



interskill  
learning

## Using SPUFI and IBM Data Studio to Access Db2

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





## Objectives

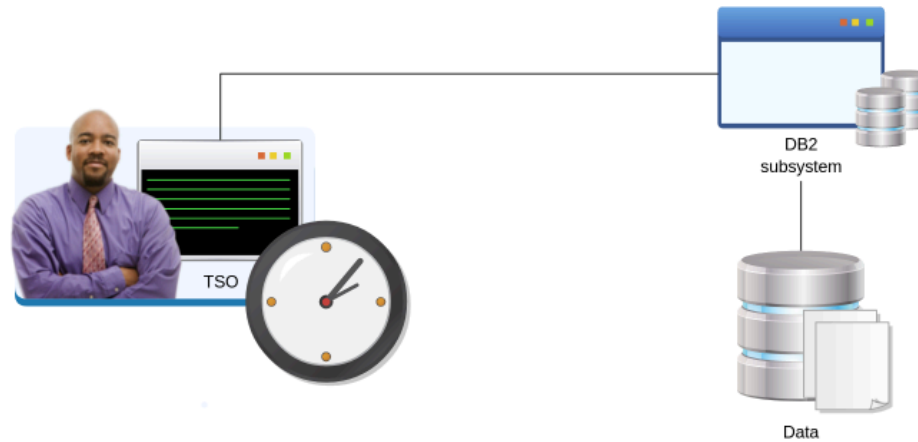
---

### Using SPUFI and IBM Data Studio to Access Db2

Db2 provides tools to execute SQL statements on Db2 database and resources. IBM Data Studio is the most common, and can be used with Db2 on all platforms. SPUFI is used on z/OS.

After completing this module, you should be able to:

- Use SPUFI to Run SQL
- Use IBM Data Studio to Run SQL



---

Db2 can be accessed via a TSO session if the TSO session has been set up to access the Db2 system. This is usually done through the addition of an extra option called DB2I (Db2 Interactive) on the Primary Option Menu. DB2I provides access to SPUFI which stands for SQL Processor Using File Input.

When Db2 is accessed from TSO it takes advantage of the dialog management functions of ISPF and is presented in a panel-driven format. The following pages will show you what the TSO screen looks like when the Db2 interactive option is available to the session.

**Click Play** to see how Db2 can be accessed via a TSO session.

```
Menu utilities Compilers Options status Help
-----
ISPF Primary Option Menu
Option ==> d2
More: -
IN INSPECT      INSPECT for c/370 and PL/I      User ID . : IBMUSER
MQ MQSeries     MQSeries 7.1.0                  Time . . : 17:18
D2 DB2I         Perform DB2 Interactive functions Terminal. : 3278
DM DB2ADM       DB2 Admin Tool                  Screen . : 1
PM DB2PM        DB2 Performance Monitor for DB2 Language. : ENGLISH
P6 DB2PE        DB2 Performance Expert for DB2 V6 Appl ID . : ISR
DC DB2DC        DATABASE Data Collection        TSO logon : DBPROCBG
QM QMF          Query Management Facility for DB2 TSO prefix: IBMUSER
QT QMFTOOL      QMF Message Tool               System ID : S0w1
CO CICS PD/MVS  CICS PD/MVS Online Customization MVS acct. : FB3
CP CICS PD/MVS  CICS Problem Determination/MVS  Release . : ISPF 7.1
OP OPC/ESA      Operations Planning/Control/ESA
W WLM           work Load Manager Administration
RS RRS          Resource Recovery Services
F File Manager  File Manager for z/OS
FD FM/DB2       File Manager/DB2
FI FM/IMS       File Manager/IMS

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



If you have TSO access and Db2 is installed, your administrator will have most likely configured the ISPF Primary Option Menu to display available Db2 products. In this example, the menu items have been scrolled several pages to display among other things, option D2 which provides access to Db2 Interactive tools and utilities, SPUFI being one of them.

Note that this menu option may be alternatively named in your environment.

Type **D2** in the Option selection field and **press Enter**.

