

## Notas técnicas de AS/400 - Tip Nro. 40

(Lo nuevo, lo escondido, o simplemente lo de siempre pero bien explicado)

---

“Notas técnicas de AS/400” se envía con frecuencia variable y absolutamente sin cargo como un servicio a nuestros clientes AS/400. Contiene notas/recursos/artículos técnicos desarrollados en forma totalmente objetiva e independiente. Teknoda es una organización de servicios de tecnología informática y NO comercializa hardware, software ni otros productos. Conteste este mail con asunto “REMOVER” si no desea recibir más esta publicación. Si desea suscribir otra dirección de e-mail para que comience a recibir los “Tips”, envíe un mensaje desde esa dirección a [letter400@teknoda.com](mailto:letter400@teknoda.com), aclarando nombre, empresa, cargo y país del suscriptor.

Visite nuestro sitio en Internet: <http://www.teknoda.com>

### Cómo ejecutar sentencias SQL usando el comando CL RUNSQLSTM

<b>Tema:</b>	Programación, Base de Datos, SQL
<b>Utilidad:</b>	Ejecutar sentencias SQL desde un archivo de texto
<b>Nivel:</b>	Intermedio.
<b>Versión:</b>	Todas las versiones

#### Lista de Tips publicados hasta la fecha:

1. Modificación de los parámetros por default que rigen en los comandos del OS/400
2. Restricción de comandos pesados a modalidad batch
3. Cómo generar un entorno de prueba para año 2000
4. Cómo salvar y restaurar spool
5. Cómo agregar pantallas de confirmación/validación para comandos delicados
6. Defragmentación del espacio en disco no utilizado : STRDSKRGZ, ENDDSKRGZ
7. Manipulación de bases de datos desde programas CL, a través de Query/400
8. Generación de spool AS/400 en formato PDF (Adobe Acrobat Reader) para almacenar en CD's
9. Cómo proteger columnas de un archivo físico o lógico
10. Cómo cambiar la pantalla de signon
11. Cómo automatizar transferencias de archivos con TCP/IP desde AS/400
12. Control de accesos sobre archivos de spool
13. Aproveche lo que ya tiene: FILE SERVING con NETSERVER/400
14. EMULACION 5250 vía Internet con lo que ya tiene instalado
15. Editor alternativo: Comando EDTF (Edit File)
16. Auditoría sobre objetos en AS/400
17. Cómo personalizar los comandos del menú de petición del sistema
18. Acceso a archivos multimiembros en un entorno cliente/servidor o SQL
19. Cómo agregar opciones de usuario al producto PDM
20. Auditoría sobre usuarios en AS/400
21. Cómo obtener línea de comandos en pantallas que no la tienen.
22. Cómo enviar por e-mail objetos de QSYS.LIB
23. Cómo transferir archivos de spool a la PC usando Operations Navigator
24. Qué es el IFS y cómo accederlo
25. Curiosidades de la programación CL – Parte I
26. Cómo gestionar y controlar la seguridad a través del menú SECTOOLS – Parte I
27. Vuelco de spool a archivos de base de datos en forma automática, usando COLAS DE DATOS
28. Recursos y curiosidades de la programación CL - Parte II
29. Cómo cargar datos a tablas DB2/400 desde otros entornos con el comando CPYFRMIMPF
30. Cómo gestionar y controlar la seguridad a través del menú SECTOOLS – Parte II
31. Acción automática ante crecimiento de la ocupación de disco
32. Sometimiento de comandos remotos con SBMRMTCMD

33. Novedades para el arranque de TCP/IP en V5R1
34. Cómo controlar la ocupación de disco: comandos RTVDSKINF y PRTDSKINF
35. En la Web: Nuevo buscador de comandos CL
36. Cómo automatizar respuestas a mensajes de consulta utilizando la lista de respuestas del sistema
37. Cómo planificar trabajos batch con/sin Operations Navigator – Parte I
38. Cómo copiar perfiles de usuarios entre distintos sistemas utilizando Operations Navigator y Management Central
39. Cómo identificar trabajos servidores de TCP/IP y/o Client Access

---

## Resumen ejecutivo e Introducción

En algún momento todo programador ha ejecutado sentencias SQL, ya sea interactivamente a través del comando STRSQL, o con RPG usando cursores SQL, o inclusive desde el entorno gráfico ofrecido por iSeries Navigator (ex Operations Navigator).

