

# **Install Apache Hive 4.0 on Windows 10 and Configure 3 Metastore Modes**

*Author: Sri Adilakshmi M*

## Table of Contents

<b>1. Overview:</b> .....	4
<b>2. Prerequisites:</b> .....	5
<b>3. Download Hive Binaries:</b> .....	5
<b>4. Set up Environment Variables:</b> .....	9
<b>5. Verify Hive Installation:</b> .....	13
<b>6. Start Hadoop Services:</b> .....	13
<b>7. Configure Embedded Derby Metastore:</b> .....	16
<b>7.1. Initialize Hive Metastore:</b> .....	16
<b>7.2. Start Beeline CLI:</b> .....	18
<b>7.3. Run Queries on Beeline CLI:</b> .....	20
<b>8. Configure Local Derby Metastore:</b> .....	27
<b>8.1. Install Apache Derby:</b> .....	27
<b>8.2. Set up Environment Variables:</b> .....	30
<b>8.3. Start Derby Network Server:</b> .....	33
<b>8.4. Initialize Local Metastore:</b> .....	34
<b>8.5. Configure Hive Site:</b> .....	35
<b>8.6. Copy Derby Libraries:</b> .....	35
<b>8.7. Run Beeline CLI:</b> .....	36
<b>8.8. Start HiveServer2 Service:</b> .....	41
<b>8.9. Beeline Remote Connection:</b> .....	42
<b>9. Configure Remote MySQL Metastore:</b> .....	44
<b>9.1. Install MySQL Server:</b> .....	44
<b>9.2. Create Metastore DB in MySQL:</b> .....	45
<b>9.3. Download MySQL JDBC Driver:</b> .....	47
<b>9.4. Configure Hive Site File:</b> .....	48
<b>9.5. Initialize Metastore DB:</b> .....	49
<b>9.6. Verify Metastore in MySQL:</b> .....	50
<b>9.7. Start Hive Metastore service:</b> .....	51

<b>9.8. Start HiveServer2 Service:</b> .....	52
<b>9.9. Run Queries in Beeline CLI:</b> .....	56
<b>9.10. Verify Metadata in MySQL:</b> .....	69
<b>10. Hive Web UI:</b> .....	71
<b>11. HCatalog and WebHCat:</b> .....	71
<b>11.1. Create Symbolic Link for Cygwin:</b> .....	72
<b>11.2. Setup Env variables for Cygwn:</b> .....	72
<b>11.3. Start HCatalog CLI:</b> .....	73
<b>11.4. Start WebHCat server:</b> .....	76

This document outlines the steps needed to install Apache Hive on Windows Operating system.

## 1. Overview:

**Apache Hive** was developed by Facebook and became an open-source ETL and data warehousing tool which is built on top of Hadoop for analyzing, querying and managing large datasets stored in HDFS. Hive uses **HQL** (Hive Query Language) as a processing engine that processes HDFS datasets such that queries executed from Hive are internally converted into MapReduce tasks for parallel computation and distribution of data.

The key components of Apache Hive include Hive CLI, Beeline CLI, HiveServer2, Hive Web Interface, Hive Driver, Hive Metastore, HCatalog and WebHCat.

**Hive Metastore** is a critical component of Hive because it is the central schema repository that stores Hive metadata including tables, columns, datatypes, data locations etc. created by Hive and this schema repository can be used by other data processing tools such as Spark, Pig etc.

Hive Metastore works in three different modes:

1. **Embedded Metastore:** In this mode, Hive Metastore service runs in the same JVM where Hive Driver service runs and it uses **Apache Derby** as metastore database that is stored on the local file system. This is the default metastore that comes with Hive installation and is used for testing purposes only. Only one embedded Derby database can access database files at any time so only one Hive session can be opened and if we try to start second Hive session, it errors out. To allow multiple Hive sessions, we can configure Derby to run as Network Server.
2. **Local Metastore:** In this mode, Hive Metastore and Hive Driver still run within the same JVM process but metastore service connects to a JDBC supported database such as MySQL that runs on a different JVM in the same machine or on different machine. Local metastore currently supports **Derby, MySQL, MSSQL, Oracle and Postgres** database systems only.
3. **Remote Metastore:** In this mode, Hive Metastore service runs in a different JVM but not in Hive Driver service JVM and metastore service connects to a remote database which could be MySQL, MSSQL, Oracle or Postgres. In Remote Metastore, Hive Client will make a connection to Hive Metastore using Thrift protocol, and Metastore server in turn communicates with the database and run queries.

This document provides instructions to install Hive 4.x version on top of Hadoop 3.x.

Note that Hive CLI has been deprecated in Hive 4.x.

## 2. Prerequisites:

The following prerequisites need to be installed before running Hive.

1. **Hadoop:** Before installing Hive, Hadoop cluster must have been installed and running. Go through [these steps](#) to install Hadoop on Windows operating system.
2. **File Archiver:** Any file archiver such as **7zip** or **WinRAR** is needed to unzip the downloaded Hive binaries. 7zip can be downloaded from the [7zip Downloads](#) website and WinRAR can be downloaded from the [RAR lab Downloads](#) website.
3. **Cygwin:** Since some Hive utilities are not compatible with Windows, we will need the Cygwin tool to run some Linux commands. You can go through [these steps](#) to install Cygwin.

