

EX-5 Create tables in Hive and write queries to access the data in the table

AIM:

To create tables in Hive and write queries to access the data in the table.

PROCEDURE:

Installation of Hive:

1. Installing Apache Derby

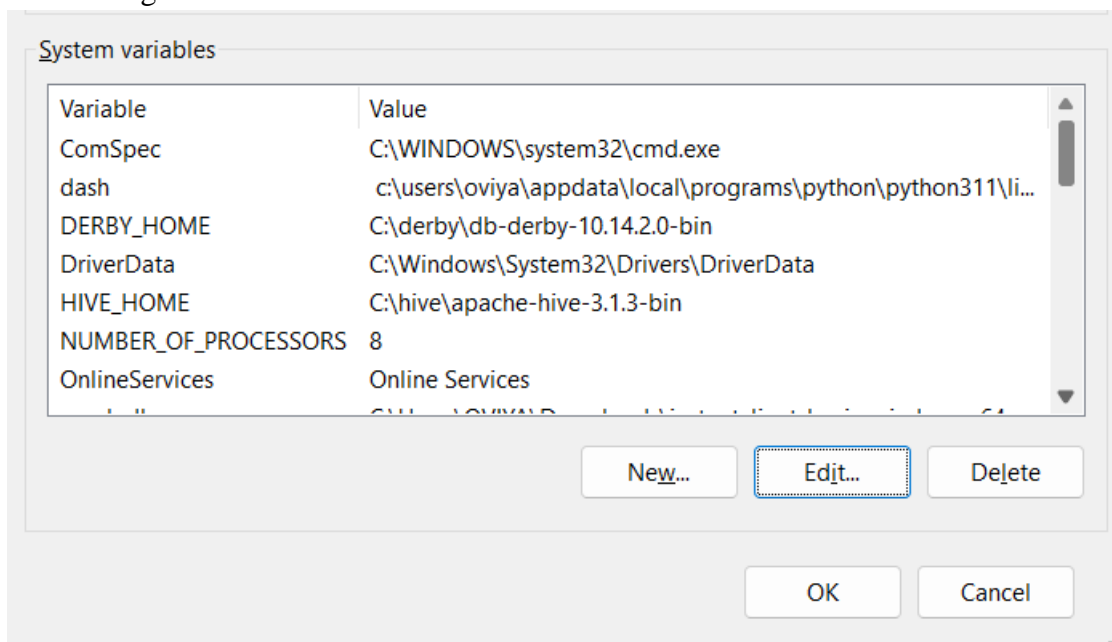
Install Apache Derby 10.14.2.0

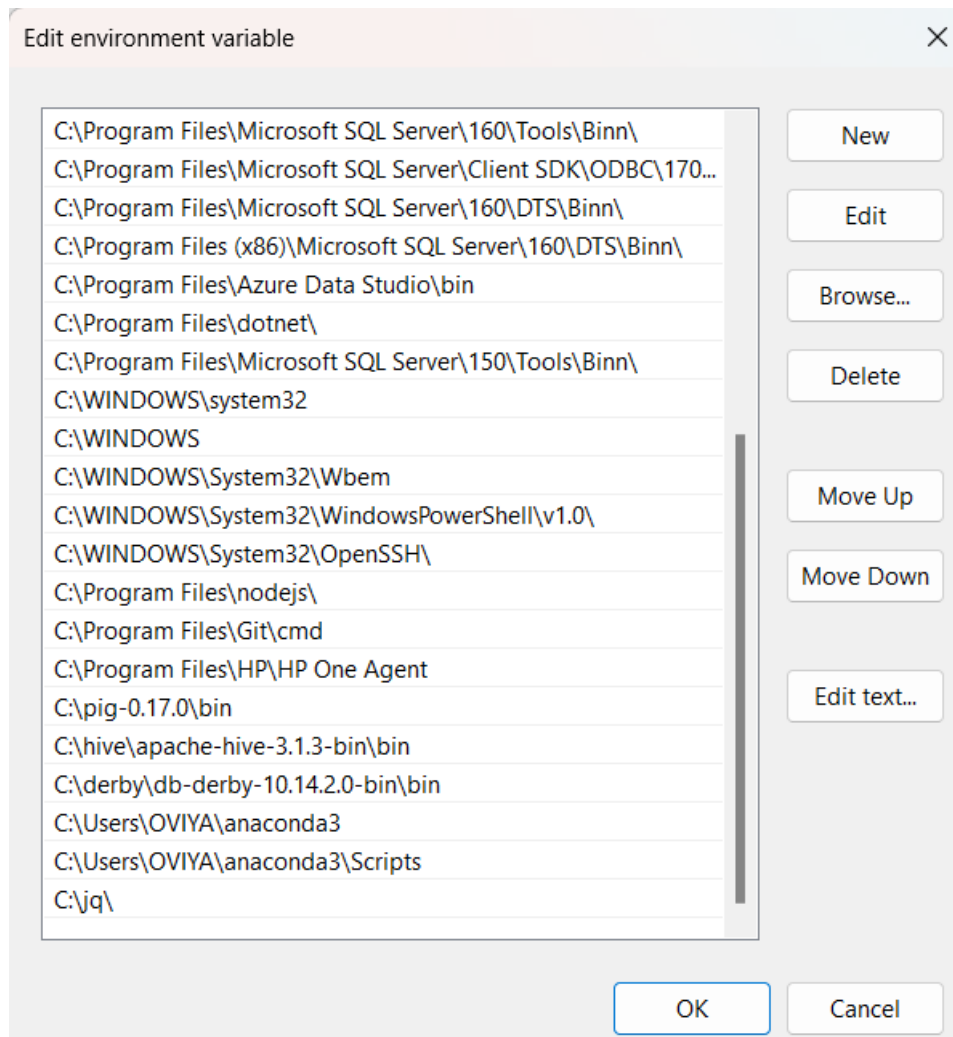
https://db.apache.org/derby/derby_downloads.html#For+Java+8+and+Higher

