Working with database fields in ConfiForms

If you are new to ConfiForms please take a look at less complicated tutorials first to get yourself familiar with ConfiForms

Creating a simple form in Confluence with ConfiForms

Multiple ways to show your form data in Confluence with ConfiForms

Basic concepts

Since version 1.20 of ConfiForms you are now able to source the data for your fields in ConfiForms directly from your databases. This requires your database to be accessible from your Confluence server and you need to provide the connection configuration, before you can use those fields. More on this below.

We have 4 type of fields which can be backed by a database:

- DB Dropdown
- DB Multi-select
- DB Checkbox
- DB Radio group
- UPDATED DB Smart Classifier (since version 1.24)
- UPDATED DB Dynamic dropdown (auto-suggest) (since version 1.39)

First and last are single choice field types and 2 in the middle support multiple choices.

Fields behave exactly as their "normal" and "smart" field counterparts, but the choices are loaded from the database and the query you configure.

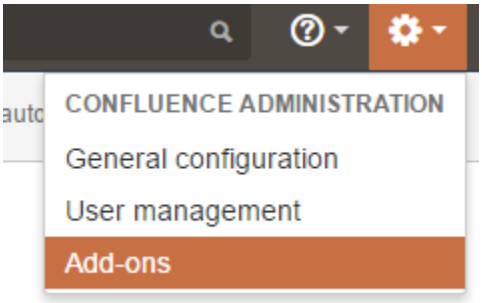
Important!

You query should return a list of data with **AT LEAST** 2 columns, where first column will be used as an ID and second column will be used as a label (other columns can be then used later in filtering, if necessary).

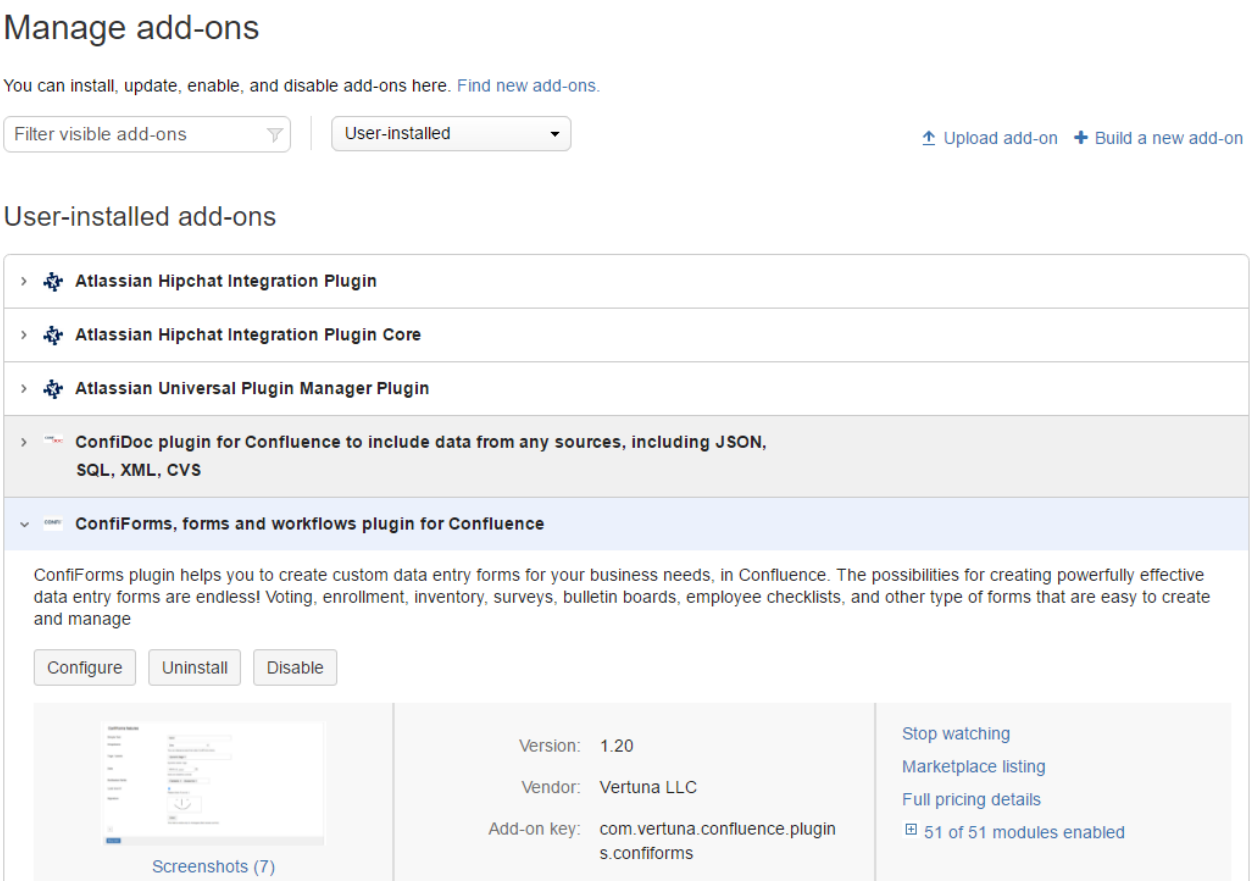
We strongly advice you to have a unique value for column ID

