



interskill
learning

SQL and SPUFI

By proceeding with this courseware you agree with [these terms and conditions](#). Interskill Learning Pty. Ltd. © 2019





Objectives

SQL and SPUFI

In this module, you will look at invoking some of the SQL statements covered in the previous module interactively using the TSO/ISPF SPUFI facility.

After completing this module, you will be able to:

- Access SPUFI and Modify Panel Defaults
- Code and Execute SQL Statements from an Input Data Set Using SPUFI
- Interpret Output Produced from SPUFI





As you begin coding SQL you will need to test it against Db2 data. A quick method of performing this is to use the SQL Processor Using File Input (SPUFI) facility, which is available through TSO/ISPF.

The following pages describe how SPUFI is accessed and the type of information you need to provide in order to invoke your SQL statements.



```
Menu Utilities Compilers Options Status Help
-----
ISPF Primary Option Menu
Option ==> d2
                                More:  +
0 Settings      Terminal and user parameters   User ID . : IBMUSER
1 View          Display source data or listings    Time. . . : 05:50
2 Edit          Create or change source data   Terminal. : 3278
3 Utilities     Perform utility functions         Screen. . : 1
4 Foreground    Interactive language processing  Language. : ENGLISH
5 Batch         Submit job for language processing  Appl ID . : ISR
6 Command       Enter TSO or workstation commands        TSO logon : DBPROC10
7 Dialog Test   Perform dialog testing                   TSO prefix: GHPROD
8 LM Facility   Library administrator functions         System ID : S0W1
9 IBM Products  IBM program development products       MVS acct. : FB3
10 SCLM         SW configuration Library Manager        Release . : ISPF 6.1
11 workplace    ISPF Object/Action workplace

----- Other Install Products -----

MQ MQSeries    MQSeries 5.3.1
D2 DB2I        Perform DB2 Interactive functions

Enter X to Terminate using log/list defaults
```



Step 1 of 2

Because SPUFI is run through TSO/ISPF you will need to have been assigned access to this product by your security administrator who will provide you with a logon ID and password. After logging on, an ISPF Primary Option panel similar to the one shown here will be displayed.

In this example you can see option D2 at the bottom of the screen, which is used to access the SPUFI facility on this system. Note that your site's systems programmer may use other panel option selection codes, or that you may need to scroll down (PF8) in your version of this panel to locate the correct option.

Type **D2** in the Option line and **press Enter**.

```
COMMAND ==>> 1 DB2I PRIMARY OPTION MENU SSID: DB11

select one of the following DB2 functions and press ENTER.

1 SPUFI (Process SQL statements)
2 DCLGEN (Generate SQL and source language declarations)
3 PROGRAM PREPARATION (Prepare a DB2 application program to run)
4 PRECOMPILE (Invoke DB2 precompiler)
5 BIND/REBIND/FREE (BIND, REBIND, or FREE plans or packages)
6 RUN (RUN an SQL program)
7 DB2 COMMANDS (Issue DB2 commands)
8 UTILITIES (Invoke DB2 utilities)
D DB2I DEFAULTS (Set global parameters)
X EXIT (Leave DB2I)

PRESS: END to exit HELP for more information
```



Step 2 of 2

You can see that a number of interactive Db2 functions can be accessed from the resulting DB2I Primary Option Menu. The Db2 subsystem which will perform those functions is identified in the top right of this, and subsequent menu panels. In this case, it is DB11. The identity of the subsystem is just one of the variables which may be set via the panel option: D DB2I DEFAULTS.

Type **1** in the command line and **press Enter** so you can access SPUFI.



```
====>                                SPUFI                                SSID: DB11

Enter the input data set name:         (Can be sequential or partitioned)
1  DATA SET NAME ... ==>
2  VOLUME SERIAL ... ==>             (Enter if not cataloged)
3  DATA SET PASSWORD ==>           (Enter if password protected)

Enter the output data set name:        (Must be a sequential data set)
4  DATA SET NAME ... ==>

Specify processing options:
5  CHANGE DEFAULTS ==> NO           (Y/N - Display SPUFI defaults panel?)
6  EDIT INPUT ..... ==> YES        (Y/N - Enter SQL statements?)
7  EXECUTE ..... ==> YES           (Y/N - Execute SQL statements?)
8  AUTOCOMMIT ..... ==> YES        (Y/N - Commit after successful run)
9  BROWSE OUTPUT ... ==> YES       (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ==>

PRESS:  ENTER to process  END to exit  HELP for more info
```

PDS Members	
CREATE TABLE	
SELECT	
INSERT INTO	
DELETE	
UPDATE	

The SPUFI panel allows you to provide details on the input, output and general SPUFI defaults to be used when invoking your SQL statements against Db2 table data and indexes.