2. Downloading Apache Hive binaries

<https://downloads.apache.org/hive/hive-3.1.2/>

3. Setting environment variables





3.1. Copy Derby libraries

we should go to the Derby libraries directory (db-derby-10.14.2.0\lib) and copy all *.jar files. Then, we should paste them within the Hive libraries directory.

3.2. Configuring hive-site.xml and Hive's Bin folder

Refer following link to download the file. Also download the guava file. Put hive-site.xml file to hive's conf location and replace hive's current guava file with this one in lib location. Also download the bin folder from link and replace the existing hive's bin folder.

<https://1drv.ms/f/s!ArSg3Xpur4Grmw0SDqW0g44T7HYU?e=wDsoBn>

4. Starting Hadoop Services

`start-all.cmd`

5. Derby Network Server: Run the following command in separate window to open Derby

```
StartNetworkServer -h 0.0.0.0
```

6. Starting Apache Hive

Go to Apache Hive's bin location with cd command and run the following command:

```
hive --service schematool -dbType derby -initSchema
```

7. Open Hive shell by typing:

```
hive
```

Create a Database:

Start by creating a database. Open the Hive CLI and follow the steps below:

1. Use the **CREATE DATABASE** statement to create a new database:

```
CREATE DATABASE sample;
```

2. Verify the database is present:

```
SHOW DATABASES;
```

3. Switch to the new database:

```
USE sample;
```

Create a Table in Hive

```
CREATE TABLE students (id INT, name STRING, age INT);
```

Add Data

```
INSERT INTO students VALUES(1,'Oviya',20);
```

```
INSERT INTO students VALUES(2,'Navya',21);
```

```
INSERT INTO students VALUES(3,'Bavya',23);
```

List Hive Tables and Data

To show all tables in a selected database, use the following statement:

```
SHOW TABLES;
```

To show table column names and data types, run:

```
DESC students;
```

To display table data, use a **SELECT** statement. For example, to select everything in a table, run:

```
SELECT * FROM students;
```

OUTPUT:

```
C:\Windows\System32>cd C:/hadoop/sbin

C:\hadoop\sbin>start-all
This script is Deprecated. Instead use start-dfs.cmd and start-yarn.cmd
starting yarn daemons

C:\hadoop\sbin>jps
20756 DataNode
20008 ResourceManager
21944 NodeManager
23996 Jps
6476 NameNode
```

```
C:\Windows\System32>StartNetworkServer -h 0.0.0.0
Fri Aug 30 18:13:29 IST 2024 : Security manager installed using the Basic server security policy.
Fri Aug 30 18:13:29 IST 2024 : Apache Derby Network Server - 10.14.2.0 - (1828579) started and ready to accept connections on port 1527
```

```
C:\hadoop\sbin>cd C:\hive\apache-hive-3.1.3-bin\bin

C:\hive\apache-hive-3.1.3-bin\bin>hive --service schematool -dbType derby -initSchema
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/C:/hadoop/share/hadoop/common/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/C:/hive/apache-hive-3.1.3-bin/lib/log4j-slf4j-impl-2.17.1.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]
```

```
Initialization script completed
schemaTool completed
```

```
C:\hive\apache-hive-3.1.3-bin\bin>hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/C:/hadoop/share/hadoop/common/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/C:/hive/apache-hive-3.1.3-bin/lib/log4j-slf4j-impl-2.17.1.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]
```

```
hive> CREATE TABLE student(
  > id INT,
  > name STRING,
  > age INT
  > );
```

```
2024-08-30 18:30:05,439 INFO conf.HiveConf: Using the default configuration
2024-08-30 18:30:05,439 INFO session.SessionState: Resolving dependencies
hive> INSERT INTO student VALUES (3,'Navya',20);
```

```
cff): SELECT * FROM student
2024-08-30 18:30:05,395 INFO ql.Driver: Completed executing command(query
-8a40d8602cff); Time taken: 0.001 seconds
OK
2024-08-30 18:30:05,396 INFO ql.Driver: OK
2024-08-30 18:30:05,396 INFO ql.Driver: Concurrency mode is disabled, not
2024-08-30 18:30:05,402 INFO mapred.FileInputFormat: Total input files to
2024-08-30 18:30:05,430 INFO exec.TableScanOperator: RECORDS_OUT_INTERMED
2024-08-30 18:30:05,430 INFO exec.SelectOperator: RECORDS_OUT_INTERMEDIAT
2024-08-30 18:30:05,431 INFO exec.ListSinkOperator: RECORDS_OUT_INTERMEDI
1      John Doe      22
2      Oviya      20
3      Navya      20
```

RESULT:

Thus to create tables in Hive and write queries to access the data in the table is completed successfully.