```
COMMAND ==>> _          DB2I PRIMARY OPTION MENU          SSID: DBBG

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
```

This option allows you to enter an SQL statement and then execute it.

This is the DB2I Primary Option Menu. You will be looking at the first option in this menu - SPUFI (Process SQL statements). This option allows you to enter an SQL statement and then execute it.

**Mouse-over** over the different menu options to discover what the other options do.

Note: Your site may have more or less options than what is displayed here, depending on the Db2 tools and facilities you have installed.



```
COMMAND ==> 1
DB2I PRIMARY OPTION MENU          SSID: DBBG
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 1 of 2

Now you will take a look at SPUFI.

Type **1** to access SPUFI and **press Enter**.



```
COMMAND ==>>                                SPUFI                                SSID: DBBG

Enter the input data set name:                (Can be sequential or partitioned)
1 DATA SET NAME ... ==>> lrnr1.sql.spufi(ex1)
2 VOLUME SERIAL ... ==>>
3 DATA SET PASSWORD ==>>
Enter the output data set name:              (Must be a sequential data set)
4 DATA SET NAME ... ==>>

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

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

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



### Step 1 of 2

The first item is an input data set where your SQL statements reside and an output data set where you want the resulting output to go.

The input data set must be pre-allocated and can be either sequential or partitioned. Using a partitioned data set allows the creation of new members or the use of an existing member.

Type **'LRNR1.SQL.SPUFI(EX1)'** to set this as the input data set and **press Tab** to move to the next item.



```
COMMAND ==>>                                SPUFI                                SSID: DB9G

Enter the input data set name:                (Can be sequential or partitioned)
1 DATA SET NAME ... ==>> 'LRNR1.SQL.SPUFI(EX1)'
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 ... ==>> SPUFI.OUTPUT

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

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

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

Type **Y** in all five option selection fields, **pressing Tab** to move between the fields and **press Enter** to process.







```
====>                                CURRENT SPUFI DEFAULTS                                SSID: DBBG

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

The maximum width of a numeric value column in your output.

Because you entered Y for CHANGE DEFAULTS on the SPUFI screen, you will now see the CURRENT SPUFI DEFAULTS screen.

These defaults can be changed but those shown here are normally used. After setting the defaults, you do not need to access this screen again.

**Mouse-over** the options for more information.





