

# JDBC Form Binder

- [Introduction](#)
- [JDBC Binder Properties](#)
  - [Configure JDBC Load Binder](#)
  - [Configure JDBC Store Binder](#)
- [Understanding JDBC Errors](#)
- [Related JDBC Binders](#)
- [Download Demo Apps](#)



## Prevent SQL injection

When using [Hash Variable](#) that uses URL parameter or user-inputted value in the SQL query, ensure that these hash variable(s) are **escaped** in the query!

Make use of hash variable escape keywords, see [Hash Variable - Escaping the Resultant Hash Variable](#).

### Example of VULNERABLE query:

```
SELECT * FROM app_fd_sample_table WHERE c_value = '#requestParam.id#'
```

To fix this, use **?sql** hash variable escape:

```
SELECT * FROM app_fd_sample_table WHERE c_value = '#requestParam.id?sql#'
```

## Introduction

**JDBC Form Binder** allows you to use custom SQL statements to retrieve and load records into your form fields. Similarly, you can write SQL statements to save the records in your form field into a database table.

JDBC form binders settings are located in the form PROPERTIES tab under "Advanced > Data Binder > Load Binder & Store Binder". JDBC form binders replaces the standard Workflow Form Binder.

In a **load** binder, you only need to configure the datasource and SELECT query. In a **store** binder, you will need to write SQL statements for SELECT, INSERT, UPDATE, and DELETE database actions.

JDBC form binders have the following exception where this binder cannot be used (revert back to Workflow Form Binder):

- Use of form element workflow variables. JDBC form binders will not populate or update the workflow variables.
- Use of file and image attachment field elements. JDBC form binders will not handle the file retrieval or file transfer into Joget file storage.

JDBC Binders comes standard in Joget v6 and DX. If you are using Joget v5, you can download the JDBC Binders from [Joget Marketplace](#).

## JDBC Binder Properties

### Configure JDBC Load Binder

Name	Description	Screens (Click to view)
Datasource	<ul style="list-style-type: none"> <li>• Custom Datasource - setup to connect to an external database, has additional configuration.</li> <li>• Default Datasource- connect to the Joget database.</li> </ul> <p>By selecting <b>Default Datasource</b>, the database your Joget is currently using will be selected. (See Figure 2).</p>	
Custom JDBC Driver	<p>Custom JDBC Driver. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above. Example: com.mysql.jdbc.Driver</p>	
Custom JDBC URL	<p>Custom JDBC URL. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above. Example: jdbc:mysql://localhost/jwdb?characterEncoding=UTF8&amp;useSSL=false</p>	

Figure 1: Advance - Load & Save Binder

Custom JDBC Username	Custom JDBC Username. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above.
Custom JDBC Password	<p>Custom JDBC Password. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p><b>i</b> <b>Test the connection parameters</b></p> <p>Click on the "Test Connection" button at the bottom of the page to quickly test out your configurations.</p> </div>
SQL SELECT Query for Load Binder	<p>The SQL query specific to your database type (MySQL, MSSQL, Oracle, etc.). Use question mark ? in your query to represent the primary key or foreign key.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p><b>Example</b></p> <pre>SELECT * FROM app_fd_table WHERE id = ?</pre> </div> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0; background-color: #fff9c4;"> <p><b>!</b> If a column name contains reserved keywords, do ensure it is encapsulated properly.</p> <p>For example for MySQL, if the column identifier itself contains a dot symbol (.), it should be encapsulated like this:</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <pre>SELECT `myAppName.myColumn` FROM app_fd_myTable;</pre> </div> </div> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p><b>i</b> <b>Table &amp; Column Naming</b></p> <ul style="list-style-type: none"> <li>For database tables created by Joget Forms, Joget adds a "c_" in front of table column names (or "t_" if your column name starts with a number) and "app_fd_" in front of database table names.</li> <li>If you use environment hash variables to store SQL query strings, use "?" noescape" to escape SQL query strings in JDBC binders to prevent the "&lt;&gt;" "not equal" operator from being converted, i.e. disables XSS prevention checking. <a href="#">Read here for more information.</a></li> </ul> </div>
Handling for field workflow variable?	Click to select if you are using workflow variables mapping in your form elements. Not applicable for grid elements.
Handling for uploaded files? Form to store uploaded files	<p>Click to select if you are using file or image attachments in your form elements.</p> <p>Select form to store uploaded files only if you are using form grid or spreadsheet.</p>