## 3. Download Hive Binaries:

After installing prerequisites, download the latest Hive 4.0.0 release from the [Apache Hive Downloads](#) mirror website.

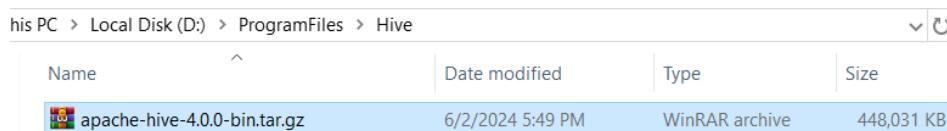
The screenshot shows the Apache Software Foundation website at https://www.apache.org/dyn/closer.cgi/hive/. The page features the Apache logo and navigation links for Community, Projects, Downloads, Learn, Resources & Tools, About, and Search. Below the navigation, it says "We suggest the following location for your download: <https://dlcdn.apache.org/hive/>". It also lists alternate download locations and instructions for verifying file integrity using PGP signatures or hashes. Sections for HTTP download and BACKUP SITES are shown, along with a "VERIFY THE INTEGRITY OF THE FILES" section containing detailed instructions and links to PGP keys.

Go to the [suggested location](#) for download and click on [hive-4.0.0/](#) from where you need to download the binary file named apache-hive-4.0.0-bin.tar.gz file which gets downloaded to your **Downloads** folder.

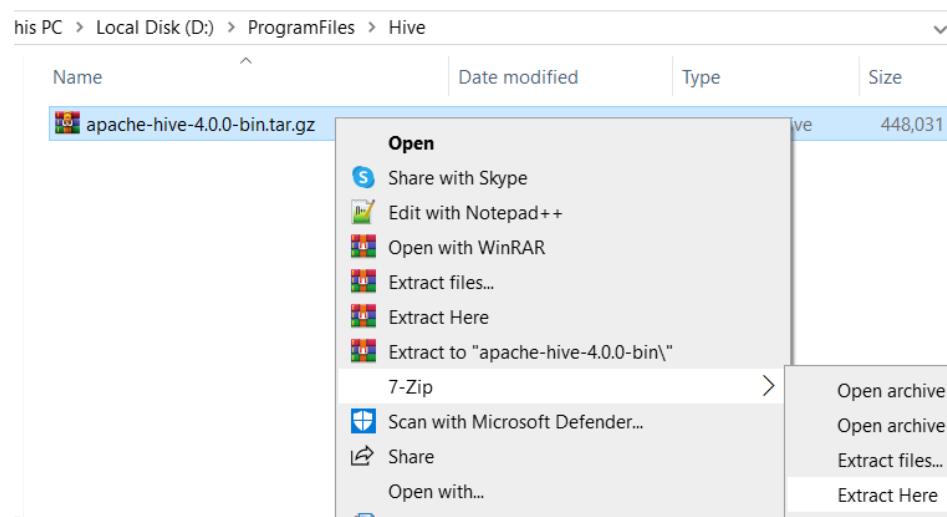
Name	Last modified	Size	Description
Parent Directory	-		
<a href="#">apache-hive-4.0.0-bin.tar.gz</a>	2024-03-25 20:58	438M	
<a href="#">apache-hive-4.0.0-bin.tar.gz.asc</a>	2024-03-25 20:58	862	
<a href="#">apache-hive-4.0.0-bin.tar.gz.sha256</a>	2024-03-25 20:58	95	
<a href="#">apache-hive-4.0.0-src.tar.gz</a>	2024-03-25 20:58	60M	
<a href="#">apache-hive-4.0.0-src.tar.gz.asc</a>	2024-03-25 20:58	862	
<a href="#">apache-hive-4.0.0-src.tar.gz.sha256</a>	2024-03-25 20:58	95	

After the binary file is downloaded, unpack it using any file archiver (7zip or WinRAR) utility as below:

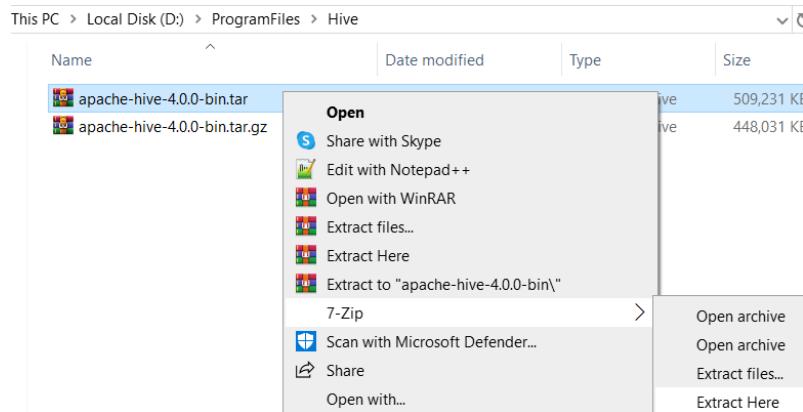
- Choose the installation directory in your machine and copy apache-hive-4.0.0-bin.tar.gz file to that directory. Here, we are choosing Hive installation directory as D:\ProgramFiles\Hive.



