsont.wmcpapf), Name from davidsont.wmcpapf;
```

Below the query, a table of results is displayed:

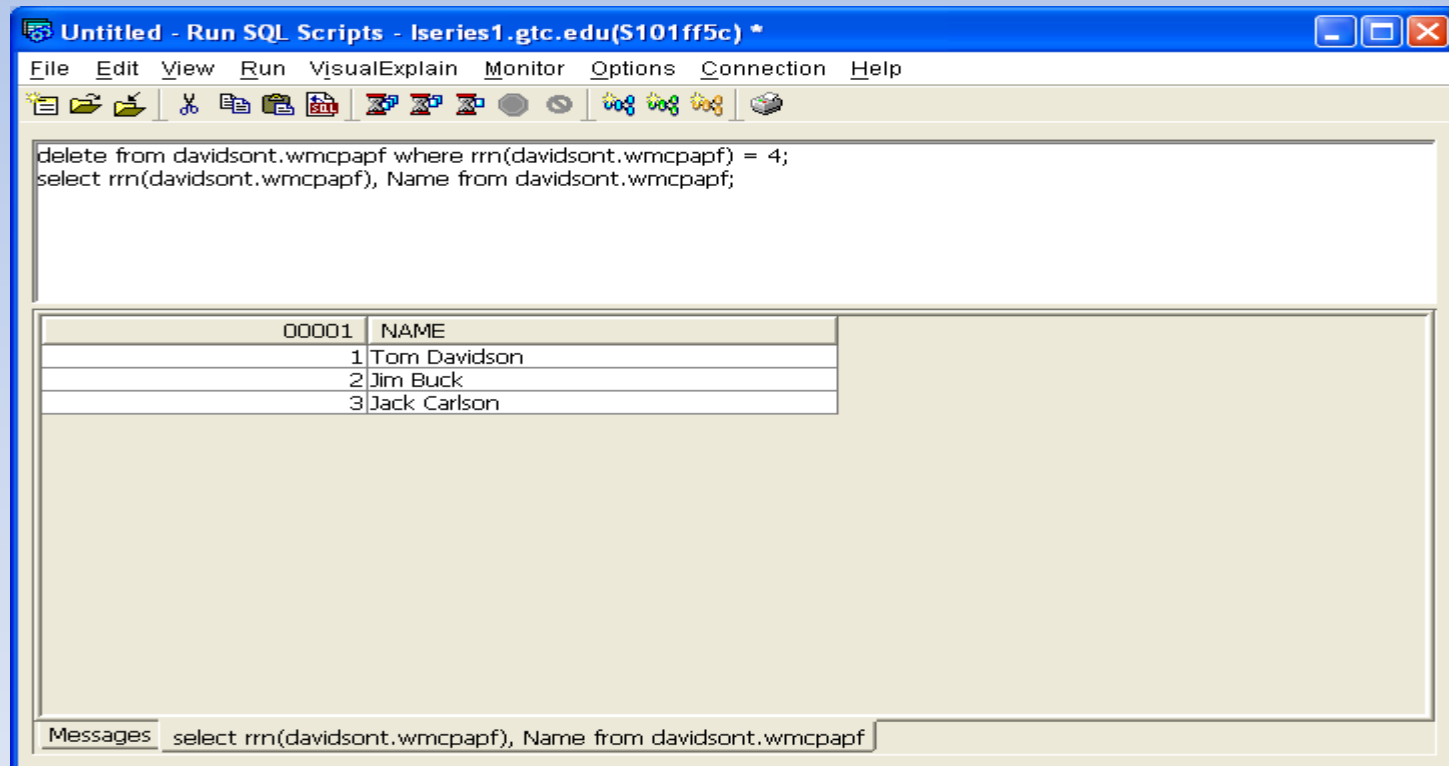
00001	NAME
1	Tom Davidson
2	Jim Buck
3	Jack Carlson
4	Tom Davidson

At the bottom of the window, a "Messages" pane shows the executed query: `select rrn(davidsont.wmcpapf), Name from davidsont.wmcpapf`.

