

# Using JDBC Data Source with Microsoft SQL Server 2005

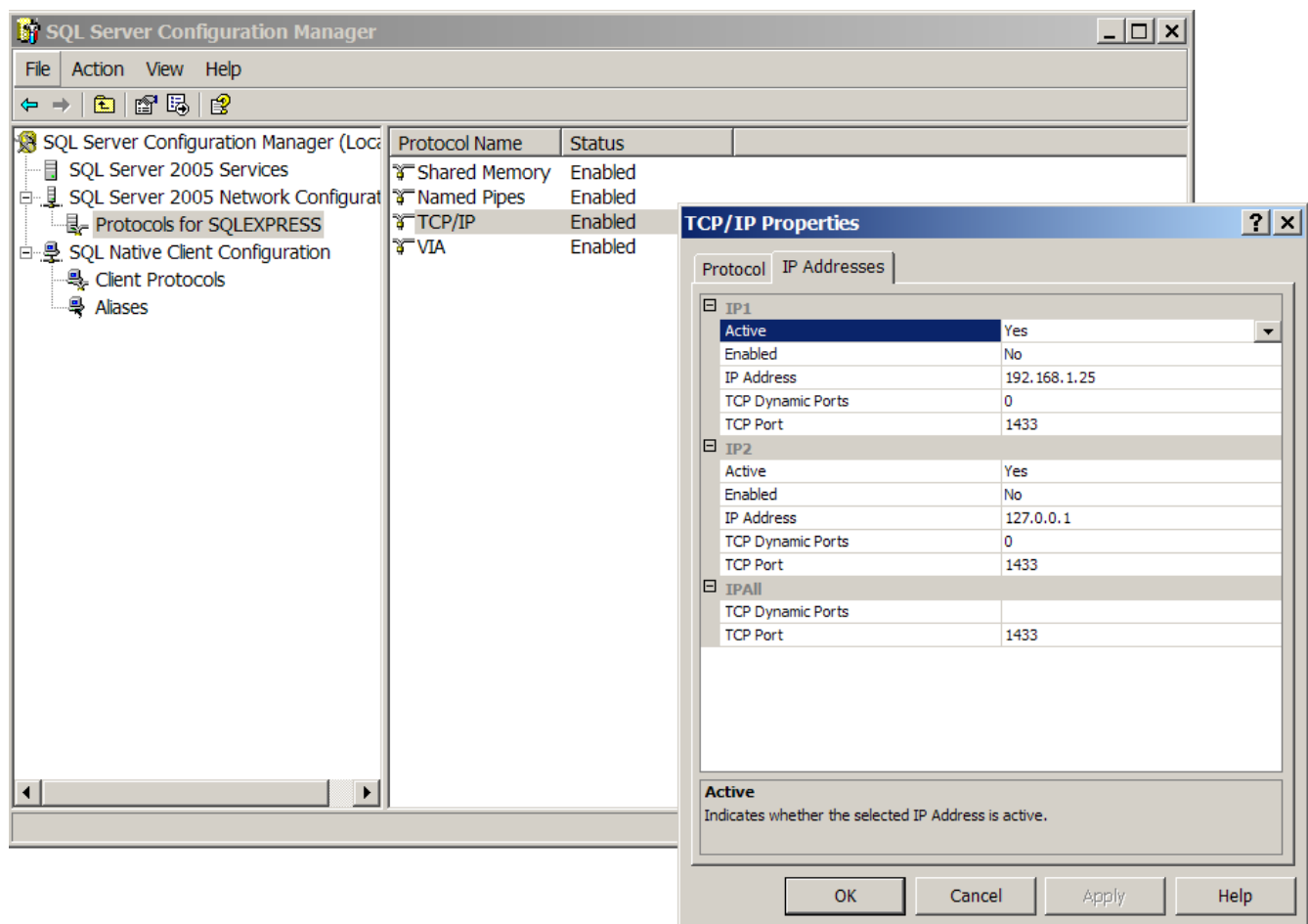
Elixir Repertoire support connectivity to Microsoft SQL Server 2005 via Microsoft SQL Server 2005 JDBC driver.

## Prerequisite

Below list the prerequisite that is need to ensure a successful connection to Microsoft SQL Server.