Let's start with Database connection configuration, so ConfiForms will be able to use it. This configuration can be done only by Confluence administrators and usage can be restricted (you can setup which user group / user can use which connection)

You can navigate to Database Connection Configuration for ConfiForms using the "Configure" button in Add-on configuration

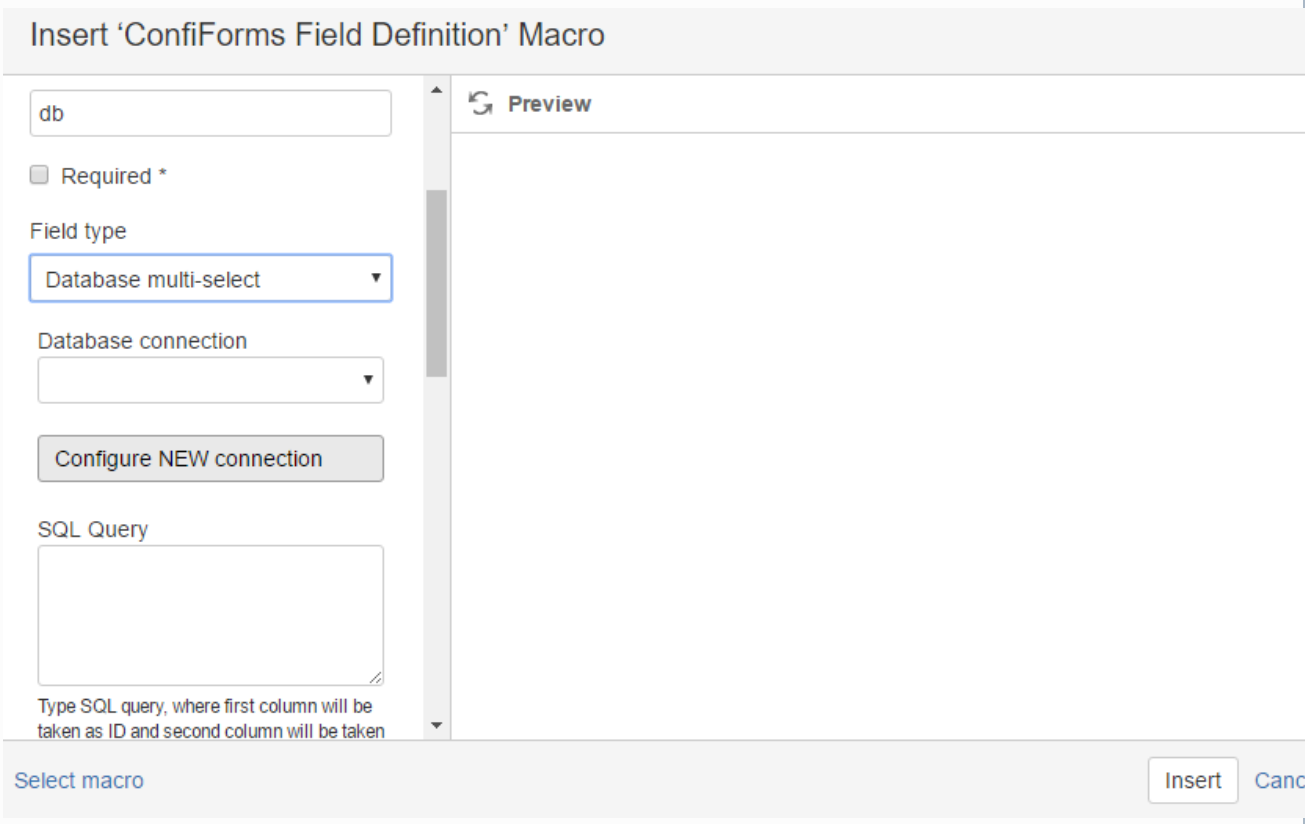


Then navigating to ConfiForms add-on

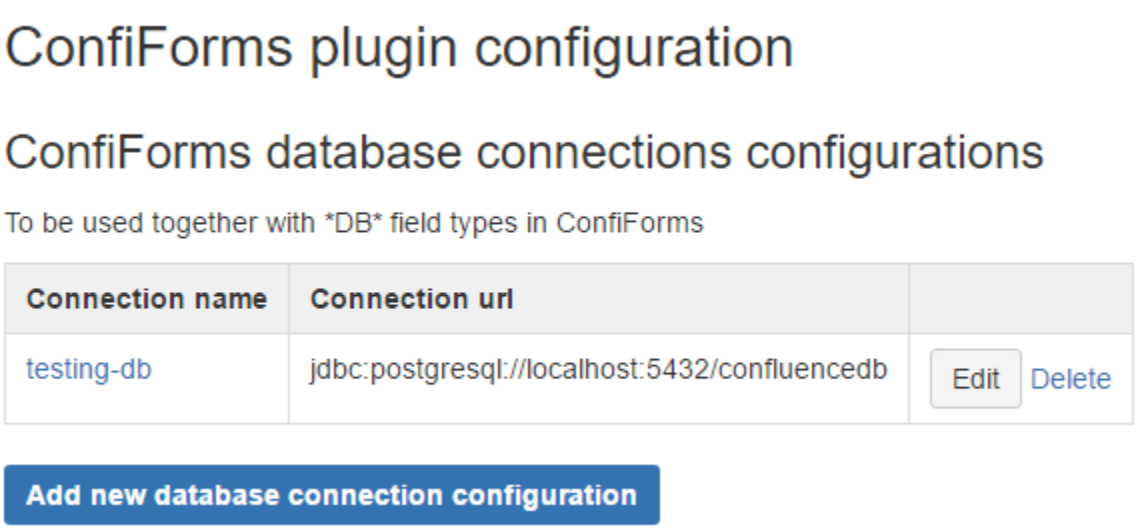


And clicking on "Configure" button. You will be routed to the following URL: https://YOUR_SERVER/admin/vertuna/confiforms/config.action

Alternatively, you can just start configuring your new form, adding new field definition of one of those DB types and clicking the "Configure NEW connection"