Tips and Techniques

Deleting duplicate records

- Solution: Delete using `RRN(davidsont.wmcpapf)` then redisplay



The screenshot shows a window titled "Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c) *". The menu bar includes File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, and Help. The toolbar contains various icons for file operations and execution. The main text area contains the following SQL script:

```
delete from davidsont.wmcpapf where rrn(davidsont.wmcpapf) = 4;  
select rrn(davidsont.wmcpapf), Name from davidsont.wmcpapf;
```

Below the script, a table displays the results of the select query:

00001	NAME
1	Tom Davidson
2	Jim Buck
3	Jack Carlson

At the bottom, a "Messages" pane shows the output of the select query: `select rrn(davidsont.wmcpapf), Name from davidsont.wmcpapf`.

Tips and Techniques

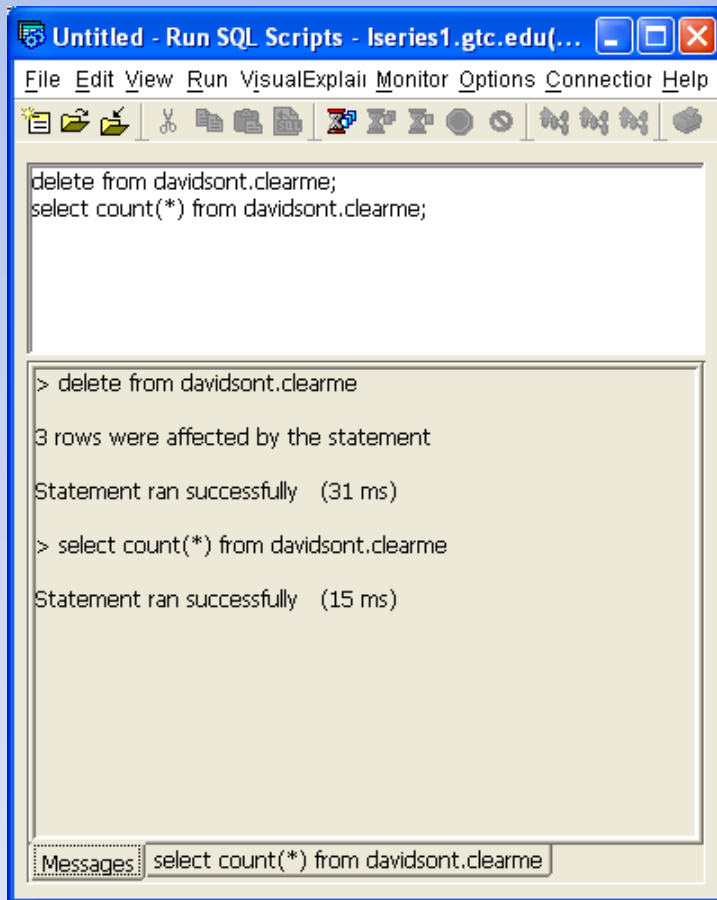
Clearing a file that is open

- Situation: You need to clear a file, but it always seems to be open by some task.
- Solution: The SQL delete statement. This does a record-by-record delete, same as if you wrote an RPG program to delete all the records.
- Gotcha: There can be no record locks

Tips and Techniques

Clearing a file that is open

- I have the file open on greenscreen



Untitled - Run SQL Scripts - lseries1.gtc.edu(...)

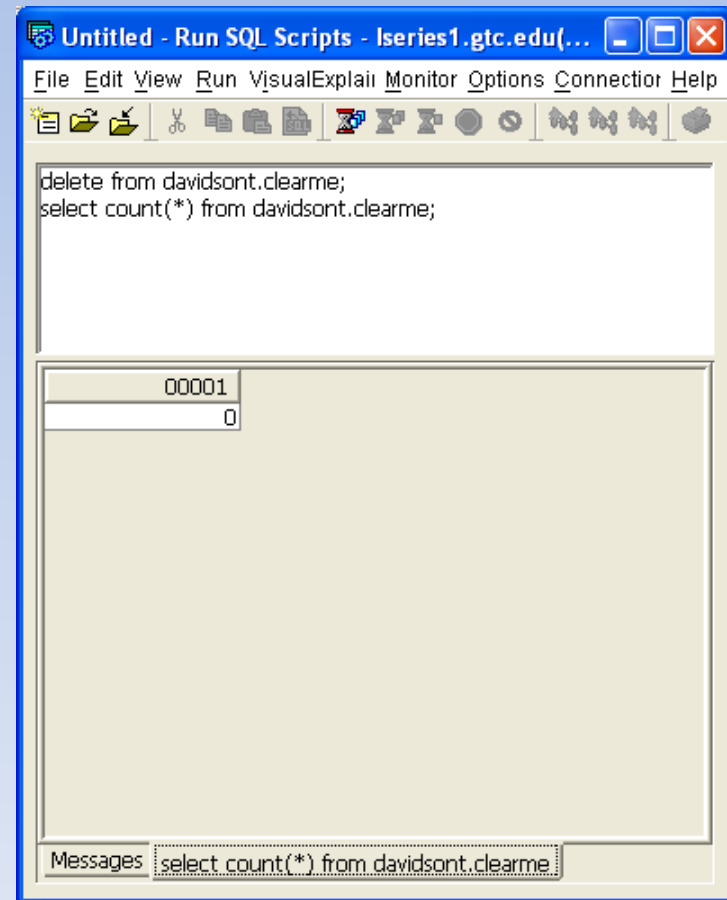
File Edit View Run Visual Explain Monitor Options Connector Help

```
delete from dauidsonr.clearme;  
select count(*) from dauidsonr.clearme;
```

> delete from dauidsonr.clearme
3 rows were affected by the statement
Statement ran successfully (31 ms)

> select count(*) from dauidsonr.clearme
Statement ran successfully (15 ms)

Messages: select count(*) from dauidsonr.clearme



Untitled - Run SQL Scripts - lseries1.gtc.edu(...)