1) You need to enable mixed mode security when you install the server so that you can connect using user name and password.

2) The JDBC provider only works with the TCP/IP protocol which is disabled by default on Sql Express. You need to enable the TCP/IP Protocol from the SQL Server Configuration Manager that ships with SqlExpress and re-start the server. Look under SQL Server 2005 Network Configuration -> Protocols for SQLEXPRESS-> TCP/IP->Enable.

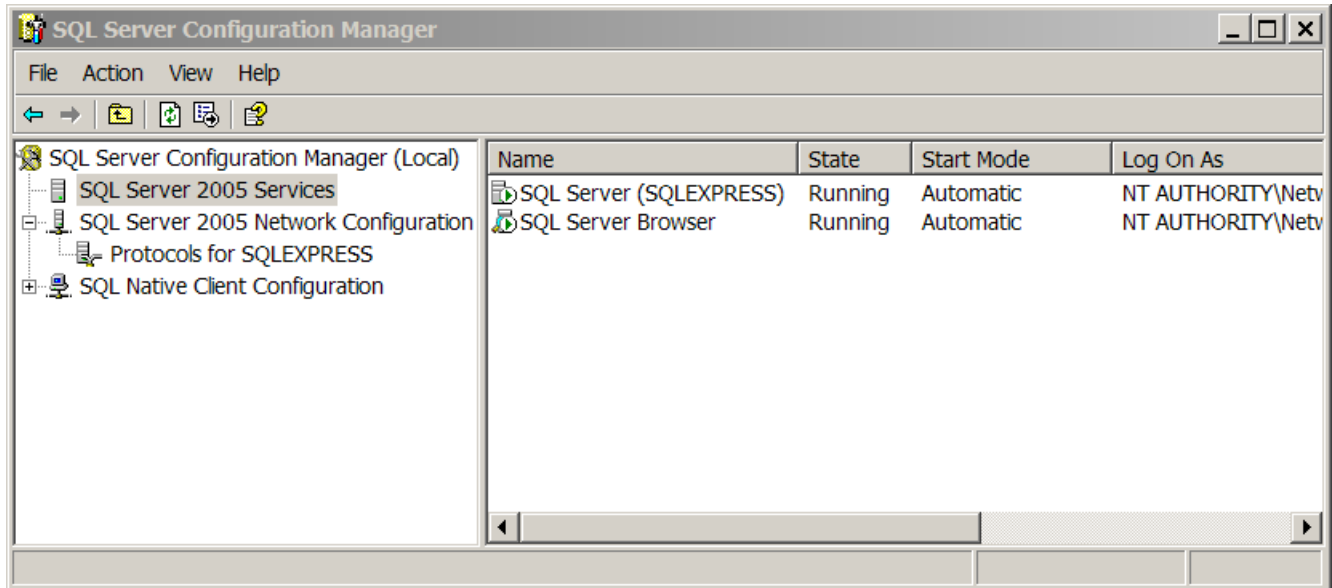


3) Change TCP dynamic port to use static port.

You may set the static TCP port. Default port is **1433**.

This is done by unset the IP addresses->TCP Dynamic->Ports as show in the location above.

4) Ensure that the SQL Server services are started



5) Download the MS SQL Server JDBC driver from Microsoft web site.

The last known location is at

<http://www.microsoft.com/downloads/details.aspx?familyid=e22bc83b-32ff-4474-a44a-22b6ae2c4e17&displaylang=en>