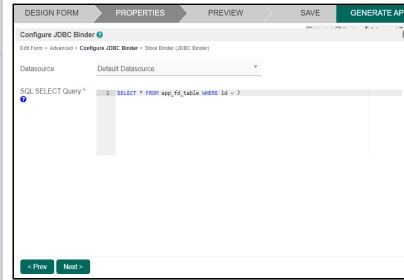


Figure 2: Configure Load JDBC Binder

**Note:** For Joget versions v5 and v6, kindly note that the JDBC load/save binders cannot be used if your form or form grid has file or image attachment field elements or if you are using workflow variables mappings in your form elements. Default to Multirow Form Binder.



If a column name contains reserved keywords, do ensure it is encapsulated properly.

For example for MySQL, if the column identifier itself contains a dot symbol ( . ), it should be encapsulated like this:

```
SELECT `myAppName.myColumn` FROM app_fd_myTable;
```

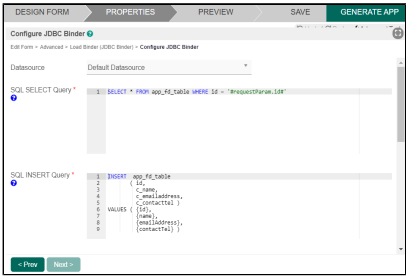
Name	Description	Screens (Click to view)
Store Binder	JDBC Binder selected (See Figure 1).	
Datasource	<ul style="list-style-type: none"> <li>• Custom Datasource - setup to connect to an external database, has additional configuration.</li> <li>• Default Datasource - connect to the Joget database.</li> </ul> <p>By selecting <b>Default Datasource</b>, the database your Joget is currently using will be selected. (See Figure 3).</p>	
Custom JDBC Driver	Custom JDBC Driver. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above. Example: com.mysql.jdbc.Driver	
Custom JDBC URL	Custom JDBC URL. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above. Example: jdbc:mysql://localhost:3307/jwdb?characterEncoding=UTF8&useSSL=false	
Custom JDBC Username	Custom JDBC Username. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above.	
Custom JDBC Password	<p>Custom JDBC Password. This field is required when <b>Custom Datasource</b> is selected in <b>Datasource</b> above.</p> <div data-bbox="228 1037 850 1178" style="border: 1px solid #ccc; padding: 10px;"> <p><b>i</b> <b>Test the connection parameters</b></p> <p>Click on the "Test Connection" button at the bottom of the page to quickly test out your configurations.</p> </div>	
SQL SELECT Query for store binder	<p>The SQL query specific to your database type (MySQL, MSSQL, Oracle, etc.). Use syntax like {field_id} in query to inject submitted form data.</p> <div data-bbox="228 1287 850 1556" style="border: 1px solid #ccc; padding: 10px;"> <p><b>Example</b></p> <pre>SELECT   * FROM   app_fd_table WHERE   id = {id}</pre> </div> <p><b>Note:</b> If you use JDBC in a form grid or spreadsheet for load or save binder, be advised the SQL syntax is different from form load or save binder. Kindly <a href="#">download</a> and view the sample app for JDBC in a form grid or spreadsheet.</p>	

Figure 3: Configure Store JDBC Binder

SQL  
INSERT  
Query

Use syntax like curly brackets **{field\_id}** in query to inject submitted form data.

**Example**

```
INSERT app_fd_table
  (id,
   c_name,
   c_emailaddress)
VALUES ({id},
       {name},
       {emailAddress})
```

You can use **{uuid}** to tell Joget to create a random unique id or a "Universally unique identifier".  
Example:

**Example**

```
INSERT app_fd_table
  (id,
   c_name,
   c_emailAddress)
VALUES ({uuid},
       {name},
       {emailAddress})
```

For JDBC SQL INSERT in a grid, use **{foreignKey}** syntax to populate the parent id into the child table foreign key value, as follows:

```
INSERT app_fd_table_child (
  id,
  c_name,
  c_fk)
VALUES (
  {uuid},
  {name},
  {foreignKey})
```

SQL  
UPDATE  
Query

Use syntax like curly brackets **{field\_id}** in query to inject submitted form data.

