Install DGLux5 Server

> **Important**
>
> These steps do not install DGLux5 for Distributed Services Architecture (DSA). To install DGLux5 for DSA, go here.

DGLux5 Server is an enterprise-ready, fully-integrated DGLux5 web server packaged with Apache Tomcat 7.0.55. Security and web hosting is handled by Tomcat, which makes it easy to customize for your specific needs and preferred user authentication mechanism (including LDAP).

DGLux5 Server supports passthrough querying of JDBC connected data-sources, which enables DGLux5 to be used to read from and write to any database and to communicate with any web services required for your visualization data feeds.

OS Compatibility

This version has been tested on Windows, OS X and Linux. Install Java JRE 7+ on the target server prior to installing DGLux5.

Browser Compatibility

DGLux5 is a 100% HTML5 application that does not require any special browser plugins. For the best user experience, use the latest version of Google Chrome.

The following web browsers are supported:

- Microsoft Edge 27+
- Google Chrome 48+
- Mozilla Firefox 40+
- Opera 27+
- Apple Safari 8.0.8+

Installing and Starting DGLux5

To install DGLux5, download the zip file from our web site, unzip it in the desired installation directory, and run the startup script:

On Windows: Double-click the (bin/startup.bat) file.

On OSX / Linux:

1. Open Terminal.app
2. cd to the bin subdirectory in the installation directory,
3. Issue the "sh ./startup.sh" command.

To shut DGLux5 down, run the shutdown.bat or shutdown.sh script.

Connecting to DGLux5

By default, Tomcat uses port 8080. To connect, launch a web browser and browse to
http://127.0.0.1:8080 (assuming you are on the server machine). Log in using the following credentials:

- username: dgSuper
- password: dg1234

When prompted, request a license for your installation and install it as instructed.

Updating DGLux5

Registered users are notified when updates to DGLux5 become available. To install an update:

1. Download DGLux5_xxxx.zip.
2. In your browser, navigate to http://127.0.0.1:8080/update.html.
3. Browse to the DGLux5_xxxx.zip file.
4. Click the "Upload" button.

After the upload, the browser refreshes and you are running the updated version of DGLux5.

Customizing DGLux5 Installation

DGLux5 runs on standard J2EE servers, which means you can use web servers other than Tomcat, such
as Jetty and IBM Websphere.

Configuring Users

To administer users in DGLux5, edit "conf/tomcat-users.xml".

For details about using an external directory server or database for user management, see Apache
Tomcat Website.

Configuring JDBC

To modify the JDBC configuration, edit "webapps/ROOT/WEB-INF/dglux5/example_db.json". The file format is as follows:

```
{
  "databases":
  [
    {
      "name" : "jdbcDemoDB",
      "driver" : "org.apache.derby.jdbc.ClientDriver",
      "url" : "jdbc:derby://localhost:1527/jdbcDemoDB",
      "user" : "optional",
      "pass" : "optional",
      "passthrough" : true
      "queries" :
      [
        {
          "name" : "getWishes",
          "query" : "select ENTRY_DATE, {$WHAT} from WISH_LIST order by
ENTRY_DATE",
          "cacheSecs" : 360
        }
      ]
    }
  ]
}
```

To use this feature, rename "example_db.json" to "_db.json". Changes to this file take effect immediately and do not require you to restart the server.

In this file you can define multiple databases that are displayed as query data sources in DGLux5.

> Ensure that an appropriate database driver JAR file is present in "webapps/ROOT/WEB-INF/lib". You can download popular database driver JAR files from these sources:
>
>   - MySQL
>   - SQL Server
>   - Oracle
>
> You must restart Tomcat after adding a new JAR.

If the passthrough setting is enabled, you can execute arbitrary queries against the database from DLGux5. Note that there is a security risk in doing so, and d=administer security accordingly. Typically

this feature is used in a controlled environment or when testing the queries required for your DGLux5 application.

A more secure approach is to disable passthrough and define query templates under queries. Predefined queries are cached, which reduces server workload in highly concurrent settings.

The important parts of a query template are the "name" and the parameters in the query string, which enclosed in "{$ }".