Firstly you should create a data set to store the SQL statements you need to test. This can be a sequential data set, but if you need to test different types of statements you may want to create a partitioned data set so that unique code can be stored in different members.

```
====>                                SPUFI                                SSID: DB11

Enter the input data set name:          (Can be sequential or partitioned)
1 DATA SET NAME ... ====> 'XAT8.SQLTEST(UPDATE)'
2 VOLUME SERIAL ... ====>          (Enter if not cataloged)
3 DATA SET PASSWORD ====>         (Enter if password protected)

Enter the output data set name:        (Must be a sequential data set)
4 DATA SET NAME ... ====>

Specify processing options:
5 CHANGE DEFAULTS ... ====> NO      (Y/N - Display SPUFI defaults panel?)
6 EDIT INPUT ..... ====> YES       (Y/N - Enter SQL statements?)
7 EXECUTE ..... ====> YES          (Y/N - Execute SQL statements?)
8 AUTOCOMMIT ..... ====> YES       (Y/N - Commit after successful run?)
9 BROWSE OUTPUT ... ====> YES       (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ====>

PRESS:  ENTER to process   END to exit           HELP for more information
```

You will enter some SQL statements later but for now, you will code your SQL statements into a data set called XAT8.SQLTEST in a member called UPDATE.

This data set name and member needs to be coded within single quotes in the input data set name field. If the name is spelt incorrectly or you forget to include the single quotes, an error message will be displayed when SPUFI attempts to execute the SQL.

```
====>                                SPUFI                                SSID: DB11
====>
Enter the input data set name:         (Can be sequential or partitioned)
1 DATA SET NAME ... ====> 'XAT8.SQLTEST(UPDATE)'
2 VOLUME SERIAL ... ====>          (Enter if not cataloged)
3 DATA SET PASSWORD ====>         (Enter if password protected)

Enter the output data set name:        (Must be sequential)
4 DATA SET NAME ... ====> RESULT ←
Specify processing options:
5 CHANGE DEFAULTS ... ====> NO      (Y/N - Display SPUFI defaults panel?)
6 EDIT INPUT ..... ====> YES       (Y/N - Enter SQL statements?)
7 EXECUTE ..... ====> YES          (Y/N - Execute SQL statements?)
8 AUTOCOMMIT ..... ====> YES       (Y/N - Commit after successful run?)
9 BROWSE OUTPUT ... ====> YES       (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ====>

PRESS:  ENTER to process   END to exit           HELP for more information
```

If your logon ID is XAT8 then the data set referenced here would be XAT8.RESULT

The result from your executed SQL statements will be stored in a sequential data set specified in option 4. If the data set name you provide is an existing one, data in it will be overwritten each time you use it. If the data set name does not exist, it will be created automatically for you.

If you forget to insert the single quotes around the data set name, the data set referenced will contain your logon ID as the high level qualifier.



```
====>                                SPUFI                                SSID: DB11

Enter the input data set name:         (Can be sequential or partitioned)
1  DATA SET NAME ... ====> 'XAT8.SQLTEST(UPDATE)'
2  VOLUME SERIAL ... ====>          (Enter if not cataloged)
3  DATA SET PASSWORD ====>        (Enter if password protected)

Enter the output data set name:       (Must be a sequential data set)
4  DATA SET NAME ... ====> RESULT

Specify processing options:
5  CHANGE DEFAULTS  ====> yes       (Y/N - Display SPUFI defaults panel?)
6  EDIT INPUT ..... ====> YES      (Y/N - Enter SQL statements?)
7  EXECUTE .....   ====> YES      (Y/N - Execute SQL statements?)
8  AUTOCOMMIT ..... ====> YES      (Y/N - Commit after successful run?)
9  BROWSE OUTPUT ... ====> YES      (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ====>

PRESS:  ENTER to process   END to exit           HELP for more information
```

There may be cases where the default processing values used by SPUFI need to be changed before your SQL statements are processed.

By setting the value for option 5 to Y or YES, the SPUFI default panel will be displayed before executing your SQL statements.