**Example**

```
UPDATE app_fd_table
SET   id = {id},
      c_name = {name},
      c_emailaddress = {emailAddress},
      c_contacttel = {contactTel}
WHERE id = {id}
```

SQL DELETE Query	SQL statement to delete form data records in Datalist/Grid element. Use syntax like curly brackets <b>{id}</b> in query to inject form data primary key value.  <b>Example</b>  <pre>DELETE FROM   app_fd_table WHERE   id = {id}</pre>	
Handling for field workflow variable?	Click the checkbox if you are using workflow variables mapping in your form elements. The JDBC binder will copy the field value to the workflow variable when you submit the form if the form is mapped to a process. Not applicable to grid elements.	<b>Note:</b> For Joget versions v5 and v6, kindly note that the JDBC load/save binders cannot be used if your form or form grid has file or image attachment field elements or if you are using workflow variables mappings in your form elements. Default to Multirow Form Binder.
Handling for uploaded files? Form to store uploaded files	Click the checkbox if you are using file or image attachments in your form elements. The JDBC binder will save your file or image attachment into the ".\wflow" folder when you save the form. Select form to store uploaded files only if you are using a form grid or spreadsheet.	

## Understanding JDBC Errors

Error Messages	Causes And Solutions
Message on screen: <b>Record cannot be found</b>	There is a JDBC configuration error. Check the <b>Joget.log</b> file for more information.
The rest of the error messages below are found in the <code>\apache-tomcat-xx.xx\logs\joget.log</code> file.	
ERROR org.joget.plugin.enterprise.JdbcLoadBinder - org.apache.commons.dbcp.SQLNestedException: Cannot load JDBC driver class	The property <b>Custom JDBC Driver</b> is incorrect
ERROR org.joget.plugin.enterprise.JdbcLoadBinder - org.apache.commons.dbcp.SQLNestedException: Cannot create PoolableConnectionFactory (Could not connect to address=(host=localhost)(port=3306)(type=master) : Socket fail to connect to host:localhost, port:3306. Connection refused: connect)	The property <b>Custom JDBC URL</b> is incorrect
ERROR org.joget.plugin.enterprise.JdbcLoadBinder - org.apache.commons.dbcp.SQLNestedException: Cannot create PoolableConnectionFactory (Could not connect to address=(host=localhost)(port=3307)(type=master) : Access denied for user 'xxx'@'localhost' (using password: YES)	The property <b>Custom JDBC Username</b> or <b>Custom JDBC Password</b> is incorrect.
ERROR org.joget.plugin.enterprise.JdbcLoadBinder - java.sql.SQLException: .....	The property <b>SQL SELECT Query</b> is incorrect. Do also check the SQL lower or upper case spelling on table and column names, especially if you are running a database on a Linux OS which is case sensitive.
ERROR org.joget.plugin.enterprise.JdbcLoadBinder - java.sql.SQLException: Could not set parameter at position 1 (values was '?') Query - conn:506(M) - SELECT * FROM app_fd_table WHERE id = "?"	Do not enclosed the ? in single or double quotation marks. Correct syntax is ? :  <pre>SELECT * FROM app_fd_table WHERE id = ?</pre>
ERROR org.joget.tutorial.JdbcOptionsBinder - com.microsoft.sqlserver.jdbc.SQLServerException: The value is not set for the parameter number 2	DB error on <b>JDBC Options</b> with "Field ID to control available options based on Grouping". You can only pass one ? parameter in the SQL.

## Related JDBC Binders

- [JDBC Options Binder](#)
- [JDBC Datalist Action](#)
- [JDBC Datalist Database Binder](#)

## Download Demo Apps

- Demo app using JDBC Form Binders for MySQL database: [APP\\_jdbc\\_load\\_store\\_form\\_binder\\_dx\\_kb.jwa](#)
  - Demo app using JDBC Form Grid Binders for MySQL database: [APP\\_form\\_grid\\_with\\_jdbc\\_binder.jwa](#)
  - Demo app using JDBC in List Grid for MySQL database: [APP\\_list\\_grid\\_multirow\\_jdbc\\_dx\\_kb.jwa](#)
- 
- Demo app using JDBC Form Binders for Oracle database: [APP\\_example\\_form\\_jdbc\\_on\\_oracle\\_v6.jwa](#)
  - Demo app using JDBC Form Grid Binders for Oracle database: [APP\\_jdbc\\_form\\_grid\\_oracle\\_dx\\_kb.jwa](#)