



# Using the MQSeries<sup>®</sup> Assist Wizard

*Version 1*



---

## Contents

Before you begin . . . . .	1	Step 7: Create a table view and save the current table definitions (optional) . . . . .	15
Step 1. Open the MQSeries wizard . . . . .	3	Step 8: Verify and create the user-defined function table (optional) . . . . .	17
Step 2. Select the user-defined function type . . . . .	5	Summary. . . . .	21
Step 3: Name the user-defined function . . . . .	7	Additional information . . . . .	23
Step 4: Locate the MQSeries source . . . . .	9		
Step 5: Specify the message format . . . . .	11		
Step 6: Define the MQ message columns . . . . .	13		



---

## Before you begin

Welcome to the MQSeries Assist wizard tutorial.

### Objective:

You can use the DB2<sup>®</sup> MQSeries Assist wizard to create a user-defined function (UDF) that prepares the contents of a queued message to be inserted into a DB2 table via a simple SQL query. This tutorial walks you through the process of using the MQSeries Assist wizard to create and run a custom-built user-defined function that uses a sample message and database.

### Duration:

This tutorial should take 30 to 60 minutes for you to complete.

### Prerequisites:

You need the following software installed to complete this tutorial:

- IBM<sup>®</sup> DB2 Stored Procedure Builder Version 7.2
- MQSeries Server
- MQSeries Application Messaging Interface
- DB2 MQSeries Integration Functions
- DB2 SAMPLE database

For detailed installation and enabling instructions for the MQSeries Server, MQSeries Application Messaging Interface, and MQSeries Integration Functions, see the *Quick Beginnings* book specific to your operating system. You can find the *Quick Beginnings* book at this Web site:  
<http://www.ibm.com/software/ts/mqseries/library/manuals>.

You must have a message in the MQSeries queue.

- If a message exists, you will need to enter the appropriate information in some steps in the tutorial.
- If you do not have a message in the MQSeries queue, you can add one by following these steps:
  1. Open DB2 Command Line Processor.
  2. At the prompt, type `connect to sample` and press Enter.
  3. At the prompt, type  

```
values DB2MQ.MQSEND('Peter Coronado*340 Acker St, San Jose, CA95342*55*5*345.25*03/05/2001')
```

on one line and press Enter.

### Overview

The GetSubscriberData user-defined function that you create in this tutorial accesses a sample message and the SAMPLE database. The user-defined function retrieves the message from the queue, parses the message by the specified delimiter character, and returns the data as a relational table row with each column converted to the specified data type.

After the message is parsed into individual chunks of data, the wizard will perform data type conversions to ensure that the data types of the raw data match the table column data types you specify. You then select the

option to have the wizard create a SQL view, which will simplify the SQL query that you use later to access the data.

This tutorial guides you through all the steps necessary to create and run the GetSubscriberData user-defined function. Follow the eight steps in order because later steps depend on tasks completed in earlier steps. The first six steps are required and the last two steps are optional. The optional steps show you how to create a table view, save the current column definitions to a file, and verify the settings you select before you create the user-defined function.

### **Conventions used in this tutorial**

This tutorial uses typographical conventions in the text to help you distinguish between the names of controls and text that you type. For example:

- Buttons and selection choices on the application interface are in boldface font:

Click **Button Name**.

- Text that you type is in example font:

This is the text that  
you type.

- File or directory names are also in example font:

.../ProgramFiles/SQLLIB/spb/projects/file.mdb.