File Edit View Run Visual Explain Monitor Options Connector Help

```
delete from dauidsonr.clearme;  
select count(*) from dauidsonr.clearme;
```

00001
0

Messages: select count(*) from dauidsonr.clearme

Gotcha's

Gotcha's

- SQL gives you NO guarantee of how the data will be ordered if it isn't in the 'ORDER BY' clause

Gotcha's

Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c) *

File Edit View Run Visual Explain Monitor Options Connection Help

select * from dauidson.gtcstp order by cstat;

CCITY	CSTAT	CZIP	CURBIL	AMTOWE	PAYDAT	PAYDA
WASHINGTON	DC	12345	23.56	0.00	20061210	2006-12
COMSTOCK	FL	90210	36.98	25.45	20061212	2006-12
SMALLVILLE	IA	75846	9.99	0.00	20061102	2006-11
SPRINGFIELD	IL	60080	26.45	0.00	20060701	2006-07
KALAMAZOO	MI	49001	38.68	90.72	20060615	2006-06
KALAMAZOO	MI	49007	43.08	155.18	20060510	2006-05
KALAMAZOO	MI	49008	140.17	185.93	20060621	2006-06
KALAMAZOO	MI	49008	20.17	106.12	20060622	2006-06
KALAMAZOO	MI	49001	20.17	102.08	20060614	2006-06
KALAMAZOO	MI	49001	31.90	95.70	20060623	2006-06
KALAMAZOO	MI	49001	22.28	66.84	20060619	2006-06
KALAMAZOO	MI	49002	20.17	160.27	20060617	2006-06
KALAMAZOO	MI	49002	198.33	344.87	20060614	2006-06
KALAMAZOO	MI	49007	20.17	152.06	20060614	2006-06
KALAMAZOO	MI	49008	20.17	206.11	20060615	2006-06
KALAMAZOO	MI	49007	48.90	146.70	20060607	2006-06
KALAMAZOO	MI	49002	20.17	95.35	20060528	2006-05
KALAMAZOO	MI	49007	20.17	92.83	20060615	2006-06
KALAMAZOO	MI	49001	24.08	181.69	20060616	2006-06
KALAMAZOO	MI	49007	20.17	345.63	20060615	2006-06
KALAMAZOO	MI	49007	20.17	172.49	20060617	2006-06
KALAMAZOO	MI	49007	20.17	226.15	20060614	2006-06
KALAMAZOO	MI	49008	20.17	60.51	20060512	2006-05
KALAMAZOO	MI	49007	20.17	257.01	20060615	2006-06
KALAMAZOO	MI	49001	100.20	623.15	20060615	2006-06

Messages: select * from dauidson.gtcstp order by cstat

Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c) *

File Edit View Run Visual Explain Monitor Options Connection Help

select * from dauidson.gtcstp order by cstat;

CCITY	CSTAT	CZIP	CURBIL	AMTOWE	PAYDAT	PAYDA
WASHINGTON	DC	12345	23.56	0.00	20061210	2006-12
COMSTOCK	FL	90210	36.98	25.45	20061212	2006-12
SMALLVILLE	IA	75846	9.99	0.00	20061102	2006-11
SPRINGFIELD	IL	60080	26.45	0.00	20060701	2006-07
SWAMP HOLLAR	MI	49061	20.17	36.03	20060617	2006-06
PULLMAN	MI	49081	20.17	128.59	20060616	2006-06
LAWRENCE	MI	49067	128.88	60.51	20060615	2006-06
LAWRENCE	MI	49067	20.17	49.28	20060616	2006-06