```
====>                                CURRENT SPUFI DEFAULTS                                SSID: DBBG

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

Step 1 of 6

**Press** Enter to continue.





```
COMMAND ==>>                                SPUFI                                SSID: DBBG
DSNE808A EDIT SESSION HAS COMPLETED. PRESS ENTER TO CONTINUE
Enter the input data set name:                (Can be sequential or partitioned)
1 DATA SET NAME ... ==>> 'LRNR1.SQL.SPUFI(EX1)'
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 ... ==>> SPUFI.OUTPUT

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?)

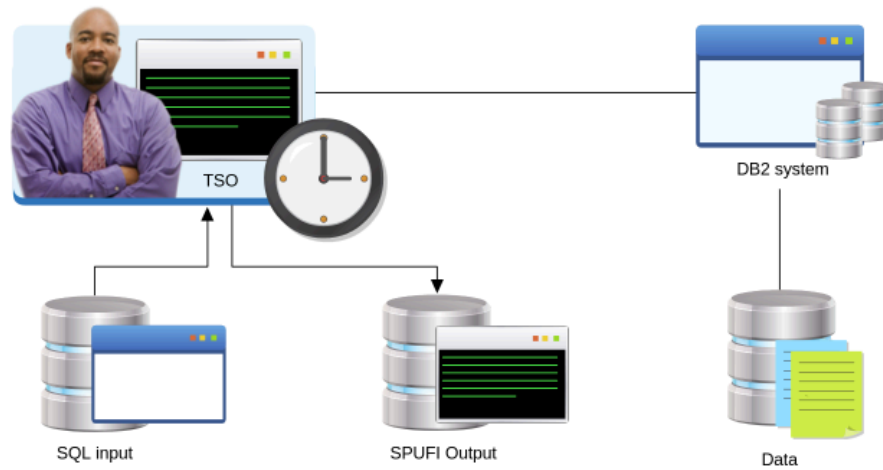
For remote SQL processing:
10 CONNECT LOCATION ==>>

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

### Step 3 of 6

Again you are presented with the SPUFI input screen. At this point you could change any of the settings or even set the EDIT INPUT back to Y and return to the EDIT screen.

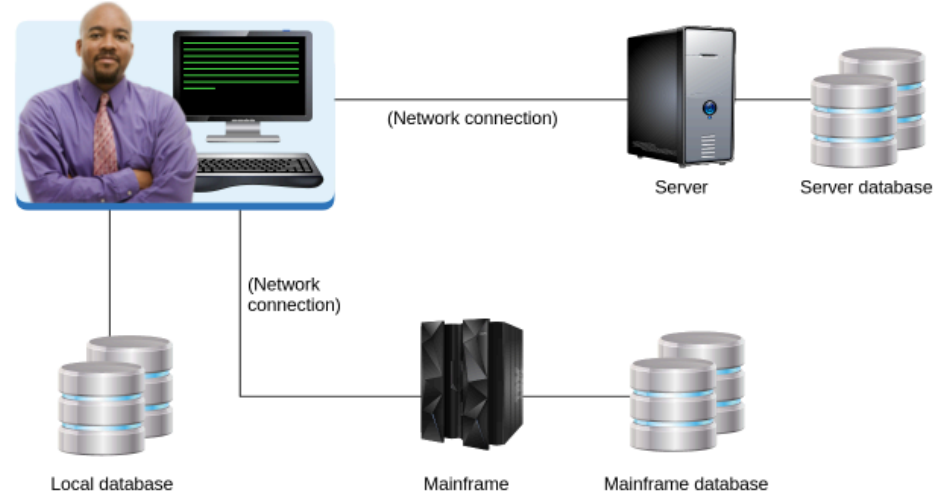