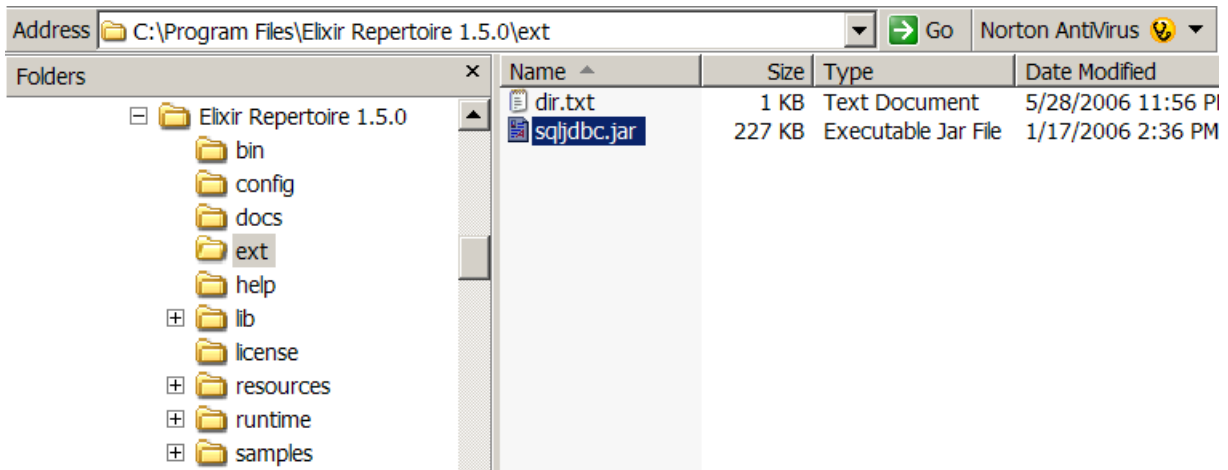
or you can do a Google search for Microsoft S SQL Server 2005 JDBC driver download.

6) Test Connection (Optional)

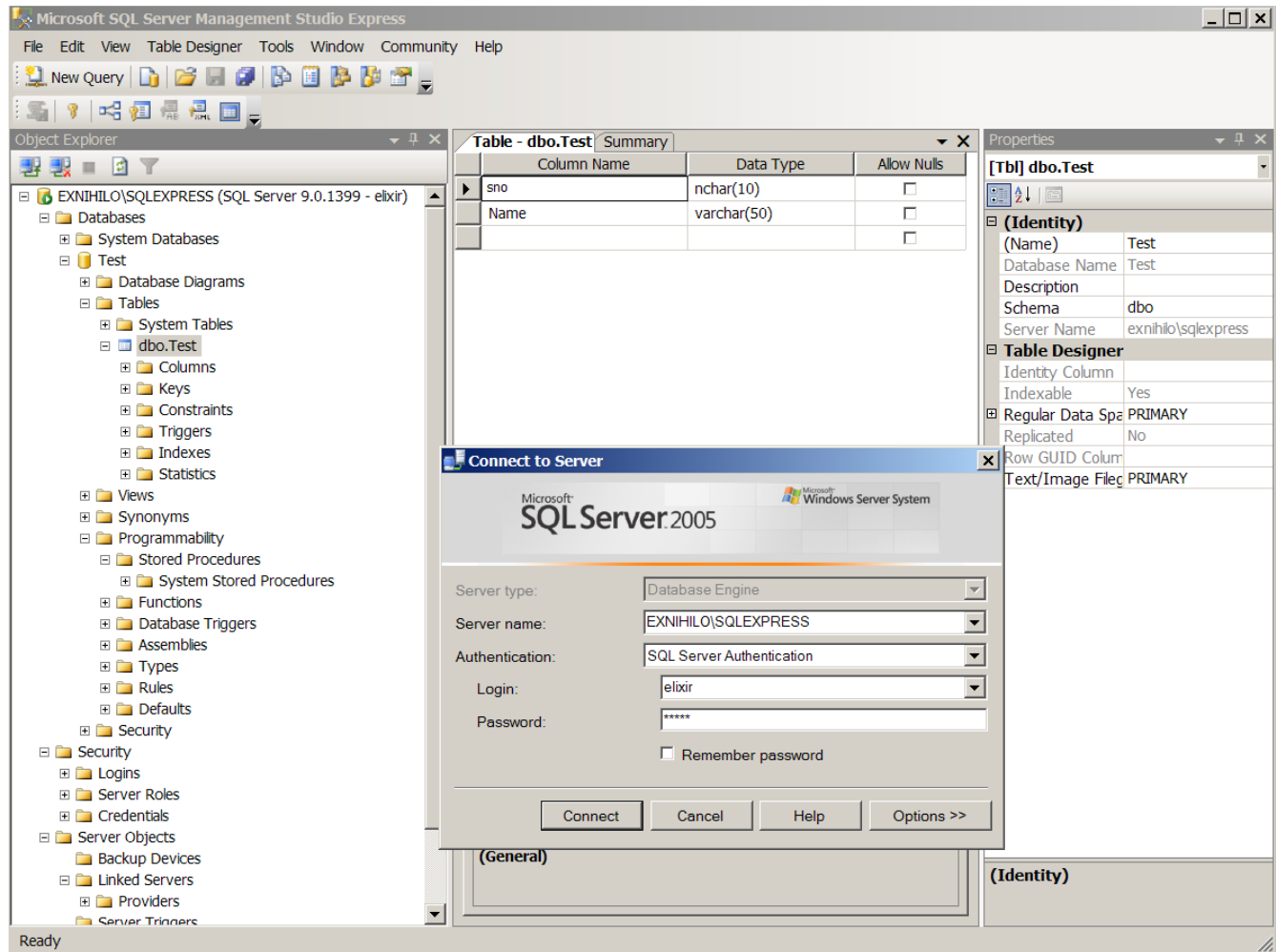
You can test the connection by write a simple Java code as

```
import java.sql.*;  
  
public class TestMSJDBCdriver{  
  
    public static void main(String[] args) {  
        try{  
            java.lang.Class.forName\("com.microsoft.sqlserver.jdbc.SQLServerDriver"\);  
            Connection c =  
                java.sql.DriverManager.getConnection(  
                    "jdbc:sqlserver://localhost\\SQLEXPRESS;user=myuser;password=mypass;");  
            System.out.println("Connected!");  
        }catch(Exception ex){  
            ex.printStackTrace();  
        }  
    }  
}
```