KALAMAZOO	MI	49001	38.68	90.72	20060615	2006-06
KALAMAZOO	MI	49007	43.08	155.18	20060510	2006-05
KALAMAZOO	MI	49008	140.17	185.93	20060621	2006-06
PORTAGE	MI	49003	30.62	91.86	20060609	2006-06
PULLMAN	MI	49081	28.05	140.58	20060616	2006-06
KALAMAZOO	MI	49008	20.17	106.12	20060622	2006-06
KALAMAZOO	MI	49001	20.17	102.08	20060614	2006-06
PORTAGE	MI	49003	46.77	117.69	20060616	2006-06
PORTAGE	MI	49001	20.17	224.81	20060617	2006-06
KALAMAZOO	MI	49001	31.90	95.70	20060623	2006-06
KALAMAZOO	MI	49001	22.28	66.84	20060619	2006-06
KALAMAZOO	MI	49002	20.17	160.27	20060617	2006-06
KALAMAZOO	MI	49002	198.33	344.87	20060614	2006-06
KALAMAZOO	MI	49007	20.17	152.06	20060614	2006-06
KALAMAZOO	MI	49008	20.17	206.11	20060615	2006-06
KALAMAZOO	MI	49007	48.90	146.70	20060607	2006-06
KALAMAZOO	MI	49002	20.17	95.35	20060528	2006-05

Messages: select * from dauidson.gtcstp order by cstat

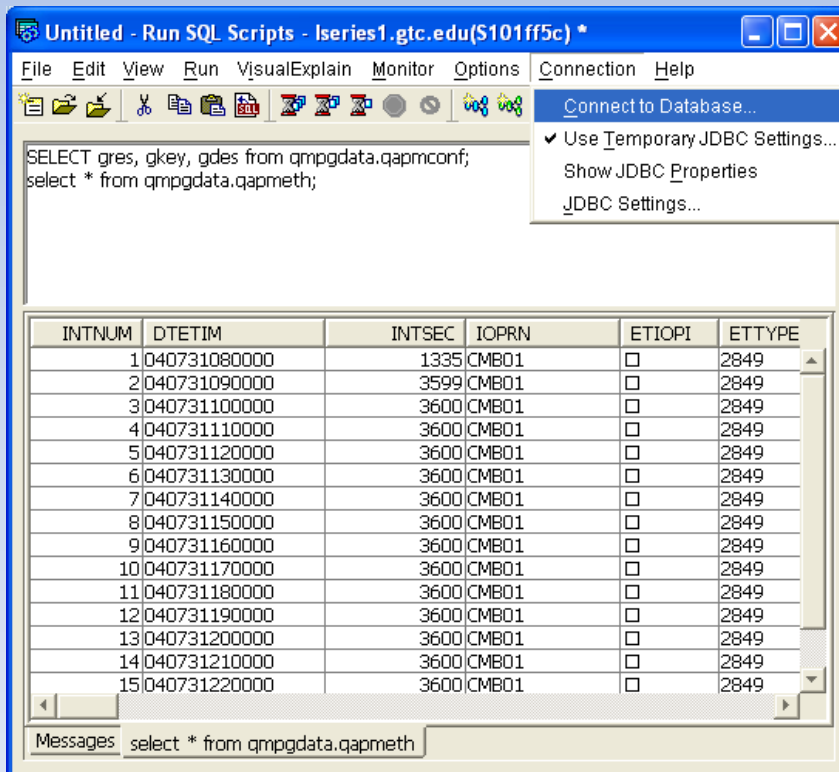
Gotcha's

- SQL leaves all the files you have used open

```
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
*****
01 *                               Display Open Files                               * 01
02 *                                                                           * 02
03 * Job . . . :   QZDASSINIT       User . . . :   QUSER           Number . . . :   796807 * 03
04 * Number of open data paths . . . . . :           4             * 04
05 *                                                                           * 05
06 *                                                                           * 06
07 *                               Member/   Record   File       I/O  ----Open---  Relative * 07
08 * File           Library   Device     Format     Type   Count Opt Shr-Nbr   Record * 08
09 * SYSROUTINE   QSYS2       SYSRO00001 SYSRO00001 PHY     2 I  NO           125 * 09
10 * QASQRESL     QSYS2       QASQRESL   SYSRO00001 LGL     2 I  NO           125 * 10
11 * QAPMCONF     QMPGDATA   Q213073737 FORMAT0001 PHY     1 I  NO            45 * 11
12 * QAPMETH      QMPGDATA   Q213073737 FORMAT0001 PHY     1 I  NO            17 * 12
13 *                                                                           * 13
21 * Press Enter to continue.                                           * 21
22 *                                                                           * 22
23 * F3=Exit      F5=Refresh   F11=Display scoping data   F12=Cancel   F16=Job menu * 23