In this example, do as the prompt suggests and **press Enter**.



---

You can use SPUFI to enter ad-hoc SQL statements. Whether they run successfully will depend on the access granted to your userid.

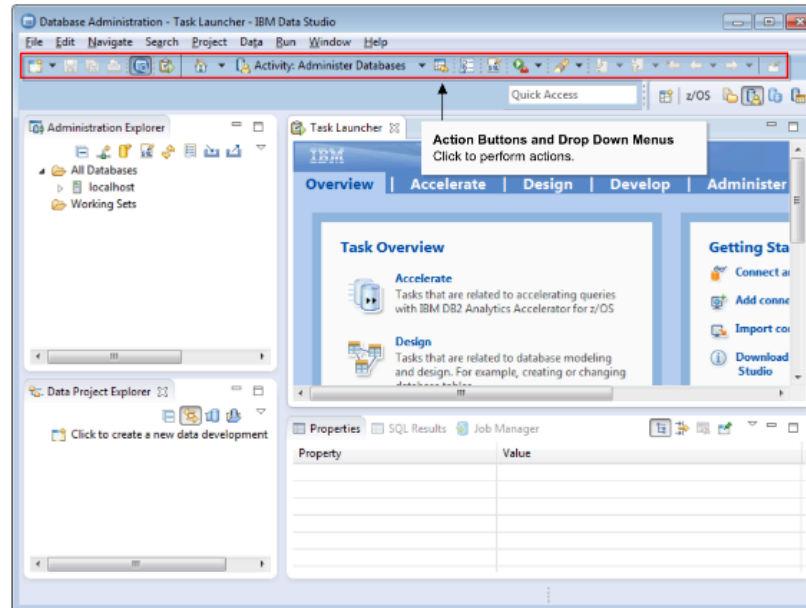
SPUFI also allows you to build up a library of data sets or members, which contains commonly used statements that can be accessed again and again by specifying the input data set. This can also be used to store the output data set for later printing or examination.



IBM Data Studio provides the user with a Windows-like interface, enabling access to Db2 data stored on:

- Local databases (Db2 databases residing on the same system that IBM Data Studio is installed)
- Remote databases (Db2 databases residing on Linux, UNIX, or Windows systems)
- Remote mainframe databases (Db2 databases residing on z/OS and IBM i)

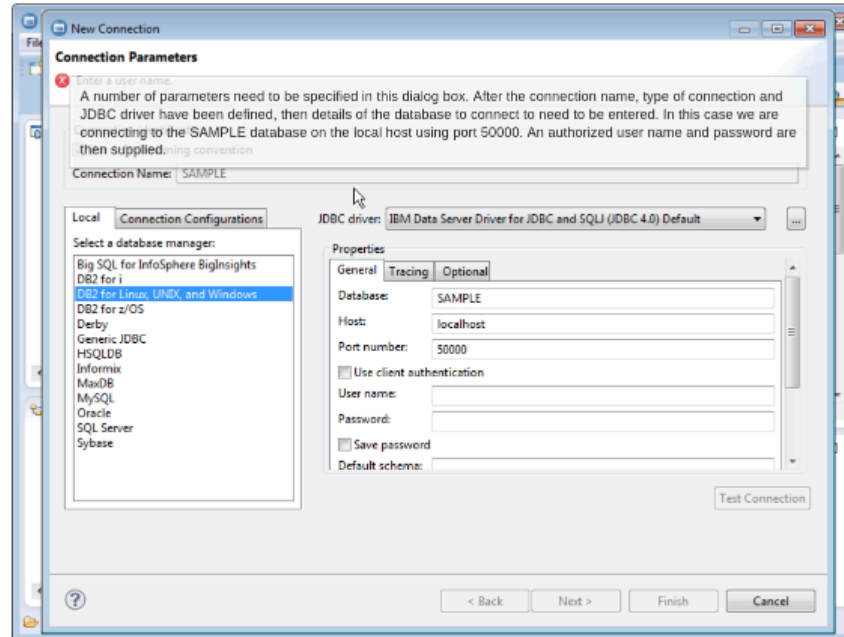




When starting IBM Data Studio for the first time you will be presented with a dialog box asking for the workspace location, which is where all your work will be saved. After this, a screen similar to the one shown here will be displayed.