El comando RUNSQLSTM (Run Structured Query Language Statement) es un comando CL que lee y procesa sentencias SQL almacenadas en un miembro fuente. Las sentencias en el miembro fuente pueden ejecutarse sin ser compiladas. Esto permite que sentencias SQL estáticas o inclusive las generadas dinámicamente puedan ejecutarse sin la necesidad de embeberlas en un lenguaje de alto nivel como RPG.

El comando RUNSQLSTM puede correr una serie de sentencias SQL, pero **está limitado a un subconjunto de sentencias SQL standard**. En la sección “*Para tener en cuenta*” se especifican todas las sentencias de ese subconjunto.

En muchas situaciones nos encontramos con la necesidad de ejecutar esas sentencias, para que sean invocadas desde un programa CL y ejecutar ese programa en entorno interactivo o batch.

Es sabido por todo usuario de AS/400 que la programación CL no permite hacer modificaciones de ningún tipo sobre archivos de bases de datos (sólo leer información con el comando RCVF). Por lo tanto, las sentencias SQL puras no podrían mezclarse con sentencias del lenguaje de control cuando es necesario realizar tareas de mantenimiento de base de datos.

Aunque el entorno de programación CL interactivo o batch sea el entorno de ejecución más común para el comando RUNSQLSTM, existen otras posibilidades de ejecución. Por ejemplo, como argumento del programa QCMDEXC.

Además, ya sea dentro de un programa CL o directamente en la línea de comandos, RUNSQLSTM puede usarse para crear Stored Procedures, usando la sentencia SQL CREATE PROCEDURE.

El proceso general para usar el comando RUNSQLSTM involucra la creación de un miembro fuente dentro de un archivo físico de fuentes (en donde el tipo de fuente no necesita ser ninguno de los tipos específicos como PF, CLP, LF o RPG, por ejemplo, sino sólo definirlo como TXT.); luego especificar la serie de sentencias SQL separadas por el carácter “;” (punto y coma) dentro de ese miembro fuente (como si se estuviera definiendo un fuente en RPG, CL, Cobol, etc), para finalmente y (opcionalmente en un programa CL) ejecutar la sentencia RUNSQLSTM mencionando ese fuente TXT, donde se encuentran las sentencias SQL a ejecutar.

---

## Consideraciones previas

- A partir de la V5R1 el comando RUNSQLSTM viene con el OS/400 (no se necesita comprar el producto DB2 Query Manager and SQL Development Kit para usarlo). Como consecuencia, las sentencias SQL Create Table, Create View, Create Index y otras sentencias se pueden usar para crear objetos de base de datos aún sin contar con el producto SQL Development Kit.

- Puesto que el comando RUNSQLSTM está esencialmente orientado al mantenimiento de base de datos y preparación de entornos de ejecución de programas, no acepta sentencias de consulta, del tipo SELECT, por lo tanto es importante tener en cuenta que la sentencia **SELECT SQL no puede ser ejecutada** con el comando RUNSQLSTM.
- A menos que el archivo sobre el que se está operando se encuentre bajo el entorno de Control de Compromiso (Commitment Control) ( y por lo tanto que el archivo esté jornalizado), debería especificarse el parámetro **COMMIT (\*NONE)** en el comando RUNSQLSTM. Si no está especificado \*NONE en este parámetro, se obtendrá un error al ejecutar las sentencias SQL contenidas en el fuente de texto, cuando sean invocadas por el comando RUNSQLSTM.
- Las sentencias SQL definidas en el miembro fuente de tipo TXT no pueden ser “prompteadas” . Esto puede ser un problema para aquellos usuarios sin conocimiento de la sintaxis de las sentencias SQL. Una buena opción para tener las sentencias sintácticamente correctas, es ejecutarlas y probarlas previamente en entorno SQL interactivo (STRSQL) y luego desde la sesión SQL salvada, copiarlas al fuente TXT que se está definiendo.
- Como se mencionó anteriormente, se puede ejecutar más de una sentencia SQL en un único miembro fuente. La manera de hacerlo es agregando al final de cada sentencia un símbolo “;” (punto y coma), excepto para la última.