---

## Step 1. Open the MQSeries wizard

To open the wizard from the IBM DB2 Stored Procedure Builder:

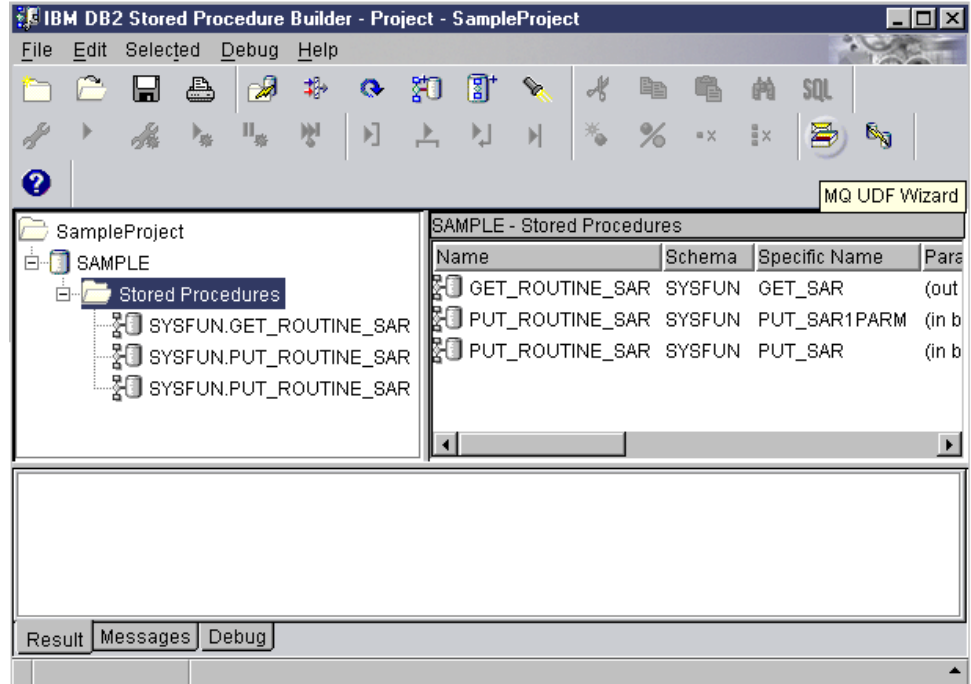
1. Open Stored Procedure Builder. In the first window, click the **New Project** tab, and make sure that the **Database Alias** is **SAMPLE**. Click **OK**.

The screenshot shows the 'IBM DB2 Stored Procedure Builder' dialog box with the 'New Project' tab selected. The dialog contains the following fields and options:

- Project\_name:** SampleProject
- Project path:** C:\SQLLIB\spb\projects
- Driver:** IBM DB2 alias
- Database section:**
  - Alias:** SAMPLE
  - Comments:** (empty)
  - Location (URL):** jdbc:db2:SAMPLE
  - Driver class:** COM.ibm.db2.jdbc.app.DB2Driver
- Use your current user ID and password
- User ID:** sloomis
- Password:** (empty)
- Filter...** button

At the bottom of the dialog are three buttons: **OK**, **Cancel**, and **Help**.

