HIVE - DROP DATABASE

This chapter describes how to drop a database in Hive. The usage of SCHEMA and DATABASE are same.

Drop Database Statement

Drop Database is a statement that drops all the tables and deletes the database. Its syntax is as follows:

```
DROP DATABASE StatementDROP (DATABASE|SCHEMA) [IF EXISTS] database_name
[RESTRICT|CASCADE];
```

The following queries are used to drop a database. Let us assume that the database name is **userdb**.

```
hive> DROP DATABASE IF EXISTS userdb;
```

The following query drops the database using **CASCADE**. It means dropping respective tables before dropping the database.

```
hive> DROP DATABASE IF EXISTS userdb CASCADE;
```

The following query drops the database using **SCHEMA**.

```
hive> DROP SCHEMA userdb;
```

This clause was added in Hive 0.6.

JDBC Program

The JDBC program to drop a database is given below.

```java
import java.sql.SQLException;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import java.sql.DriverManager;

public class HiveDropDb {
   private static String driverName = "org.apache.hadoop.hive.jdbc.HiveDriver";

   public static void main(String[] args) throws SQLException {

      // Register driver and create driver instance
      Class.forName(driverName);

      // get connection
      Connection con = DriverManager.getConnection("jdbc:hive://localhost:10000/default",
"", "");
      Statement stmt = con.createStatement();
      stmt.executeQuery("DROP DATABASE userdb");

      System.out.println("Drop userdb database successful.");

      con.close();
   }
}
```

Save the program in a file named HiveDropDb.java. Given below are the commands to compile and execute this program.

```
$ javac HiveDropDb.java
$ java HiveDropDb
```

Output:

```
Drop userdb database successful.
```