After your query templates are configured, you can invoke them from DGLux5. For example, if your query is defined in "_db.json" as:

```
"getWishes": "select {$WHEN}, {$WHAT} from {$WHERE} order by {$WHEN}"
```

You invoke the query in DGLux5 as follows:

```
<queryName> where <key>=<val> and <another_key>=<another_val>
```

For example:

```
getWishes where WHEN=ENTRY_DATE and WHERE=WISH_LIST AND WHAT=WISH_ITEM
```
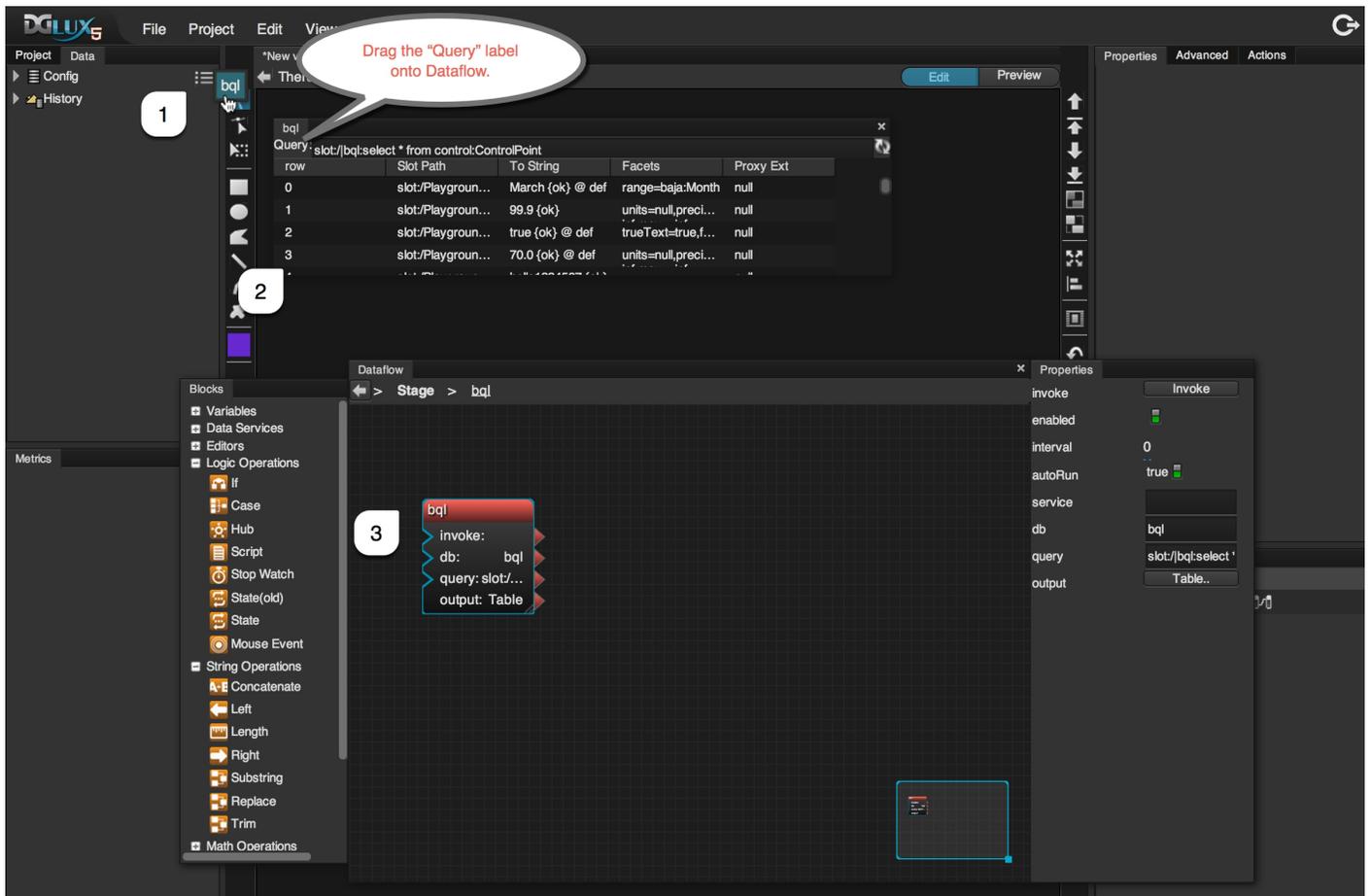
"where", "and" and variable names are case-insensitive. The query name and values are case-sensitive. The "where" and "and" clauses must be delimited by white space.

If a key or value has the quote (") or equal (=) character, escape them using double quotes. For example "this has ""quotes"" in it".

Executing Queries from DGLux5

1. Click on "Data" tab in the upper left panel.
2. To display configured database connections, click the hamburger icon.
3. Select the desired database connection.
4. Type your query and press ENTER.
5. To create a block that returns the query's result set, drag "Query" label onto your dataflow.

Previous: Install DGLux5 for Niagara AX

Next: Update DGLux5

From:
https://wiki.dglogik.com/ - **DGLogik**

Permanent link:
https://wiki.dglogik.com/dglux5_wiki:installation:dgserver

Last update: **2021/09/20 14:43**