2. Click the Stored Procedures folder in the left pane.



3. Click the **MQ UDF Wizard** (the envelope and the box) toolbar button. The MQSeries Assist wizard opens.



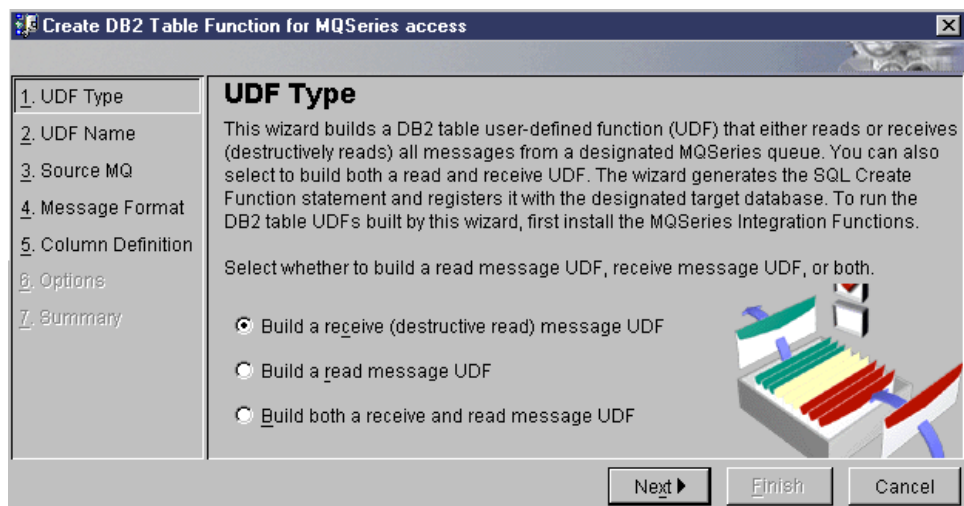
---

## Step 2. Select the user-defined function type

The user-defined function that you will create can handle messages from the queue in two ways. It can read the message from the queue and leave it on the queue, or it can receive (destructively read) a message from the queue and remove it from the queue. The wizard offers a third option to build two user-defined functions – one which reads messages and one which receives messages. For this example, you no longer need the message after you take it off of the queue and enter it into the database.

To build a user-defined function that receives messages from the queue:

1. Click **Build a receive (destructive read) message UDF**.



2. Click **Next**.

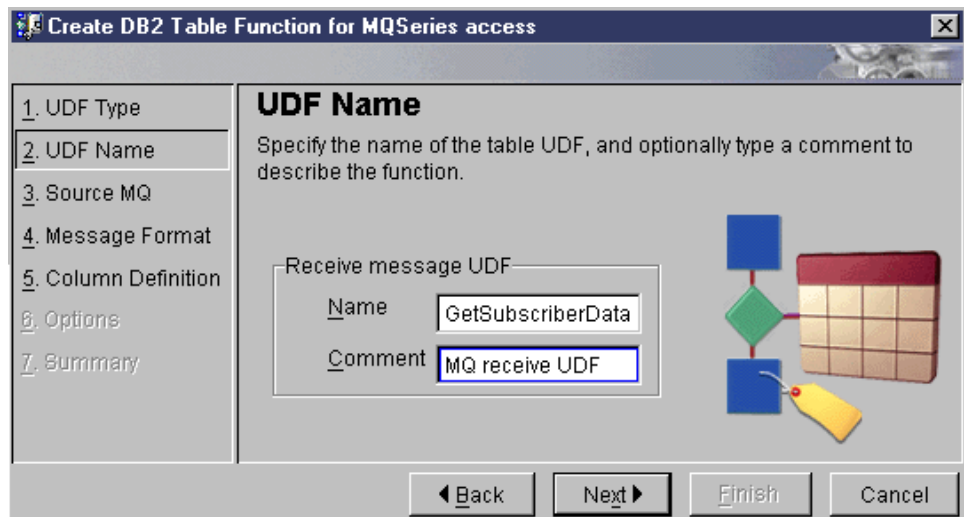


---

## Step 3: Name the user-defined function

To name the user-defined function:

1. In the **Name** field, type GetSubscriberData. (If you are using your own message, you can change the name of the user-defined function.)
2. In the **Comment** field, you can type MQ receive UDF.



The screenshot shows a dialog box titled "Create DB2 Table Function for MQSeries access". On the left is a vertical list of steps: 1. UDF Type, 2. UDF Name (highlighted), 3. Source MQ, 4. Message Format, 5. Column Definition, 6. Options, and 7. Summary. The main area is titled "UDF Name" and contains the instruction: "Specify the name of the table UDF, and optionally type a comment to describe the function." Below this is a "Receive message UDF" section with two input fields: "Name" containing "GetSubscriberData" and "Comment" containing "MQ receive UDF". To the right of the input fields is a small diagram with a blue square, a green diamond, a blue square, and a yellow tag. At the bottom are four buttons: "Back", "Next", "Finish", and "Cancel".

3. Click Next.



---

## Step 4: Locate the MQSeries source

You must identify the queue where the message is located. The service point name identifies the location of the queue. The default location is automatically configured when you install and enable the DB2 MQSeries Integration Functions.

