

# HiveJDBCInterface

## Hive JDBC Driver

- [Hive JDBC Driver](#)
  - [Integration with Pentaho](#)
  - [Integration with Squirrel SQL Client](#)

The current JDBC interface for Hive only supports running queries and fetching results. Only a small subset of the metadata calls are supported.

To see how the JDBC interface can be used, see [sample code](#).

### Integration with Pentaho

1. Download pentaho report designer from the [pentaho website](#).
2. Overwrite report-designer.sh with the code provided below.

```
#!/bin/sh

HADOOP_CORE={{ls $HADOOP_HOME/hadoop-*-core.jar}}
CLASSPATH=.:$HADOOP_CORE:$HIVE_HOME/conf

for i in ${HIVE_HOME}/lib/*.jar ; do
    CLASSPATH=$CLASSPATH:$i
done

CLASSPATH=$CLASSPATH:launcher.jar

echo java -XX:MaxPermSize=512m -cp $CLASSPATH -jar launcher.jar
java -XX:MaxPermSize=512m -cp $CLASSPATH org.pentaho.commons.launcher.Launcher
```

3. Build and start the hive server with instructions from [HiveServer](#).
4. Compile and run the Hive JDBC client code to load some data (I haven't figured out how to do this in report designer yet). See [sample code](#) for loading the data.
5. Run the report designer (note step 2).

```
$ sh reporter-designer.sh
```

6. Select 'Report Design Wizard'.
7. Select a template - say 'fall template' - next.
8. Create a new data source - JDBC (custom), Generic database.
9. Provide Hive JDBC parameters. Give the connection a name 'hive'.

```
URL: jdbc:hive://localhost:10000/default
Driver name: org.apache.hadoop.hive.jdbc.HiveDriver
Username and password are empty
```

10. Click on 'Test'. The test should succeed.
11. Edit the query: select 'Sample Query', click edit query, click on the connection 'hive'. Create a new query. Write a query on the table testHiveDriverTable, for example, select \* from testHiveDriverTable. Click next.
12. Layout Step: Add PageOfPages to Group Items By. Add key and value as Selected Items. Click next. And Finish.
13. Change the Report header to 'hive-pentaho-report'. Change the type of the header to 'html'.
14. Run the report and generate pdf. You should get something like the report attached here.

### Integration with Squirrel SQL Client

1. Download, install and start the Squirrel SQL Client from the [Squirrel SQL website](#).
2. Select 'Drivers -> New Driver...' to register the Hive JDBC driver.
  - a. Enter the driver name and example URL:

```
Name: Hive
Example URL: jdbc:hive://localhost:10000/default
```

3. Select 'Extra Class Path -> Add' to add the following jars from your local Hive and Hadoop distribution.

```
HIVE_HOME/build/dist/lib/*.jar  
HADOOP_HOME/hadoop-*-core.jar
```

4. Select 'List Drivers'. This will cause Squirrel to parse your jars for JDBC drivers and might take a few seconds. From the 'Class Name' input box select the Hive driver:

```
org.apache.hadoop.hive.jdbc.HiveDriver
```

5. Click 'OK' to complete the driver registration.
6. Select 'Aliases -> Add Alias...' to create a connection alias to your Hive server.
  - a. Give the connection alias a name in the 'Name' input box.
  - b. Select the Hive driver from the 'Driver' drop-down.
  - c. Modify the example URL as needed to point to your Hive server.
  - d. Leave 'User Name' and 'Password' blank and click 'OK' to save the connection alias.
7. To connect to the Hive server, double-click the Hive alias and click 'Connect'.

When the connection is established you will see errors in the log console and might get a warning that the driver is not JDBC 3.0 compatible. These alerts are due to yet-to-be-implemented parts of the JDBC metadata API and can safely be ignored. To test the connection enter *SHOW TABLES* in the console and click the run icon.

Also note that when a query is running, support for the 'Cancel' button is not yet available.