7) Deploy the driver into Elixir Repertoire ext directory. You have to the same if you are using Elixir Repertoire Server.



8) You can view the tables in Microsoft SQL Server 2005 Management Studio



## Creating a JDBC Data Source in Elixir Repertoire

These are the following step to create JDBC Data Source in Elixir Repertoire.

### 1) Obtain the JDBC connection parameters

They are the Microsoft server database location, database name , port (1433 default), user name and password. Please note the driver name and URL has been modified in SQL Server 2005.

Here is the different show in the table.

| <i>Server</i> | <i>MS SQL Server 2005</i>                    | <i>MS SQL Server 2000</i>                    |
|---------------|--|--|
| JDBC Driver   | com.microsoft.sqlserver.jdbc.SQLServerDriver | com.microsoft.jdbc.sqlserver.SQLServerDriver |
| URL           | jdbc:sqlserver://                            | jdbc:microsoft:sqlserver://                  |

Please refer to the driver read me information for reasons for the change.

**Datasource Wizard**

**Define JDBC Datasource**  
Enter JDBC datasource parameters

Name: MSSQL SQLServer 2005 Data Source  
Description:

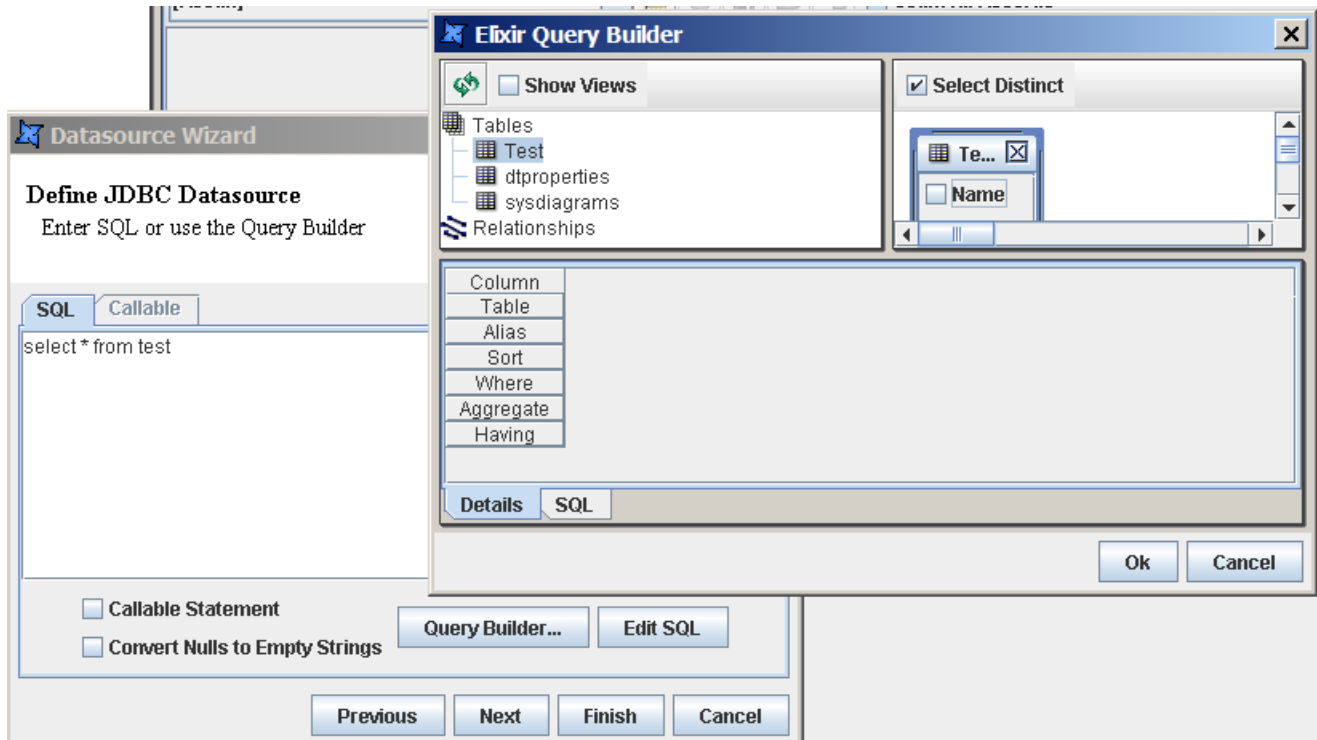
**JDBC** | JNDI

Driver Suggestions: [Dropdown]  
Driver: com.microsoft.sqlserver.jdbc.SQLServerDriver  
URL: jdbc:sqlserver://EXNHILO:1433;DatabaseName=Test

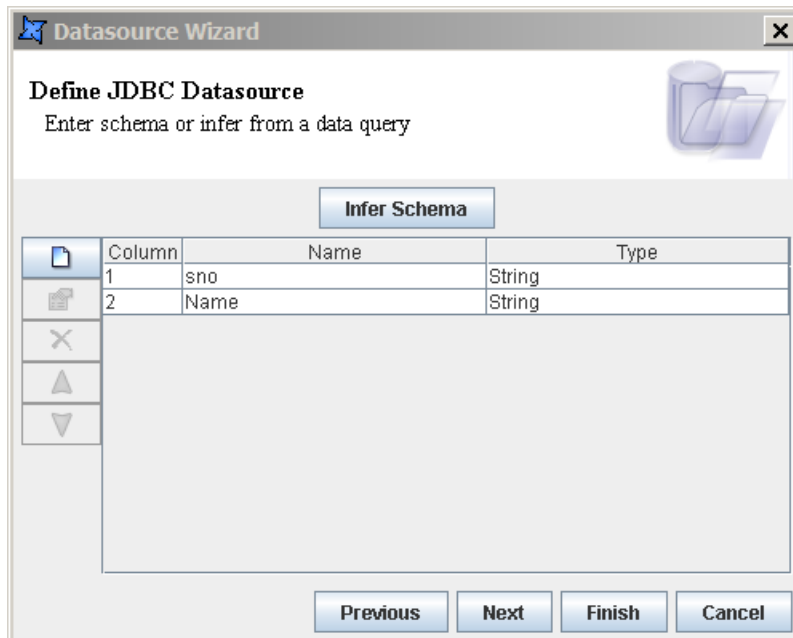
User: elixir  
Password: \*\*\*\*\*  Hide Password