To locate the location of the queue (source):

1. Click **Use default specification**.

The screenshot shows a wizard window titled "Create DB2 Table Function for MQSeries access". The left sidebar contains a list of steps: 1. UDF Type, 2. UDF Name, 3. Source MQ (highlighted), 4. Message Format, 5. Column Definition, 6. Options, and 7. Summary. The main area is titled "Source MQ" and contains the following text: "Service points define locations from which the MQSeries messages are read. Policies define the quality of service options that should be used for a given messaging operation. Service points and policies must be defined in an MQSeries AMI repository." Below this, it says: "Specify the MQSeries location, either by using the default specifications or by explicitly specifying the service point name and a policy name (policy name is optional)." There are two radio button options: "Use default specification" (which is selected) and "Specify service point and policy". Below the second option are two text input fields labeled "Service point name" and "Policy name". To the right of the text is an illustration of a file folder with colored tabs and a computer monitor. At the bottom of the window are four buttons: "Back", "Next" (highlighted with a dashed border), "Finish", and "Cancel".

2. Click **Next**.



---

## Step 5: Specify the message format

For the user-defined function to parse the message correctly, you must specify the format that the message will be received in. The message can be formatted in three different ways:

- You can specify if the message is delimited by a specific character, and what that character is.
- You can specify if the message is in fixed length format, meaning each piece of data would be separated into columns with exact spacing.
- You can tell the wizard to read the message's column format and definitions from a file (i.e.: an XML file).

(If you are using your own message, you might need to specify different choices for steps three through five.)

To determine and specify the format of your message:

1. Click **Show Sample Content**. If you are using the sample message from the beginning of the tutorial, you can see that the message uses a \* delimiter character.
2. Click **OK**.
3. Click **Specify column data format**.
4. Click **Delimited**.
5. In the **Delimiter character** field, type \*.

1. UDF Type  
2. UDF Name  
3. Source MQ  
4. Message Format  
5. Column Definition  
6. Options  
7. Summary

### Message Format

Specify how the column data is identified within the source message. If you have previously saved the column data format and definition in a file, enter the file name.

Column data format

Specify column data format

Delimited

Delimiter character

Fixed-length

Read column data format and definitions from a file

Filename  ...