It consists of several smaller windows, called views which can be added, modified and deleted, and then saved as a perspective that can be opened or closed at any time.

**Click Play** to see the features of this screen.



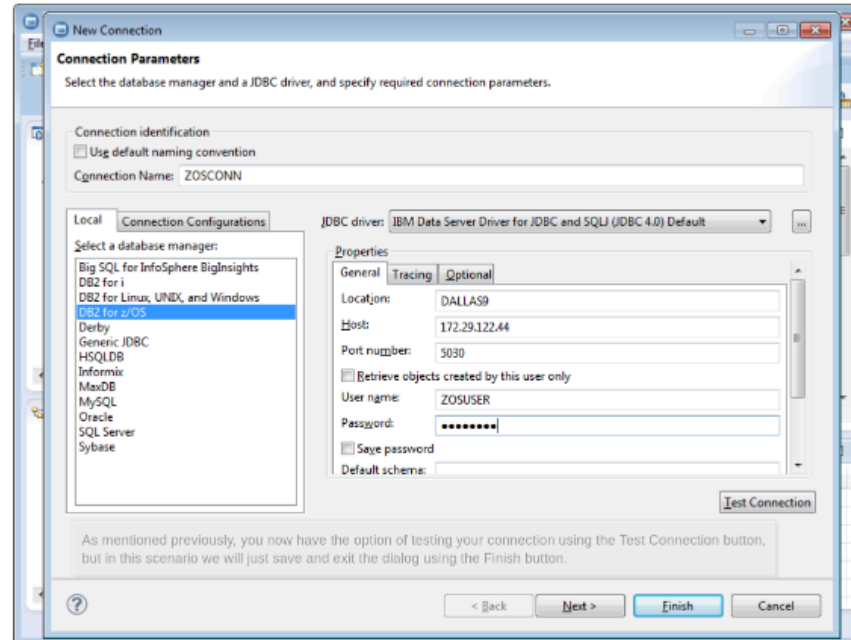
Now that IBM Data Studio has been started, it needs to access a database so that you can interact with its data. There are several methods used to perform this task.

**Click Play** to see how to connect to a local database that is running on the same system as IBM Data Studio.





## Use IBM Data Studio to Execute SQL > Remote Databases



The process to connect to a Db2 database on z/OS is similar.

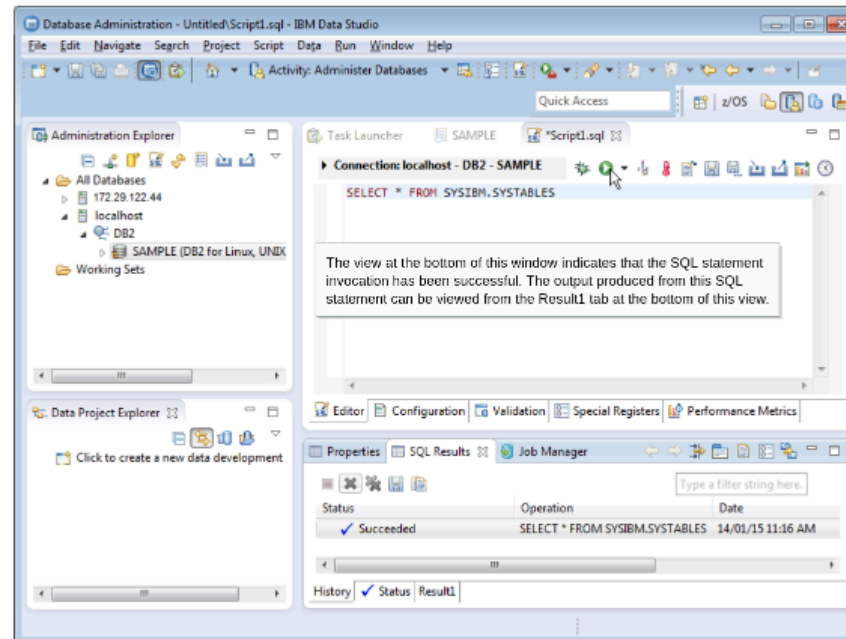
**Click Play** to see how this is performed.







## Use IBM Data Studio to Execute SQL > Execute SQL



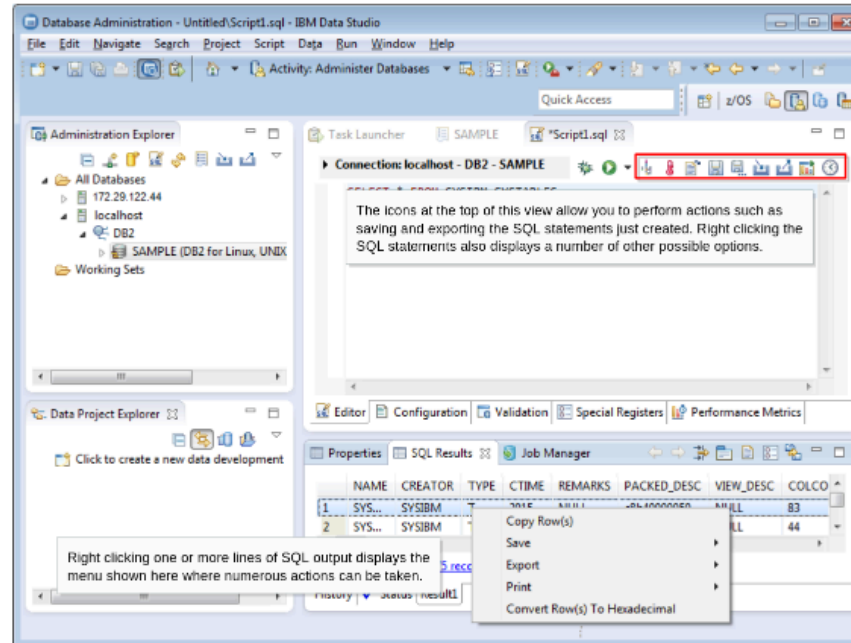