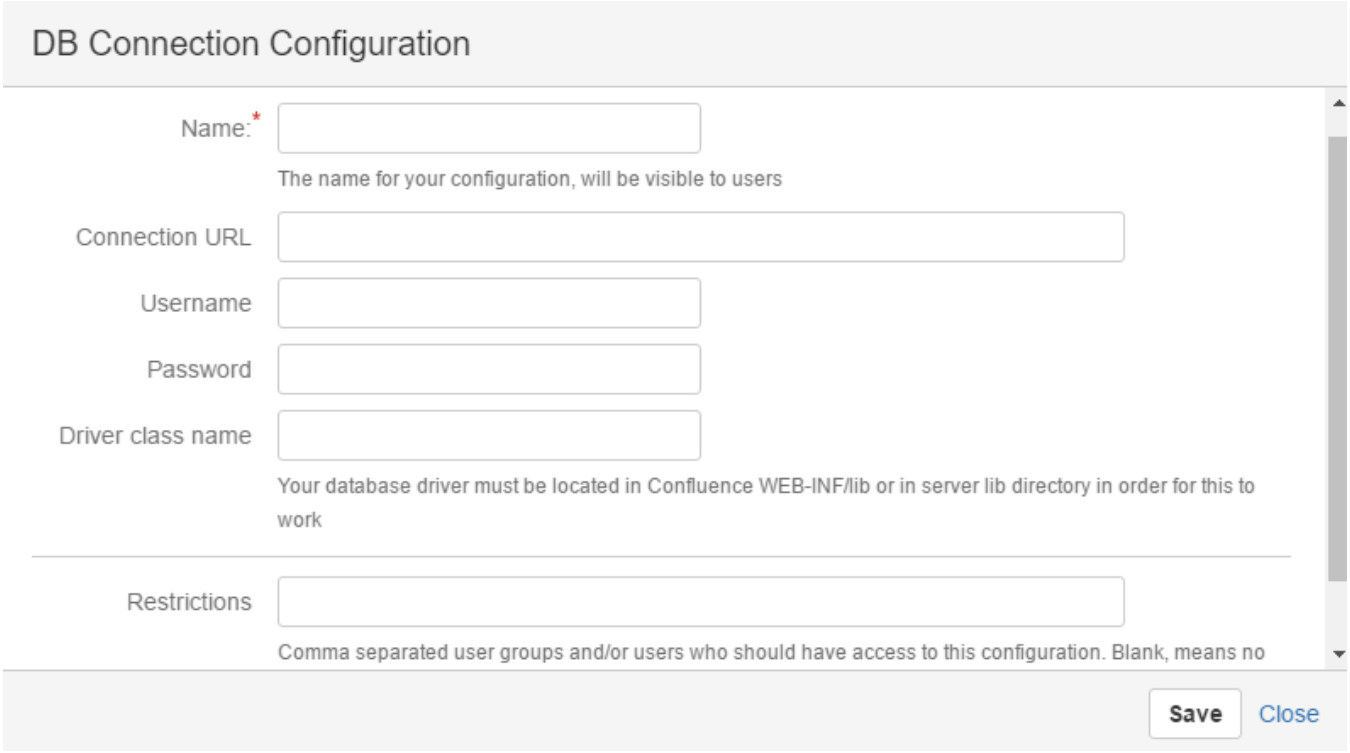


Configuration UI will be looking something like this:



Initially you will not have any connections defined, so the table will be empty.

Click on "Add new database connection configuration" button and you will be presented with a dialog to configure new connection



Give your connection a name, then compile a Connection URL, we use JDBC to connect to the database and you will need to provide correct JDBC connection string, class name for your database driver and you need to make sure the database driver is placed in the **<install-dir>/lib** directory of your Confluence server (restart will be needed). Confluence has PostgreSQL drive bundled together with the server, nothing additional is needed. For MySQL, Oracle, Microsoft SQL databases you will need to place the driver into the correct place.

JDBC Driver Location

Put the driver jar in **<install-dir>/lib** - this is the directory that has other tomcat jars. If you have a **much older version of Confluence** that does not have this directory, look for **<install-dir>/common/lib** instead.

Here is the table of drivers and their connection strings we have compiled for our other plugin, ConfiDoc, but which is completely valid for ConfiForms

