apache > db > derby > papers > DerbyTut

Apache Derby

The Apache DB Project
http://db.apache.org/

Last Published: 05/07/2020 21:39:00

Font size: Reset -a +a

Step 2: ij Basics

Introduction

`ij` is an interactive SQL scripting tool that comes with Derby. It can be used with the Derby Embedded JDBC driver or with a client JDBC driver, such as the Derby Network Client.

This section uses the Derby Embedded JDBC driver to introduce a few `ij` features. The information presented here is minimal, just enough to get you started. For a more complete introduction, see the Getting started with ij section in the Tools and Utilities Guide.

Set the environment

To set up the environment, follow the "Configure Embedded Derby" instructions.

Start up ij

Start up `ij` with this command:

```
java org.apache.derby.tools.ij
```

You should see the output shown below:

```
ij version 10.4
ij>
```

The error below means the class path isn't set correctly:

```
java org.apache.derby.tools.ij
Exception in thread "main" java.lang.NoClassDefFoundError: org/apache/derby/tools/ij
```

For help resolving class path problems, see the "Configure Embedded Derby" and "Verify Derby" instructions.

Create a database

To create a database, specify the `create=true` attribute in the connection URL. For example, the command below creates a new database called `MyDbTest`:

```
ij> connect 'jdbc:derby:MyDbTest;create=true';
```

Now exit `ij`:

```
ij> exit;
```

List the contents of your directory with `ls` (UNIX) or `dir` (Windows). On UNIX you'll see output like this:

```
$ ls
MyDbTest  derby.log
```

The `MyDbTest` directory contains the files that make up the database you just created. The `derby.log` file is an error log, which is helpful whenever things don't work as expected.

Connect to a database

Start up `ij` again and connect to the database you just created:

```
java org.apache.derby.tools.ij
ij> connect 'jdbc:derby:MyDbTest';
```

We'll take a quick look at the protocol ("`jdbc:derby:`") and database ("`MyDbTest`") in this connection URL.

Protocol

Internally, `ij` determines by default which driver to load from the protocol ("`jdbc:derby:`"). In this case, it knows to load the embedded JDBC driver. We could also have specified the protocol with a property as shown below:

```
java -Dij.protocol=jdbc:derby: org.apache.derby.tools.ij
ij> connect 'MyDbTest';
```

For more information about `ij` startup properties see Starting ij using properties in the Tools and Utilities Guide.

Database

Connecting to the `MyDbTest` database in the connection URL above works because the `MyDbTest` database directory is in the current working directory; *i.e.*, the directory where you started up `ij`.

Let's say that your current directory location is `/home/bill/databases` and that you decide to change your directory to a different place entirely. You can connect to the `MyDbTest` database by specifying the complete directory path, like this:

```
java org.apache.derby.tools.ij
ij> connect 'jdbc:derby:/home/bill/databases/MyDbTest';
```

You could also specify the Derby system home for the database like this:

```
java org.apache.derby.tools.ij -Dderby.system.home=/home/bill/databases
ij> connect 'jdbc:derby:MyDbTest';
```

The *Developer's Guide* provides information about where Derby looks for databases:

- Connecting to databases
- Defining the system directory

Execute SQL statements

Once you connect to a database, you can execute SQL statements. `ij` expects each statement to be terminated with a semicolon (;); for example:

```
ij> create table derbyDB(num int, addr varchar(40));
ij> insert into derbyDB values (1956,'Webster St.');
ij> insert into derbyDB values (1910,'Union St.');
ij> update derbyDB set num=180, addr='Grand Ave.' where num=1956;
ij> select * from derbyDb;
```

Disconnect from a database

The disconnect command disconnects from the current database:

```
ij> disconnect;
```

Exit

The exit command quits out of `ij` and, in embedded mode, shuts down the Derby database:

```
ij> exit;
```

Run SQL Scripts

You can execute SQL scripts in `ij` as shown below:

```
ij> run 'my_file.sql';
```

You can also run SQL scripts from the command line:

```
java org.apache.derby.tools.ij my_file.sql
```

The Tools and Utilities Guide provides more information about running SQL scripts from `ij`.

After completing these `ij` steps, you're ready to move to "Step 3: Embedded Derby".

Last Published: 05/07/2020 21:39:00