Show Sample Content

Back Next Finish Cancel

6. Click **Next**.





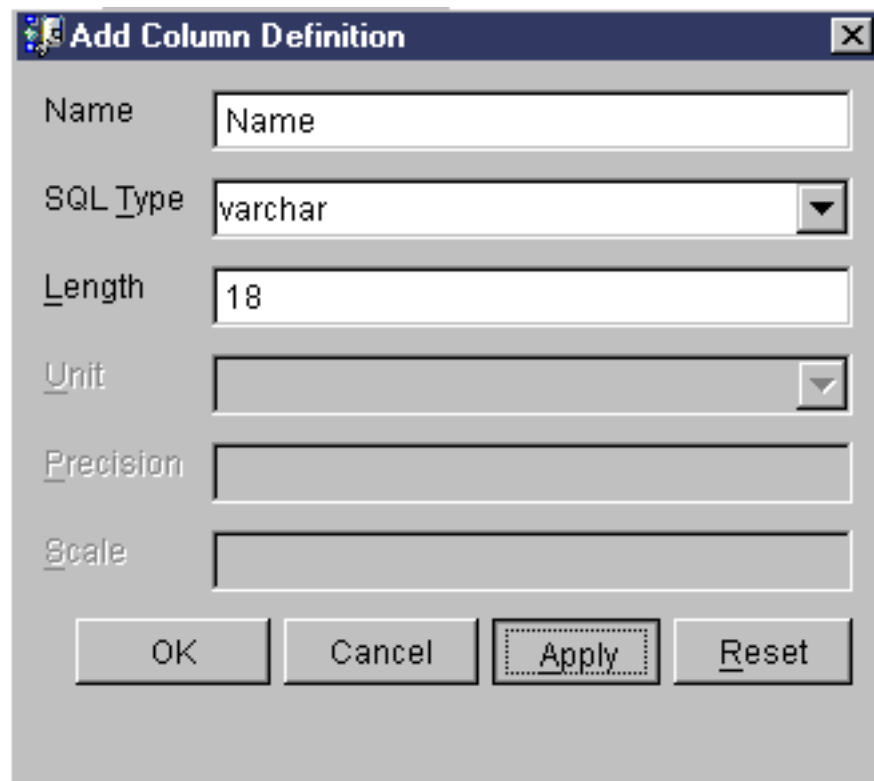
---

## Step 6: Define the MQ message columns

The contents of each delimited segment of the message must have a defined data type so that it can be correctly inserted into a DB2 table. (If you are using your own message, you need to add column definitions that correspond to the data in your message.)

To specify the data type for each piece of the message:

1. Click **Add**. The Add Column Data window opens.
2. Type Name in the **Name** field.
3. Select varchar in the **SQL Type** field.
4. Type 18 in the **Length** field.
5. Click **Apply** to add the column definition and reset the form so that you can add more definitions. The message, Column definition added successfully located at the bottom of the window indicates the status of the column definition that you just added.