List of connection strings for various drivers and databases

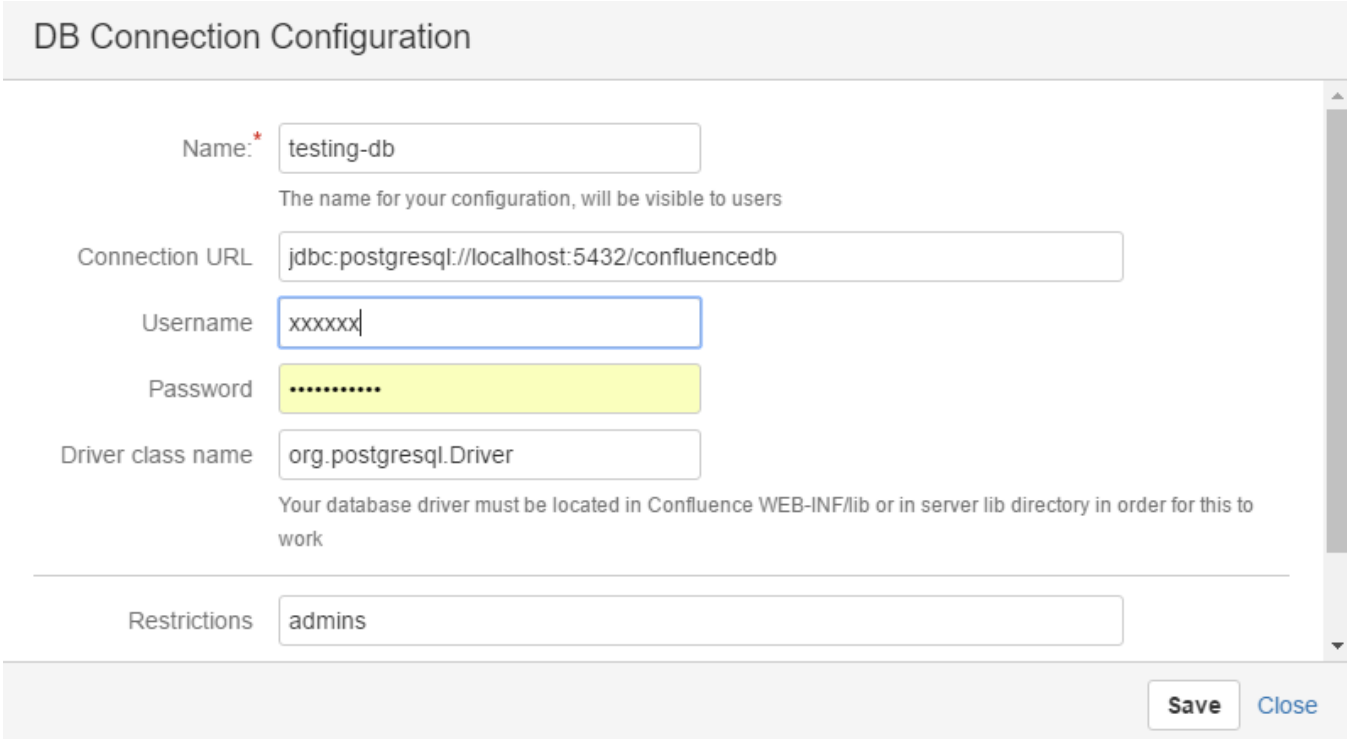
Driver	Connection string

MySql/
com.mysql.jdbc.Driver
	`jdbc:mysql://:/?user=&password=`
	For mySQL driver you will need to enable "generateSimpleParameterMetadata" parameter