24 *                                                                           * 24
*****
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
```

Gotcha's

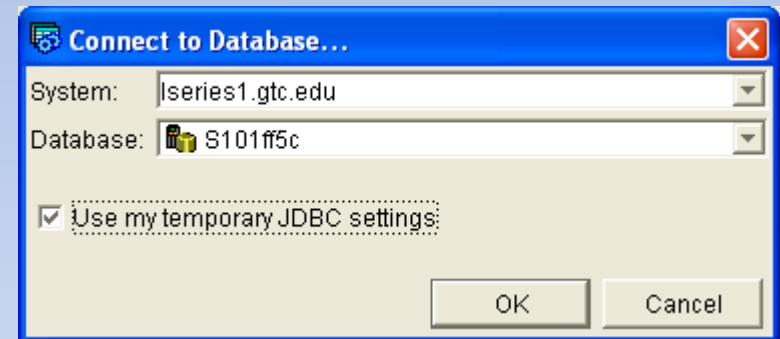
- SQL leaves all the files you have used open
- Solution: Open a new connection



The screenshot shows a window titled "Untitled - Run SQL Scripts - Iseries1.gtc.edu(S101ff5c)". The menu bar includes File, Edit, View, Run, Visual Explain, Monitor, Options, Connection, and Help. A "Connect to Database..." menu item is highlighted, with a sub-menu showing "Use Temporary JDBC Settings..." (checked), "Show JDBC Properties", and "JDBC Settings...". The main text area contains the SQL query: `SELECT gres, gkey, gdes from qmpgdata.qapmconf; select * from qmpgdata.qapmeth;`. Below the query is a table with the following data:

INTNUM	DTETIM	INTSEC	IOPRN	ETIOPI	ETTYPE
1	040731080000	1335	CMB01	<input type="checkbox"/>	2849
2	040731090000	3599	CMB01	<input type="checkbox"/>	2849
3	040731100000	3600	CMB01	<input type="checkbox"/>	2849
4	040731110000	3600	CMB01	<input type="checkbox"/>	2849
5	040731120000	3600	CMB01	<input type="checkbox"/>	2849
6	040731130000	3600	CMB01	<input type="checkbox"/>	2849
7	040731140000	3600	CMB01	<input type="checkbox"/>	2849
8	040731150000	3600	CMB01	<input type="checkbox"/>	2849
9	040731160000	3600	CMB01	<input type="checkbox"/>	2849
10	040731170000	3600	CMB01	<input type="checkbox"/>	2849
11	040731180000	3600	CMB01	<input type="checkbox"/>	2849
12	040731190000	3600	CMB01	<input type="checkbox"/>	2849
13	040731200000	3600	CMB01	<input type="checkbox"/>	2849
14	040731210000	3600	CMB01	<input type="checkbox"/>	2849
15	040731220000	3600	CMB01	<input type="checkbox"/>	2849

At the bottom, a "Messages" pane shows the command: `select * from qmpgdata.qapmeth`.



The screenshot shows a "Connect to Database..." dialog box. The "System:" field is set to "Iseries1.gtc.edu" and the "Database:" field is set to "S101ff5c". The "Use my temporary JDBC settings:" checkbox is checked. There are "OK" and "Cancel" buttons at the bottom right.

This ends your ODBC connection, releasing the files

Summary

- SQL doesn't need to be complex to be useful
- BIF's can make your life easier, just remember the rules
- Watch out for the “Gotcha's”

For More Information

- IBM Information Center -
<http://publib.boulder.ibm.com/infocenter/iseres/v5r4/index.jsp>
 - Go to Programming/Languages/SQL