## Ejemplo de uso del comando RUNSQLSTM

El uso del comando RUNSQLSTM implica realizar los siguientes pasos:

### 1) Definición del miembro fuente de tipo TXT:

Es necesario previamente crear la biblioteca y el archivo físico de fuentes donde se va a definir el nuevo miembro: Ejemplo: **TEKBIB/QXTSRC/(fuenteSQL)**

El siguiente es el grupo de sentencias SQL a definir en el miembro fuente *Fuentesql* (que va a ser procesadas por el comando RUNSQLSTM):

```

0001 -- Se realiza primero la sentencia UPDATE
0002 UPDATE teklib.clieproc SET clicred = clicred *3 where nrocli= '3';
0003 CREATE VIEW teklib.vista1 AS SELECT nomcli, domicilio, clicred
0004 FROM teklib.clieproc WHERE clicred >= 500; /* Crea una vista */

```

Donde:

- La línea 1 es un comentario. El símbolo “--” indica que todo lo que sigue a este símbolo es un comentario.
- La línea 2 es una sentencia SQL: UPDATE. Notar que la sentencia debe finalizar con un símbolo “;” (punto y coma).
- La línea 3 es una sentencia SQL: CREATE VIEW. Esto crea una vista SQL, o "Archivo lógico" sobre el AS/400, que continúa en la línea 3
- La línea 4 es la continuación de la 3. Notar que hay un segundo estilo de comentario en esta línea (es el estilo de los comentarios en fuentes CL). Esta línea también incluye el “;” al final de la sentencia SQL.

2) **Ejecutar el comando RUNSQLSTM** con los parámetros indicados para el ejemplo. Se puede definir un fuente de programa CL donde incorporarlo o elegir otro entorno de ejecución permitido para el comando:

**RUNSQLSTM SRCFILE(TEKLIB/QXTSRC) srcmbr(fuenteSQL) COMMIT(\*NONE)**

Cuando RUNSQLSTM ejecuta las sentencias SQL contenidas en el miembro fuente indicado, se genera automáticamente un archivo de spool que se envía al archivo de impresora QSYSPRT, a menos que se especifique otro archivo de impresora en el parámetro PRTFILE (Print File) del comando. El listado (archivo de spool) generado tiene el nombre del fuente que contiene las sentencias SQL (en el ejemplo: *fuentesql*).

### **Para tener en cuenta:**

- Para estar seguro que se han escrito correctamente las sentencias SQL que se desean procesar vía el comando RUNSQLSTM, usar el parámetro **PROCESS(\*SYN)** para chequear la sintaxis de las sentencias antes de que se ejecuten. Sólo “Promptear” el comando e ingresar los parámetros. Si existen errores, se recibirá un mensaje SQL9010. En este caso también se genera un archivo de spool indicando los errores de cada sentencia individualmente. Especificar el comando de la siguiente manera:

```
RUNSQLSTM SRCFILE(TEKLIB/QXTSRC) SRCMBR(FUENTESQL) COMMIT(*NONE)
PROCESS(*SYN)
```

- Cuando se usa el comando RUNSQLSTM, está disponible el **Procesador de Sentencias SQL** (el SQL Statement processor).
- El Procesador de Sentencias SQL permite que las sentencias SQL sean ejecutadas desde un miembro fuente. Las sentencias en el miembro fuente pueden ser ejecutadas o cambiadas más de una vez, sin compilar el fuente.
- Las sentencias **que pueden ser usadas** con el procesador de sentencias SQL son: ALTER TABLE, CALL, COMMENT ON, COMMIT, CREATE ALIAS, CREATE DISTINCT TYPE, CREATE FUNCTION, CREATE INDEX, CREATE PROCEDURE, CREATE SCHEMA, CREATE TABLE, CREATE TRIGGER, CREATE VIEW, DECLARE GLOBAL TEMPORARY TABLE, DELETE, DROP, GRANT, INSERT, LABEL ON, LOCK TABLE, RELEASE SAVEPOINT, RENAME, REVOKE, ROLLBACK, SAVEPOINT, SET PATH, SET SCHEMA, SET TRANSACTION, UPDATE

Copyright Agosto 2004 - Teknoda S.A. – iSeries, AS/400 y OS/400 son marcas registradas de IBM.

Dudas o consultas a : [nsalmun@teknoda.com](mailto:nsalmun@teknoda.com)