Type this value now and **press Enter** to display this panel.



```

====> CURRENT SPUFI DEFAULTS SSID: DB11
Enter the following to control your SPUFI session:
1 SQL TERMINATOR .. ==> ; (SQL Statement Terminator)
2 ISOLATION LEVEL ==> RR (RR=Repeatable Read, CS=Cursor Sensitivity, UR=Uncommitted Read)
3 MAX SELECT LINES ==> 250 (Max lines to be returned)
4 ALLOW SQL WARNINGS==> NO (Continue fetching after warning)
5 CHANGE PLAN NAMES ==> NO (Change the plan names used)
6 SQL FORMAT..... ==> SQL (SQL, SQLCOMNT, or SQLPL)
Output data set characteristics:
7 SPACE UNIT ..... ==> TRK (TRK or CYL)
8 PRIMARY SPACE ... ==> 6 (Primary space allocation)
9 SECONDARY SPACE .. ==> 5 (Secondary space allocation)
10 RECORD LENGTH ... ==> 4092 (LRECL=Logical record length)
11 BLOCK SIZE ..... ==> 4096 (Size of one block)
12 RECORD FORMAT ... ==> VB (RECFM=F, FB, FBA, V, VB, or VBA)
13 DEVICE TYPE ..... ==> SYSDA (Must be DASD unit name)
Output format characteristics:
14 MAX NUMERIC FIELD ==> 33 (Maximum width for numeric fields)
15 MAX CHAR FIELD .. ==> 80 (Maximum width for character fields)
16 COLUMN HEADING .. ==> NAMES (NAMES, LABELS, ANY or BOTH)
17 FOR BIT DATA ... ==> ASIS (ASIS or HEX)
PRESS: ENTER to process END to exit HELP for more information

```

There are several characters that can not be used as the SQL terminator:

- blank
- comma
- double or single quote
- left or right parenthesis
- underscore

The pages that follow contain details on the defaults you can overtype and update on the CURRENT SPUFI DEFAULTS panel.

As you have seen in previous modules, the default SQL statement terminator is a semicolon. If your SQL statement does not contain this character as part of its data processing, then SPUFI will not know when to terminate the statement.

```
====>                                CURRENT SPUFI DEFAULTS                                SSID: DB11
Enter the following to control your SPUFI session:
1  SQL TERMINATOR ..====> ;           (SQL Statement Terminator)
2  ISOLATION LEVEL  ..====> RR         (RR=Repeatable Read, CS=Cursor Stability,
                                         UR=Uncommitted Read)
3  MAX SELECT LINES ..====> 250       (Max lines to be return from SELECT)
4  ALLOW SQL WARNINGS ..====> NO      (Continue fetching after sqlwarning)
5  CHANGE PLAN NAMES ..====> NO      (Change the plan names used by SPUFI)
6  SQL FORMAT..... ..====> SQL       (SQL, SQLCOMNT, or SQLPL)
Output data set characteristics:
7  SPACE UNIT ..... ..====> TRK      (TRK or CYL)
8  PRIMARY SPACE ... ..====> 6       (Primary space allocation 1-999)
9  SECONDARY SPACE . ..====> 5       (Secondary space allocation 0-999)
10 RECORD LENGTH ... ..====> 4092    (LRECL=Logical record length)
11 BLOCK SIZE ..... ..====> 4096    (Size of one block)
12 RECORD FORMAT ... ..====> VB      (RECFM=F, FB, FBA, V, VB, or VBA)
13 DEVICE TYPE .... ..====> SYSDA    (Must be DASD unit name)
Output format characteristics:
14 MAX NUMERIC FIELD ..====> 33      (Maximum width for numeric fields)
15 MAX CHAR FIELD .. ..====> 80      (Maximum width for character fields)
16 COLUMN HEADING .. ..====> NAMES   (NAMES, LABELS, ANY or BOTH)
17 FOR BIT DATA ... ..====> ASIS    (ASIS or HEX)
PRESS: ENTER to process  END to exit  HELP for more information
```

SQL statements can be executed at various isolation levels to prevent them from affecting other applications that are running. Refer to the IBM manual for additional information on this topic.

If you are using your SQL SELECT statements for testing purposes, you may want to limit the amount of data that is returned from it. The MAX SELECT LINES field can contain a number from 1 to 99999999 and will prevent any more lines than the number specified from being selected.