Previous Next Finish Cancel

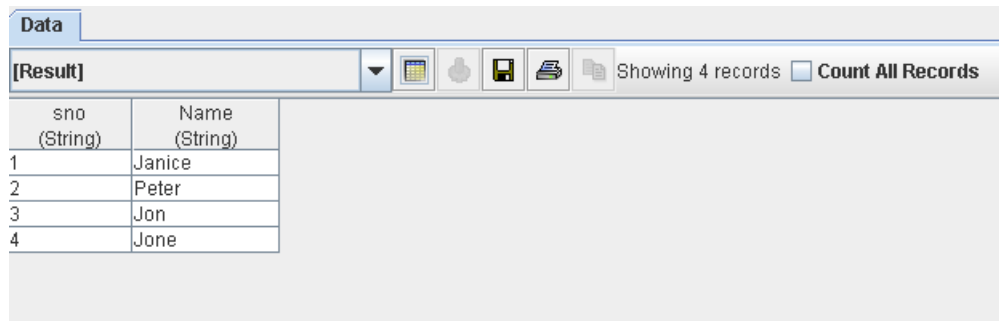
2) Step to the next page to obtain the SQL Statement as



3) Infer the schema



4) Finally you test the result of the SQL Statement in the Data->Load Data.



The screenshot shows a software interface for viewing data. At the top, there is a tab labeled "Data". Below the tab is a header area with the text "[Result]" on the left, a dropdown arrow, and several icons (grid, refresh, save, print, copy) in the center. On the right of the header, it says "Showing 4 records" followed by a checkbox labeled "Count All Records". The main area contains a table with two columns: "sno (String)" and "Name (String)". The table has four rows of data.

| sno<br>(String) | Name<br>(String) |
|-----------------|------------------|
| 1               | Janice           |
| 2               | Peter            |
| 3               | Jon              |
| 4               | Jone             |

## Using SQL Server Store Procedure

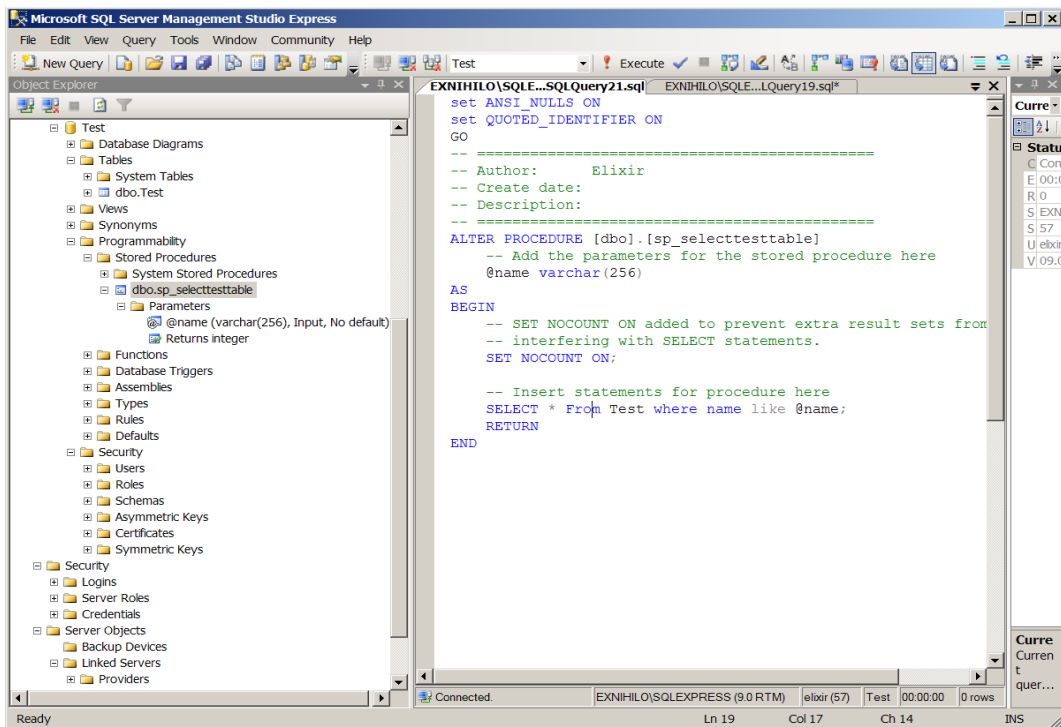
You can use SQL Server store procedure with Elixir Repertoire JDBC data source.