Once a connection to a database has been made and you have the correct authorization, you are able to enter SQL statements.

**Click Play** to see how SQL statements are entered for the localhost connection that was created earlier.





## Use IBM Data Studio to Execute SQL > Saving SQL

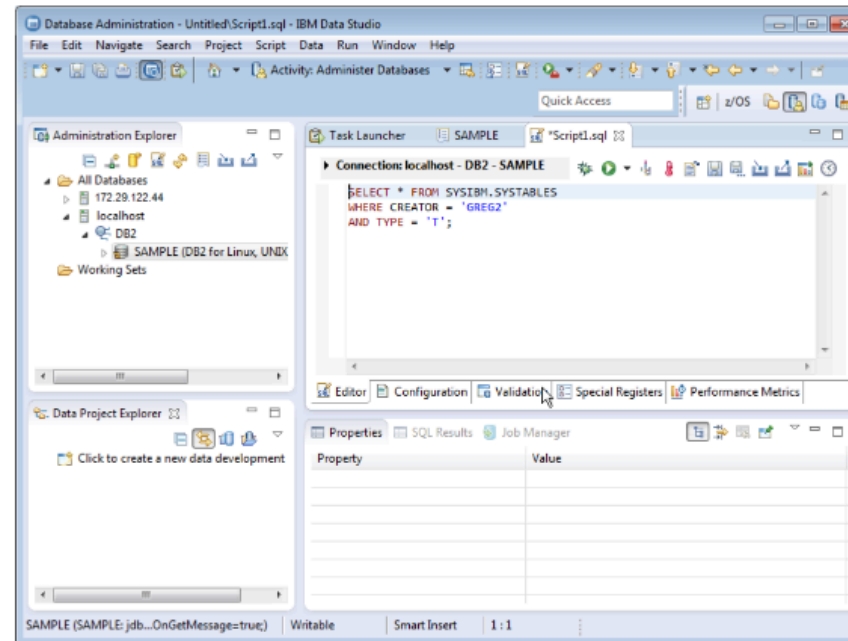


A number of actions can be performed against the SQL statements and the resulting output.





## Use IBM Data Studio to Execute SQL > Importing SQL



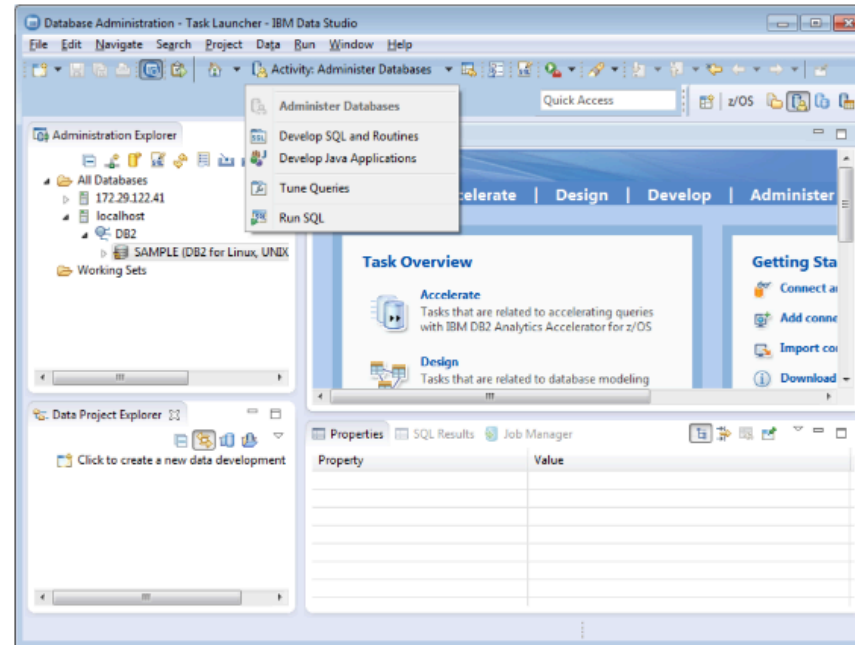
Previously saved SQL statements can also be imported into IBM Data Studio and run. In this scenario, a request for new SQL script has been invoked.

**Click Play** to see how to import existing SQL code.





## Use IBM Data Studio to Execute SQL > Other Features



Creating and running SQL statements is only one of many tasks that can be performed by IBM Data Studio. It is also capable of performing tasks such as: administering databases, monitoring database health, running jobs, developing database applications and tuning queries.