Mouse-over the other highlighted options in this section for a description of their purpose.

```
====>                                CURRENT SPUFI DEFAULTS                                SSID: DB11
====>
Enter the following to control your SPUFI session:
 1 SQL TERMINATOR .. ==> ;              (SQL statement Terminator)
 2 ISOLATION LEVEL  ==> RR              (RR=Repeatable Read, CS=Cursor stability,
                                         UR=Uncommitted Read)
 3 MAX SELECT LINES ==> 250            (Max lines to be return from SELECT)
 4 ALLOW SQL WARNINGS==> NO            (Continue fetching after sqlwarning)
 5 CHANGE PLAN NAMES ==> NO            (Change the plan names used by SPUFI)
 6 SQL FORMAT..... ==> SQL            (SQL, SQLCOMNT, or SQLPL)
Output data set characteristics:
 7 SPACE UNIT ..... ==> TRK            (TRK or CYL)
 8 PRIMARY SPACE ... ==> 6            (Primary space allocation 1-999)
 9 SECONDARY SPACE . ==> 5            (Secondary space allocation 0-999)
10 RECORD LENGTH ... ==> 4092         (LRECL=Logical record length)
11 BLOCK SIZE ..... ==> 4096         (Size of one block)
12 RECORD FORMAT ... ==> VB           (RECFM=F, FB, FBA, V, VB, or VBA)
13 DEVICE TYPE .... ==> SYSDA         (Must be DASD unit name)
Output format characteristics:
14 MAX NUMERIC FIELD ==> 33           (Maximum width for numeric fields)
15 MAX CHAR FIELD .. ==> 80           (Maximum width for character fields)
16 COLUMN HEADING .. ==> NAMES        (NAMES, LABELS, ANY or BOTH)
17 FOR BIT DATA ... ==> ASIS         (ASIS or HEX)
PRESS: ENTER to process   END to exit   HELP for more information
```



If the output produced from your SQL is to be used for other purposes such as being fed as input into another program, you may need to change some of the output data set characteristics or output format characteristics. Normally once all defaults have been defined in this panel, you will not need to access it every time you process your SQL so the CHANGE DEFAULTS option from the previous panel can be set to NO.

The F3 key can be used to exit from this panel and return to the main SPUFI panel, or if you have made any required changes, the Enter key can be pressed to continue processing of your SQL statements.

Press F3 to return to the SPUFI panel.



```
SPUFI                                SSID: DB11
====>
DSNE800A NO DEFAULT VALUES WERE CHANGED. PRESS ENTER TO CONTINUE
Enter the input data set name:        (Can be sequential or partitioned)
1 DATA SET NAME ... ==> 'XAT8.SQLEST(UPDATE)'
2 VOLUME SERIAL ... ==>
3 DATA SET PASSWORD ==>            (Enter if password protected)

Enter the output data set name:      (Must be a sequential data set)
4 DATA SET NAME ... ==> RESULT

Specify processing options:
5 CHANGE DEFAULTS ==> *              (Y/N - Display SPUFI defaults panel?)
6 EDIT INPUT ..... ==> YES          (Y/N - Enter SQL statements?)
7 EXECUTE ..... ==> YES             (Y/N - Execute SQL statements?)
8 AUTOCOMMIT ..... ==> YES         (Y/N - Commit after successful run?)
9 BROWSE OUTPUT ... ==> YES        (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ==>

PRESS:  ENTER to process   END to exit           HELP for more information
```

In this scenario there were no changes made to the defaults, and a message informing you of that fact is displayed at the top of the SPUFI panel.

The CHANGE DEFAULTS field is also changed to * to indicate that it has been processed and that when the Enter key is pressed the defaults panel will not be displayed.



```
====>                                SPUFI                                SSID: DB11
DSNE800A NO DEFAULT VALUES WERE CHANGED. PRESS ENTER TO CONTINUE
Enter the input data set name:         (Can be sequential or partitioned)
1  DATA SET NAME ... ==> 'XAT8.SQLTEST(UPDATE)'
2  VOLUME SERIAL ... ==>          (Enter if not cataloged)
3  DATA SET PASSWORD ==>        (Enter if password protected)

Enter the output data set name:       (Must be a sequential data set)
4  DATA SET NAME ... ==> RESULT

Specify processing options:
5  CHANGE DEFAULTS ==> *          (Y/N - Display SPUFI defaults panel?)
6  EDIT INPUT ..... ==> YES      (Y/N - Enter SQL statements?)
7  EXECUTE ..... ==> YES         (Y/N - Execute SQL statements?)
8  AUTOCOMMIT ..... ==> YES      (Y/N - Commit after successful run?)
9  BROWSE OUTPUT ... ==> YES     (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ==>

PRESS:  ENTER to process   END to exit   HELP for more information
```



Step 1 of 2

There are two methods of entering your SQL statements through SPUFI. You can create your input data set or data set member and using ISPF, code all your required statements prior to invoking them through SPUFI. Another method is to type YES into the EDIT INPUT field so that when SPUFI is invoked, the input data set will be opened using ISPF Edit and you can type in the statements required for that invocation.