1) Define the Store Procedure as

```
set ANSI_NULLS ON
set QUOTED_IDENTIFIER ON
GO
=====
-- Author:                Elixir
-- Create date:
-- Description:
=====
ALTER PROCEDURE [dbo].[sp_selecttesttable]
    -- Add the parameters for the stored procedure here
    @name varchar(256)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    SELECT * From Test where name like @name;
    RETURN
END
```

where **dbo.sp\_selecttesttable** is the procedure name and it takes in a parameter name of type varchar. The return type is the resultset.



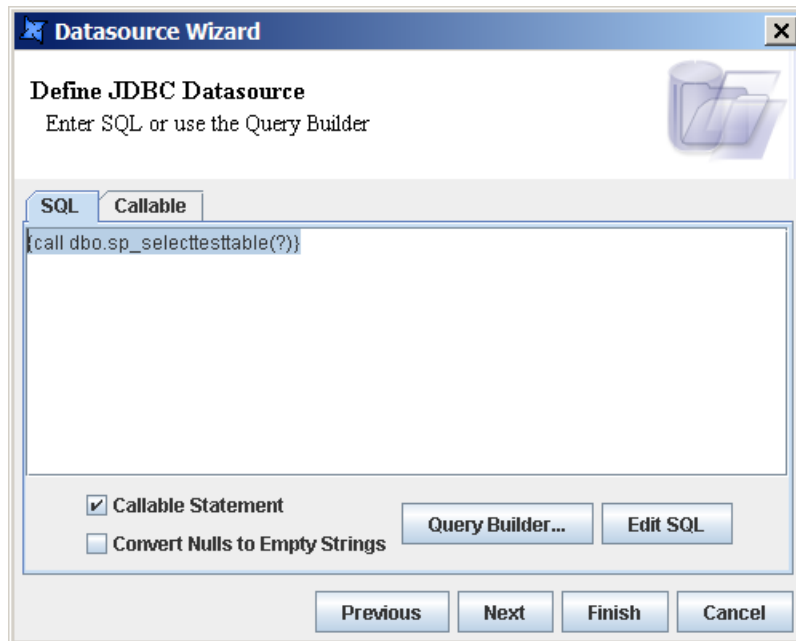
2) Test the SQL Store procedure. Once it is return back the right resultset you can create your data source.

3) At the JDBC SQL entry wizard page, enters the store procedure as

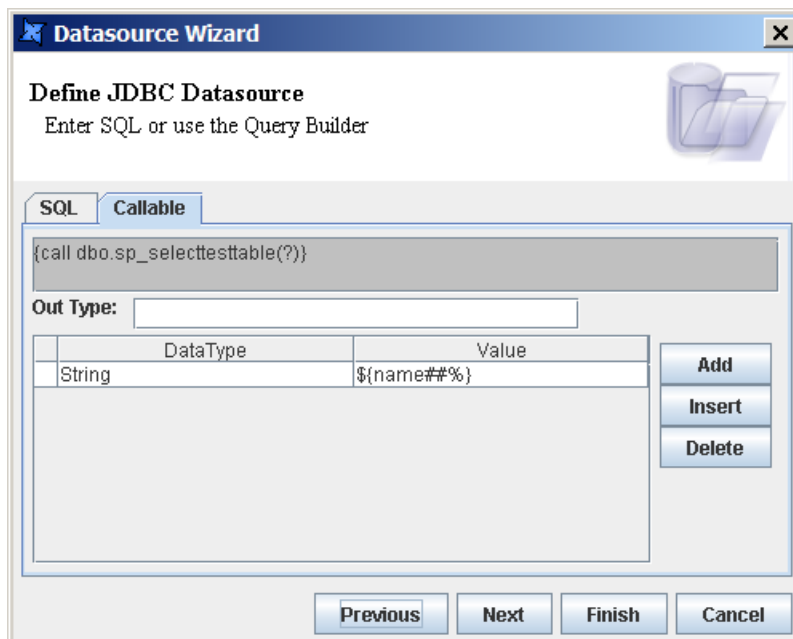
where call procedure-name([parameter],[parameter]...) to indicate the parameters as ?  
example

```
{call dbo.sp_selecttesttable(?)}
```