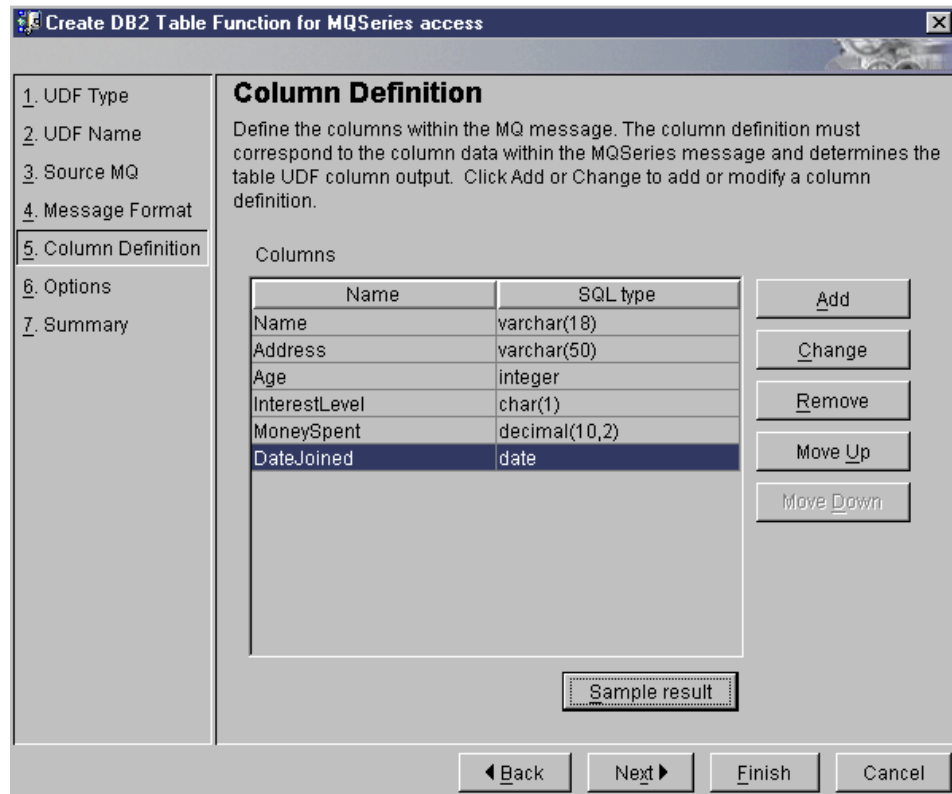


6. Add the remaining column definitions using the following data types:

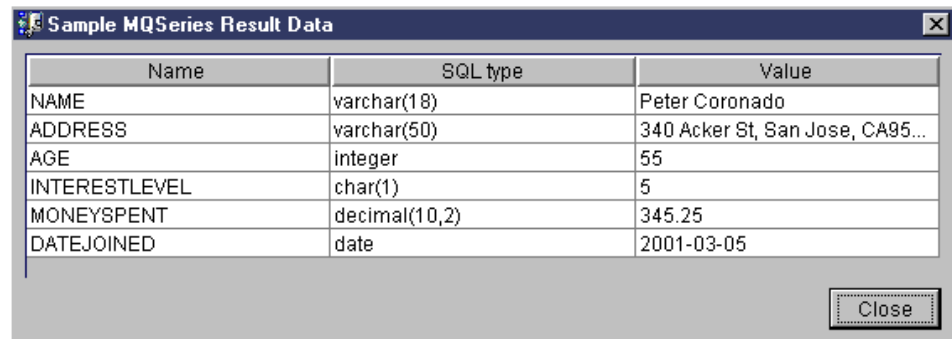
Name	SQL Type	Length	Precision	Scale
Address	varchar	50		
Age	integer			
InterestLevel	char	1		
MoneySpent	decimal		10	2

Name	SQL Type	Length	Precision	Scale
DateJoined	date			

- Click **OK**.



- Click **Sample result** to see how the message data will be parsed according to the column definitions that you specified. The result should look like this:



- Click **Close**.
- Click **Next**.

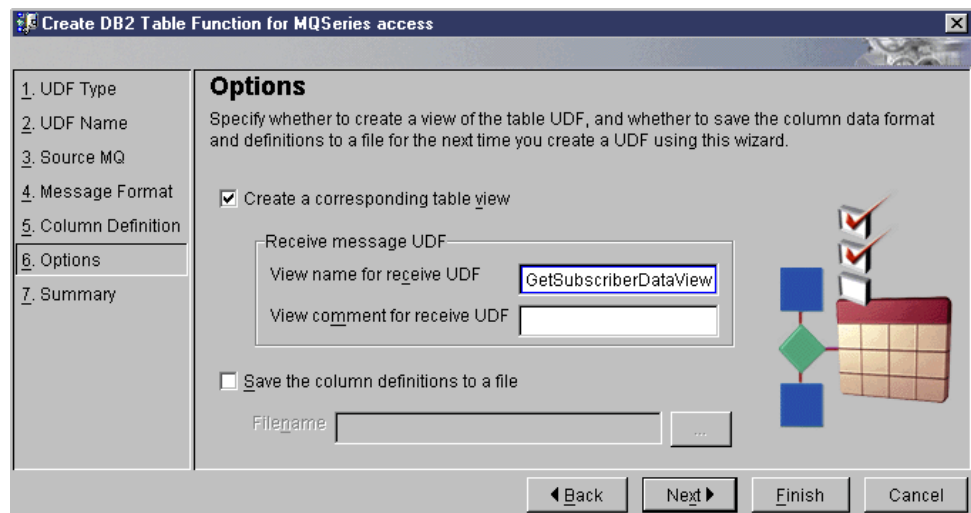
---

## Step 7: Create a table view and save the current table definitions (optional)

Using a table view simplifies access to the user-defined function by shortening the SQL query that you will use later. By saving the current table definitions, you can reuse the definitions the next time you use the MQSeries Assist wizard.

To create a table view:

1. Select the **Create a corresponding table view** checkbox.
2. Type GetSubscriberDataView in the **View name for the receive UDF** field. (If you are using your own message, you might want to change the name of the view.)
3. You can type an appropriate comment in the **View comment for receive UDF** field.