This option is already set to YES on this panel so **press Enter** so you can see how SPUFI displays the input data.

```
====>                                SPUFI                                SSID: DB11
DSNE808A EDIT SESSION HAS COMPLETED. PRESS ENTER TO CONTINUE
Enter the input data set name:        (Can be sequential or partitioned)
1 DATA SET NAME ... ==> 'XAT8.SQLEST(UPDATE)'
2 VOLUME SERIAL ... ==>          (Enter if not cataloged)
3 DATA SET PASSWORD ==>          (Enter if password protected)

Enter the output data set name:      (Must be a sequential data set)
4 DATA SET NAME ... ==> RESULT

Specify processing options:
5 CHANGE DEFAULTS ==> *           (Y/N - Display SPUFI defaults panel?)
6 EDIT INPUT ..... ==> *         (Y/N - Enter SQL statements?)
7 EXECUTE ..... ==> YES          (Y/N - Execute SQL statements?)
8 AUTOCOMMIT ..... ==> YES      (Y/N - Commit after successful run?)
9 BROWSE OUTPUT ... ==> YES      (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ==>

PRESS:  ENTER to process   END to exit       HELP for more information
```

The message at the top of the SPUFI panel indicates that the editing has been performed on the input data set. Normally you will want to begin processing the SQL statements that you have entered in your input data set, so the EXECUTE field should be set to Y or YES before pressing the Enter key.

If this field is set to NO and BROWSE OUTPUT field is set to YES, then no SQL processing will be performed but any existing data in the output data set specified will be displayed.

```
====>                                SPUFI                                SSID: DB11
DSNE808A EDIT SESSION HAS COMPLETED. PRESS ENTER TO CONTINUE
Enter the input data set name:          (Can be sequential or partitioned)
1 DATA SET NAME ... ==> 'XAT8.SQLEST(UPDATE)'
2 VOLUME SERIAL ... ==>
3 DATA SET PASSWORD ==>

Enter the output data set name:         (Must be a sequential data set)
4 DATA SET NAME ... ==> RESULT

Specify processing options:
5 CHANGE DEFAULTS ... ==> *           (Y/N - Display SPUFI defaults panel?)
6 EDIT INPUT ..... ==> *            (Y/N - Enter SQL statements?)
7 EXECUTE ..... ==> YES              (Y/N - Execute SQL statements?)
8 AUTOCOMMIT ..... ==> YES          (Y/N - Commit after successful run?)
9 BROWSE OUTPUT ... ==> YES         (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ==>

PRESS:  ENTER to process   END to exit       HELP for more information
```

If the EXECUTE field is set to YES in your SPUFI panel, the next stage when you press Enter will see the execution of your SQL statements.

If there are many statements in your input data set or the table you are interrogating is large, it may take considerable time to process the statements. If this is unacceptable, you can interrupt the processing using the PA1 key with the amount of output created depending on how much processing the statement(s) were able to perform before being interrupted.

If SPUFI terminates while executing your statements it could indicate that Db2 resource limits have been defined by your Db2 administrator and that your SQL execution has exceeded them.


```
====>                                SPUFI COMMIT OR ROLLBACK                SSID: DB11
DSNE361I SPUFI PROCESSING COMPLETE
Explanation:

The last SQL statement ran with a return code of -204
Because AUTOCOMMIT was set to NO, you must now decide whether to:

* COMMIT (save) your database changes or to
* ROLLBACK (erase) your database changes or to
* DEFER the decision, which will allow you to continue processing
  another input data set. If you select DEFER, it is possible
  that you will tie up system resources that others need.

1  NEXT ACTION ==> COMMIT      (Enter COMMIT, ROLLBACK, or DEFER)

PRESS:  ENTER to process                                HELP for more information
```

Normally when processing SQL statements you will want the results saved to the Db2 tables if they have executed successfully. If you are testing statements then you can be given the option to commit them to the database by specifying N or NO in the AUTOCOMMIT field on the SPUFI panel. This will result in the screen shown here being displayed.



```
====>                                SPUFI                                SSID: DB11

Enter the input data set name:         (Can be sequential or partitioned)
1 DATA SET NAME ... ==>> 'XAT8.SQLTEST(UPDATE)'
2 VOLUME SERIAL ... ==>>
3 DATA SET PASSWORD ==>>

Enter the output data set name:       (Must be a sequential data set)
4 DATA SET NAME ... ==>> RESULT

Specify processing options:
5 CHANGE DEFAULTS ... ==>> *          (Y/N - Display SPUFI defaults panel?)
6 EDIT INPUT ..... ==>> *          (Y/N - Enter SQL statements?)
7 EXECUTE ..... ==>> YES          (Y/N - Execute SQL statements?)
8 AUTOCOMMIT ..... ==>> YES       (Y/N - Commit after successful run?)
9 BROWSE OUTPUT ... ==>> YES       (Y/N - Browse)

For remote SQL processing:
10 CONNECT LOCATION ==>> TESTDB

PRESS:  ENTER to process   END to exit           HELP for more information
```

If SPUFI is unable to connect to your specified CONNECT LOCATION, then the output data set will display return codes and error messages relating to the attempt.