	`jdbc:mysql://localhost:3306/mydb?generateSimpleParameterMetadata=true`

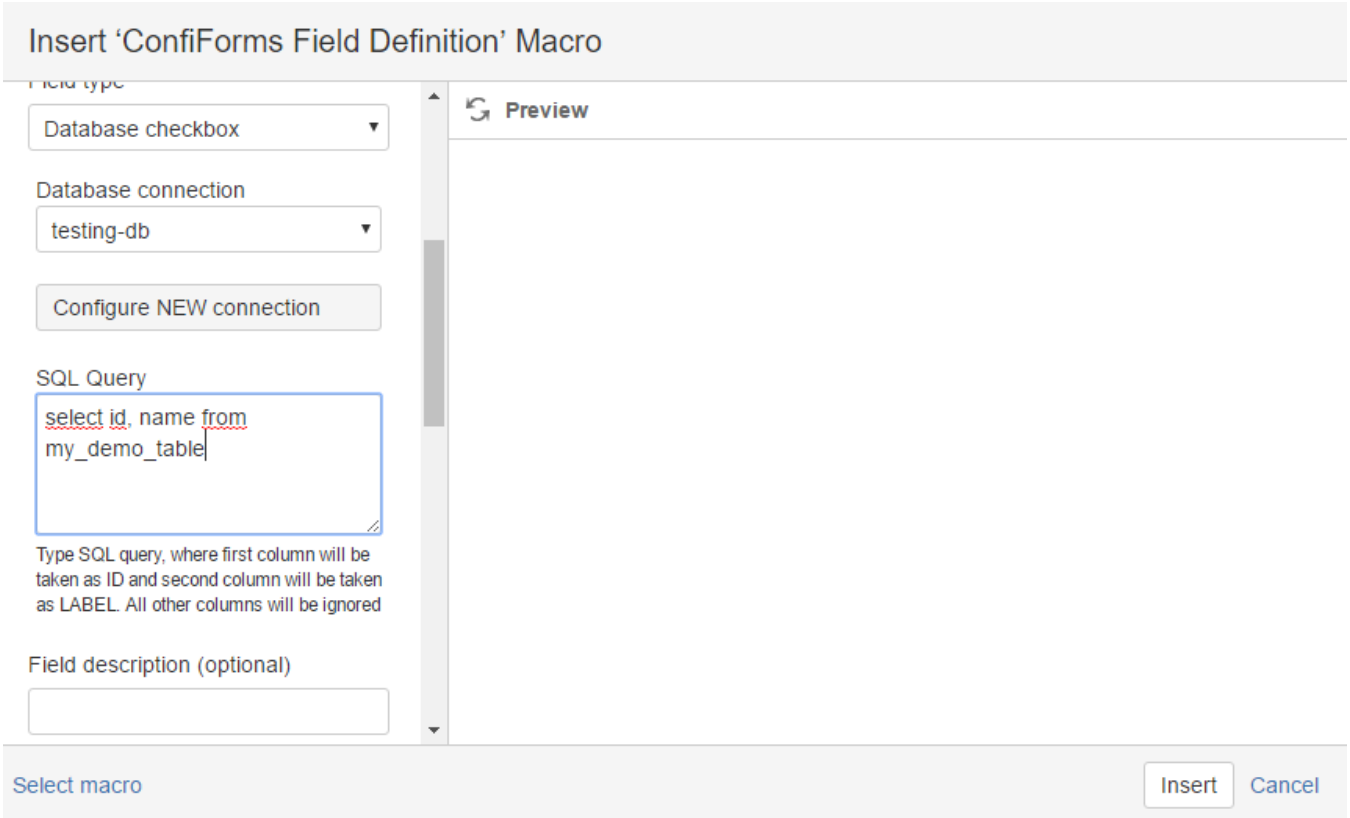
PostgreSQL/ org.postgresql.Driver	`jdbc:postgresql://:/?user=&password=`
IBMDB2/ COM.ibm.db2.jdbc.app. DB2Driver	`jdbc:db2://:/?user=&password=`