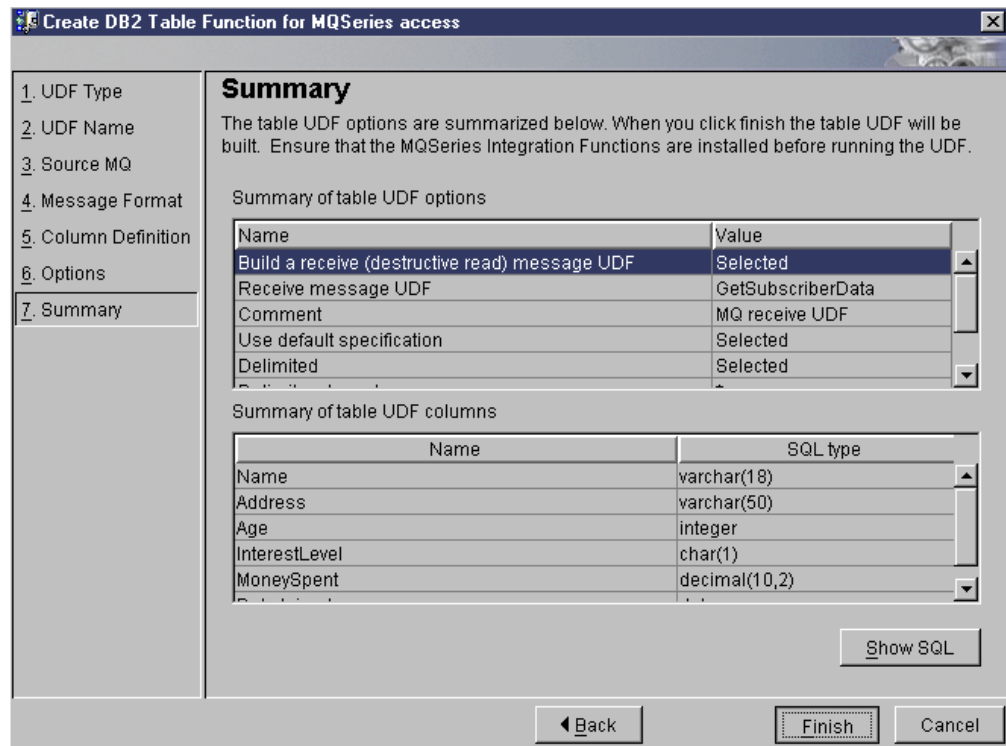
4. Click Next.



---

## Step 8: Verify and create the user-defined function table (optional)

In this step, you can view the settings that you selected for the user-defined function. You also have the option of viewing the code that is created by this wizard. The code builds the user-defined function and optional view that you chose to create.



To view the SQL code that the wizard creates:

1. Click **ShowSQL**. The code includes functionality to retrieve and parse the message, as well as to create the database table. (If you are using your own message, the code shown here will be different.)

```

CREATE FUNCTION GETSUBSCRIBERDATA()
  RETURNS TABLE ( NAME varchar(18),
                  ADDRESS varchar(50),
                  AGE integer,
                  INTERESTLEVEL char(1),
                  MONEYSPENT decimal(10,2),
                  DATEJOINED date )
LANGUAGE SQL
NOT DETERMINISTIC
EXTERNAL ACTION
READS SQL DATA
RETURN SELECT
  VARCHAR(DB2MQ.GETCOL(T.MSG,*,1),18),
  VARCHAR(DB2MQ.GETCOL(T.MSG,*,2),50),
  INT(DB2MQ.GETCOL(T.MSG,*,3)),
  CHAR(DB2MQ.GETCOL(T.MSG,*,4),1),
  DEC(DB2MQ.GETCOL(T.MSG,*,5),10,2),
  DATE(DB2MQ.GETCOL(T.MSG,*,6)) FROM TABLE
(DB2MQ.MQRECEIVEALL()) AS T;

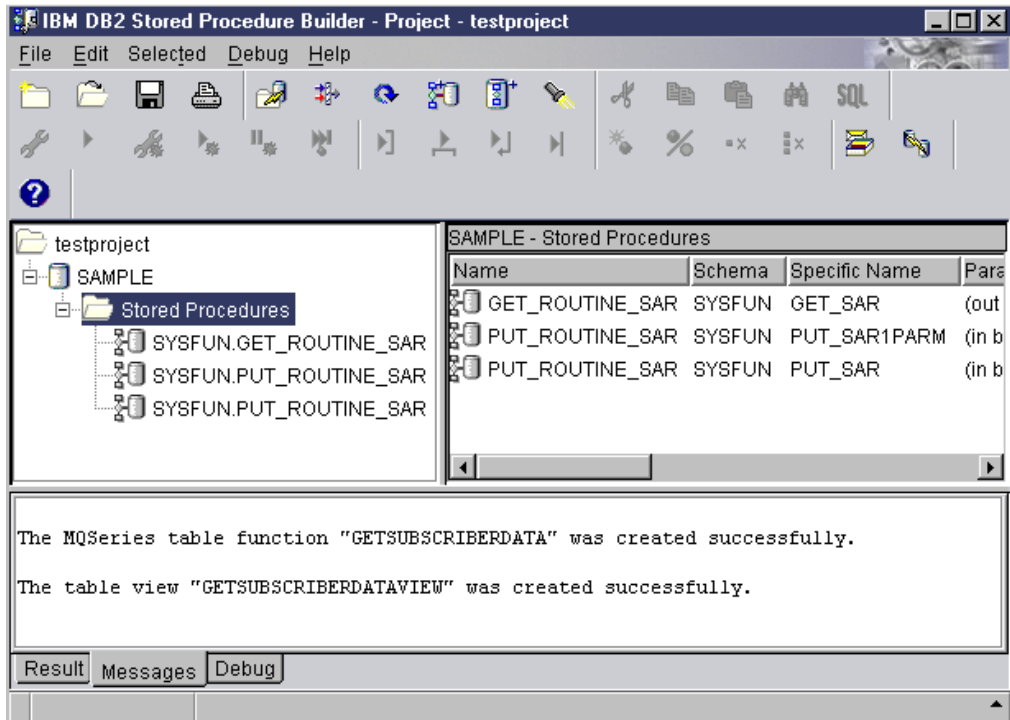
```

Close

2. Click **OK** to close the ShowSQL window.

To create the user-defined function click **Finish**.

In the bottom pane of the Stored Procedure Builder, click the **Message** tab. You should see two messages that confirm the successful creation of the table function and the table view.



To run the user-defined function and verify that it works successfully:

1. Open IBM DB2 Command Line Processor.
2. To connect to the SAMPLE database, type `connect to sample` at the prompt, and press Enter.
3. To view the data that the DB2 view selects, type `select * from GetSubscriberDataView` at the prompt and press Enter. This command displays the one record the view selects from the data in the queued message.
4. To test the user-defined function, you must add a new message to the queue. To do this, type  
`values DB2MQ.MQSEND('Peter Coronado*340 Acker St, San Jose, CA95342*55*5*345.25*03/05/2001')`  
and press Enter.
5. Type `select * from table(GetSubscriberData()) as T` and press Enter. This displays the one record the user-defined function creates from the data in the queued message.





---

## Summary

By going through this tutorial you completed the following tasks:

- You created a user-defined function that retrieves a message from a queue, parses the message, and returns the message as a row of a relational table.
- You defined the final column data types for each segment of the message.
- You created a table view to simplify future access to the user-defined function.



---

## Additional information

The following resources can provide more information on this topic:

- *DB2 Application Development Guide* . This online book has detailed information about using IBM DB2 Stored Procedure Builder and user-defined functions to build applications.

<ftp://ftp.software.ibm.com/ps/products/db2/info/vr6/htm/db2a0/index.htm>

- DB2 Universal Database<sup>®</sup> - What's New Version 7.2 . This document provides information about the MQSeries Assist wizard and related technologies in chapter two.

<http://www.ibm.com/software/data/db2/udb/pdfs/db2v72whatsnew.pdf>

- IBM MQSeries Family Home page. This Web site has links to information about individual MQSeries family products as well as support information.

<http://www.ibm.com/software/ts/mqseries/>