4) Enable this as callable statement



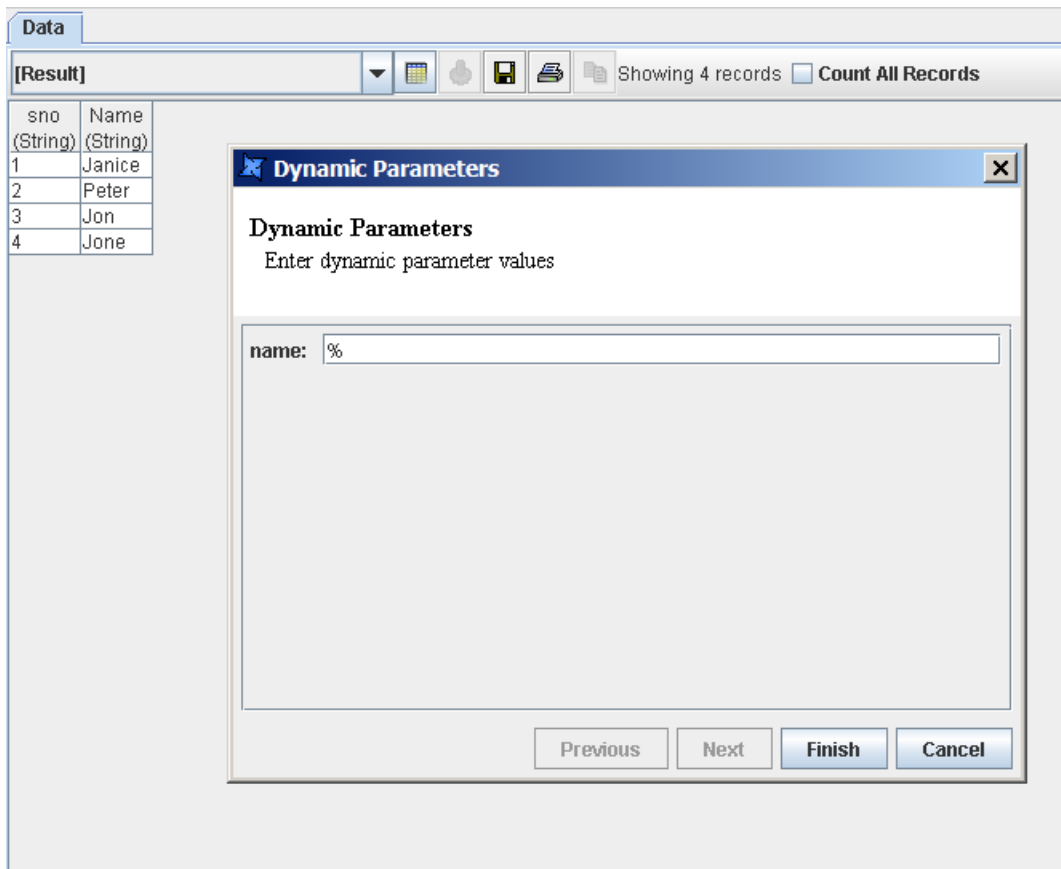
5) At callable tab enter the parameters





The in parameter is only one in our case a string of the name. `#{name##%}` is a special syntax used to indicate this field is a dynamic parameters.

5) Now you can test the data source.



The screenshot shows a data viewer window titled "Data" with a toolbar and a table of results. The table has two columns: "sno (String)" and "Name (String)". The data rows are: 1 Janice, 2 Peter, 3 Jon, and 4 Jone. The toolbar includes icons for navigation and a status bar showing "Showing 4 records" and a "Count All Records" checkbox. Overlaid on the window is a "Dynamic Parameters" dialog box with a title bar, a close button, and the text "Dynamic Parameters" and "Enter dynamic parameter values". The dialog contains a text input field labeled "name:" with a "%" character inside. At the bottom of the dialog are four buttons: "Previous", "Next", "Finish", and "Cancel".

| sno (String) | Name (String) |
|--------------|---------------|
| 1            | Janice        |
| 2            | Peter         |
| 3            | Jon           |
| 4            | Jone          |