MSSQL(MicrosoftDriver)/ com.microsoft.jdbc.sqlserver. SQLServerDriver	`jdbc:microsoft:sqlserver://:;databaseName=?user=&password=`
MSSQL(Weblogic)/ weblogic.jdbc.mssqlserver4. Driver	`jdbc:weblogic:mssqlserver4:@:`
MSSQL(SprintaDriver)/ com.inet.tds.TdsDriver	`jdbc:inetdae::?database=?user=&password=`
MicrosoftSQLServer (JTurboDriver)/ com.ashna.jturbo.driver.Driver	`jdbc:JTurbo://://user=/password=`
Microsoft SQL Server/ com.microsoft.sqlserver.jdbc. SQLServerDriver	`jdbc:sqlserver://yourserver.database.windows.net:1433;database=YourDatabase; user=yourusername@yourserver;password=yourpassword;encrypt=true; trustServerCertificate=false;hostNameInCertificate=*.database.windows.net; loginTimeout=30;` https://docs.microsoft.com/en-us/sql/connect/jdbc/building-the-connection-url?view=sql-server-2017
Microsoft SQL Server (aternative, newer) / com. microsoft.jdbc.sqlserver.SQLS erverDriver	jdbc:sqlserver://YourServer:Port;databaseName=YourDatabase More details on https://docs.microsoft.com/en-us/sql/connect/jdbc/using-the-jdbc-driver https://docs.microsoft.com/en-us/sql/connect/jdbc/building-the-connection-url?view=sql-server-2017
OracleThin/ oracle.jdbc.driver. OracleDriver	`jdbc:oracle:thin:/@::`
Oracle(OranhoDriver)/ com.inet.pool.PoolDriver	`jdbc:inetpool:inetora:?database=&user=&password=&sid=`
Sybase(jConnect5.2)/ ncom.sybase.jdbc2.jdbc. SybDriver	`jdbc:sybase:Tds::?user=&password=`
PointBaseEmbeddedServer/ com.pointbase.jdbc. jdbcUniversalDriver	`jdbc:pointbase://embedded:/?user=&password=`
Cloudscape/ COM.cloudscape.core. JDBCDriver	`jdbc:cloudscape:?user=&password=`
CloudscapeRMI/ RmiJdbc.RJDriver	`jdbc:rmi://:/jdbc:cloudscape:?user=&password=`
Firebird(JCA-JDBCDriver)/ org.firebirdsql.jdbc.FBDriver	`jdbc:firebirdsql://:/?user=&password=`
IDSServer/ ids.sql.IDSDriver	`jdbc:ids://:/conn?dsn=''`
InformixDynamicServer/ com.informix.jdbc.IfxDriver	`jdbc:informix-sqli://:/:INFORMIXSERVER=`
InstantDB/ org.enhydra.instantdb.jdbc. idbDriver	`jdbc:idb:`
Interbase(InterClientDriver)/ interbase.interclient.Driver	`jdbc:interbase:///`
HypersonicSQL/ org.hsql.jdbcDriver	`jdbc:HypersonicSQL:?user=&password=`

You configuration for connecting to PostgreSQL will look something like this:



When it is saved, you will be able to use it in one of the DB fields:



That's it - you field is ready!

Of course you can re-use your database connections, have multiple DB fields on the same form, etc. Fields behave and act as normal ConfiForms fields.

In exports to Excel and CSV their labels will be used, for exports as JSON/XML we will export their internal ID's

Take a look at one demo configuration which shows you how to create **"dependent"** database dropdown fields Working with Dependent DB dropdowns