If your SQL statements need to be processed on a remote database server, then the location of that server can be entered into the CONNECT LOCATION field.





You know how to define your input and output data sets to SPUFI.

You can modify your SPUFI session defaults.



You know how to interrupt SQL processing once it has begun executing.

You can submit your SQL statements to a remote database.

In this section you have seen how SPUFI is used to execute SQL statements and the defaults and processing options that can be invoked as part of this process.

In the next section you will look at the format of output produced by SPUFI and how to interpret messages displayed.





```
====>                                SPUFI                                SSID: DB11

Enter the input data set name:          (Can be sequential or partitioned)
1  DATA SET NAME ... ===> 'XAT8.SQLTEST(UPDATE)'
2  VOLUME SERIAL ... ===>              (Enter if not cataloged)
3  DATA SET PASSWORD ===>             (Enter if password protected)

Enter the output data set name:         (Must be a sequential data set)
4  DATA SET NAME ... ===> RESULT

Specify processing options:
5  CHANGE DEFAULTS ... ===> NO         (Y/N - Display SPUFI defaults panel?)
6  EDIT INPUT ..... ===> NO          (Y/N - Enter SQL statements?)
7  EXECUTE ..... ===> YES             (Y/N - Execute SQL statements?)
8  AUTOCOMMIT ..... ===> YES         (Y/N - Commit after successful run?)
9  BROWSE OUTPUT ... ===> YES        (Y/N - Browse output data set?)

For remote SQL processing:
10 CONNECT LOCATION ===>

PRESS:  ENTER to process   END to exit           HELP for more information
```

Now that you have identified the input and output data sets to be used by SPUFI you will look at interpreting the output produced from the execution of your SQL statements.

With the EXECUTE field set to YES, **press Enter** to execute the SQL statements in the XAT8.SQLTEST(UPDATE) data set.



```
Menu utilities Compilers Help Browsing data set XAT8.RESULT.
BROWSE XAT8.RESULT Line 00000000 Col 001 080
Command =====> Scroll =====> CSR
***** Top of Data *****
-----+-----+-----+-----+-----+-----+
SELECT * FROM CHICAGO.EMP;
-----+-----+-----+-----+-----+-----+
EMPLYNO  FNAME      LNAME      DEPT      DOB      SALARY
-----+-----+-----+-----+-----+-----+
000001   THOMAS      FAARLT     SALES     1985-04-17 52500
000002   TY          NGUYEN     MARKETTING 1986-12-08 41000
000003   SONIA      ELLIOT     SUPPORT    1990-08-07 37750
000004   JOHN       GEYER      SUPPORT    1961-05-27 35000
000006   REECE      TERN       DEVELOPMENT 1977-01-12 41850
000007   EVA        OPOLSKY    SALES     1985-06-30 51200
000010   GREG       WILSON     SALES     1985-04-14 44800
000011   WILMA      WILSON     SALES     1985-02-18 39850
000014   MOHINDER   SINGH      PRODUCTION 1988-02-26 40025
000015   EVAN       WILSON     SUPPORT    1977-07-01 36660
000016   DOLORES    QUINTANA   NETWORK    1980-09-28 49920
000018   HEATHER    HAMILTON   HR          1987-12-24 45850
000019   BRUCE      IANSON     OPERATIONS 1983-11-19 41075
000021   ANTHIA     STOCKLEY   MARKETTING 1987-10-03 43580
```

Displays all information from the CHICAGO.EMP table.

Column/field names and their values are displayed.

The ISPF Browse facility is opened and the result of your SQL execution is displayed. At the top of the output produced, the following data is displayed:

- A copy of the SQL statements from the input data set
- The data produced as a result of the processed SQL statements





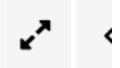
```
Menu Utilities Compilers Help
-----
BROWSE      KATE.RESULT      Line 00000000 Col 001 080
Command    =====>           Scroll ==> CSR
000183     LI                1991-10-14
000184     PAULA             MASONI      MARKETTING 1979-05-24
DSNE610I  NUMBER OF ROWS DISPLAYED IS 169
DSNE616I  STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 100
-----
UPDATE CHICAGO.EMP
SET SALARY = SALARY * 1.05;
-----
DSNE615I  NUMBER OF ROWS AFFECTED IS 169
DSNE616I  STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 0
-----
DSNE617I  COMMIT PERFORMED, SQLCODE IS 0
DSNE616I  STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 0
-----
DSNE601I  SQL STATEMENTS ASSUMED TO BE BETWEEN COLUMNS 1 AND 72
DSNE620I  NUMBER OF SQL STATEMENTS PROCESSED IS 2
DSNE621I  NUMBER OF INPUT RECORDS READ IS 4
DSNE622I  NUMBER OF OUTPUT RECORDS WRITTEN IS 191
***** Bottom of Data *****
```

Updating the salary values
in CHICAGO.EMP table.

Successfully processed SELECT
statements invoked through SPUFI,
return a SQL code of 100.

An SQL code of 0 for any other
SQL statements indicates that
processing was successful.

As you scroll towards the bottom of the data, you may encounter warning or error messages associated with the invocation, and will also see summary statistics.



```
Menu Utilities Compilers Help
-----
BROWSE XAT8.RESULT Line 00000000 Col 001 080
Command =====> Scroll =====> CSR
***** Top of Data *****
-----
SELECT * FROM CHIG.EMP; 00010012
-----
DSNT408I SQLCODE = -204, ERROR: CHIG.EMP IS AN UNDEFINED NAME
DSNT418I SQLSTATE = 42704 SQLSTATE RETURN CODE
DSNT415I SQLERRP = DSNX0TL SQL PROCEDURE DETECTING ERROR
DSNT416I SQLERRD = -500 0 0 -1 0 0 SQL DIAGNOSTIC INFORMATION
DSNT416I SQLERRD = X'FFFFFF0C' X'00000000' X'00000000' X'00000000'
X'00000000' X'00000000' SQL DIAGNOSTIC INFORMATION
-----
DSNE618I ROLLBACK PERFORMED, SQLCODE IS 0
DSNE616I STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 0
-----
DSNE601I SQL STATEMENTS ASSUMED TO BE BETWEEN COLUMNS 1 AND 80
DSNE620I NUMBER OF SQL STATEMENTS PROCESSED IS 1
DSNE621I NUMBER OF INPUT RECORDS READ IS 1
DSNE622I NUMBER OF OUTPUT RECORDS WRITTEN IS 17
***** Bottom of Data *****
```

This message occurs when an error is encountered and the AUTOCOMMIT field on the SPUFI panel is set to YES. Rollback will undo any data changes made since the last commit.

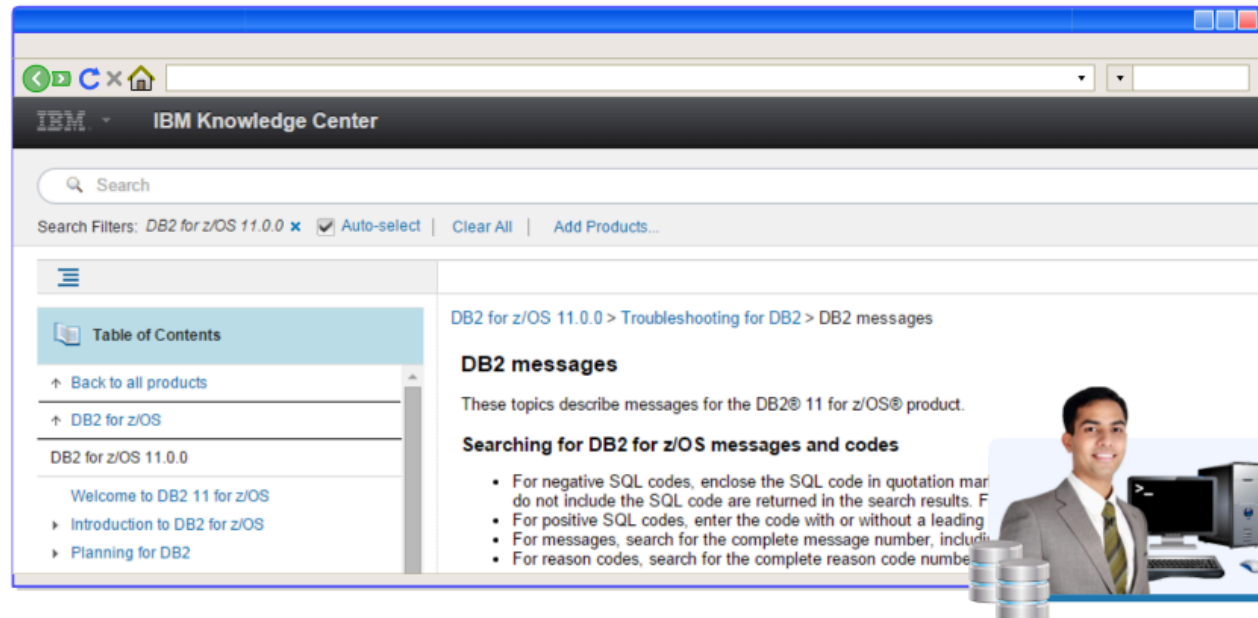
The example shown here highlights a message indicating that there is a problem. The message says that the table name specified in the SELECT statement is undefined, which indicates that a table of that name could not be located.

As a result of this error, rollback processing is performed.

```
Menu Utilities Compilers Help
-----
BROWSE      XAT8.RESULT                               Line 00000000 Col 001 080
Command ==>>                                         Scroll ==>> CSR
***** Top of Data *****
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
CREATE TABLE COURSE5.DET                               00010001
  (EMPNO CHAR(6) NOT NULL,                               00020001
  COURSENM VARCHAR(36) NOT NULL,                        00030001
  SCORE CHAR(6),                                       00040001
  ADMRDEPT CHAR(3) NOT NULL,                           00050001
  LOCATION CHAR(16),                                    00060001
  PRIMARY KEY(EMPNO);                                  00070001
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
DSNT408I  SQLCODE = 104, ERROR: ILLEGAL SYMBOL "<END-OF-STATEMENT>". SOME
          SYMBOLS THAT MIGHT BE LEGAL ARE: , )
DSNT418I  SQLSTATE = 42601 SQLSTATE RETURN CODE
DSNT415I  SQLERRP = DSNHPARS SQL PROCEDURE DETECTING ERROR
DSNT416I  SQLERRD = 3 0 0 -1 452 502 SQL DIAGNOSTIC INFORMATION
DSNT416I  SQLERRD = X'00000003' X'00000000' X'00000000' X'FFFFFFF'
          X'000001C4' X'000001F6' SQL DIAGNOSTIC INFORMATION
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
DSNE618I  ROLLBACK PERFORMED, SQLCODE IS 0
DSNE616I  STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS 0
```

In this example the messages indicate that there is an illegal symbol being used to terminate one of the SQL statements. In fact, the problem is associated with a missing corresponding closed parenthesis on the last line.

This highlights that at times it can be difficult to determine the actual cause of a problem, from the messages displayed.



Additional information on messages produced by Db2 can be obtained through IBM's Db2 for z/OS 11 Knowledge Center portal.



```
File Edit Edit_Settings Menu Utilities Compilers Test Help
-----
EDIT      XA8T.CNTL(XABSPUFI) - 01.03                Columns 00001 00072
Command  ==>>>                                     Scroll ==>> CSR
***** ***** Top of Data *****
000100 //XABSPUFI JOB CLASS=A,MSGCLASS=X,REGION=64M
000200 //DSN EXEC PGM=IKJEFT01,DYNAMNBR=20
000300 //STEPLIB DD DSN=DSN110.SDSNLOAD,DISP=SHR
000310 //SYSPRINT DD SYSOUT=*
000400 //SYSPRT DD SYSOUT=*
000500 //SYSTSIN DD *
000600 DSN SYSTEM(DB11)
000700 RUN PROGRAM(DSNTEP2) PLAN(DSNTEP1) -
000800 LIB('DSN110.DB11.RUNLIB.LOAD') -
000810 PARM('/ALIGN(LHS) MIXED TOLWARN(YES)')
000900 END
001000 //SYSIN DD *
001100 SELECT * FROM CHICAGO.EMP;
001200 /*
***** ***** Bottom of Data *****
```

Before concluding this module on SPUFI, it should be mentioned that there is a batch equivalent that you may prefer to use if you have a large amount of SQL statements you need to invoke or there is a big Db2 table you need to access. Several batch programs are available that perform functions similar to those found in SPUFI, with DSNTEP2 and DSNTEP4 being the most popular.

An example of batch SPUFI is shown above.



Summary

SQL and SPUFI

In this module, you discovered how SPUFI can be used to invoke SQL statements interactively.

You should now be able to:

- Access SPUFI and Modify Panel Defaults
- Code and Execute SQL Statements from an Input Data Set Using SPUFI
- Interpret Output Produced from SPUFI



interskill
learning

Db2 Fundamentals Mastery Test

By proceeding with this courseware you agree with [these terms and conditions](#). Interskill Learning Pty. Ltd. © 2019