- Right click on the file and choose **7-Zip -> Extract Here** option which extracts a new packed file apache-hive-4.0.0-bin.tar.



- Next, unpack apache-hive-4.0.0-bin.tar file using 7zip utility.



- The tar file extraction may take few minutes to finish. After finishing, you see a folder named apache-hive-4.0.0-bin which consists of Hive binaries and libraries.

Name	Date modified	Type	Size
bin	6/2/2024 5:55 PM	File folder	
conf	6/2/2024 5:55 PM	File folder	
contrib	6/2/2024 5:55 PM	File folder	
examples	6/2/2024 5:55 PM	File folder	
hcatalog	6/2/2024 5:55 PM	File folder	
jdbc	6/2/2024 5:55 PM	File folder	
lib	6/2/2024 5:55 PM	File folder	
licenses	6/2/2024 5:55 PM	File folder	
scripts	6/2/2024 5:55 PM	File folder	
LICENSE	1/22/2020 8:40 PM	File	19 KB
licenses.xml	1/22/2020 8:40 PM	Microsoft Edge HT...	145 KB
NOTICE	1/22/2020 8:40 PM	File	1 KB
RELEASE_NOTES.txt	1/22/2020 8:40 PM	Text Document	23 KB

#### Note:

By default, Apache Hive is built to run on Linux Operating system. To make it running on Windows OS, we should use **Cygwin** utility to execute Linux commands from Windows. However, we can run some Hive utilities directly on Windows without Cygwin by downloading .cmd files from the [HadiFadl GitHub repository](#). These .cmd files are sufficient to start Hive from Windows command line but you will not be able to run some utilities such as schematool, metastore, etc. directly in which case Cygwin is required.

Alternatively, I would suggest you to download .cmd files from [my GitHub repository](#) for the corresponding Hive version. For this installation, download files from [hive-4.0.0](#) and paste under the exact folder structure where ever Hive is installed. This makes you to start Hive as well as run other utilities including schematool, metastore, metatool, etc. from Windows itself.

- Copy .cmd files in [hive-4.0.0/bin](#) folder to D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\bin folder.

Name	Date modified	Type	Size
ext	6/2/2024 5:55 PM	File folder	
beeline	1/22/2020 8:40 PM	File	1 KB
beeline.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
derbyserver.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
hive	1/22/2020 8:40 PM	File	11 KB
hive.cmd	6/2/2024 12:33 PM	Windows Comma...	8 KB
hive-config.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
hive-config.sh	1/22/2020 8:40 PM	SH File	3 KB
hiveserver2	1/22/2020 8:40 PM	File	1 KB
hiveserver2.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
hqlsql	1/22/2020 8:40 PM	File	1 KB
hqlsql.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
init-hive-dfs.cmd	6/2/2024 12:33 PM	Windows Comma...	3 KB
init-hive-dfs.sh	1/22/2020 8:40 PM	SH File	3 KB
metatool	1/22/2020 8:40 PM	File	1 KB
metatool.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
repistats.sh	1/22/2020 8:40 PM	SH File	6 KB
schematool	1/22/2020 8:40 PM	File	1 KB
schematool.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB

- Copy .cmd files in [hive-4.0.0/bin/ext](#) folder to D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\bin\ext folder.

Name	Date modified	Type	Size
util	1/22/2020 8:40 PM	File folder	
beeline.cmd	6/2/2024 12:33 PM	Windows Comma...	3 KB
beeline.sh	1/22/2020 8:40 PM	SH File	3 KB
cleardanglingscratchdir.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
cleardanglingscratchdir.sh	1/22/2020 8:40 PM	SH File	2 KB
cli.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
cli.sh	1/22/2020 8:40 PM	SH File	2 KB
debug.cmd	6/2/2024 12:33 PM	Windows Comma...	4 KB
debug.sh	1/22/2020 8:40 PM	SH File	4 KB
fixacidkeyindex.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
fixacidkeyindex.sh	1/22/2020 8:40 PM	SH File	2 KB
help.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
help.sh	1/22/2020 8:40 PM	SH File	2 KB
hiveburninclient.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
hiveburninclient.sh	1/22/2020 8:40 PM	SH File	2 KB
hiveserver2.cmd	6/2/2024 12:33 PM	Windows Comma...	4 KB
hiveserver2.sh	1/22/2020 8:40 PM	SH File	4 KB
hqlsql.cmd	6/2/2024 12:33 PM	Windows Comma...	3 KB
hqlsql.sh	1/22/2020 8:40 PM	SH File	2 KB
jar.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB

- Copy .cmd files in [hive-4.0.0/bin/ext/util](#) folder to D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\bin\ext folder.

Name	Date modified	Type	Size
execHiveCmd.cmd	6/2/2024 12:33 PM	Windows Comma...	2 KB
execHiveCmd.sh	1/22/2020 8:40 PM	SH File	2 KB

## 4. Set up Environment Variables:

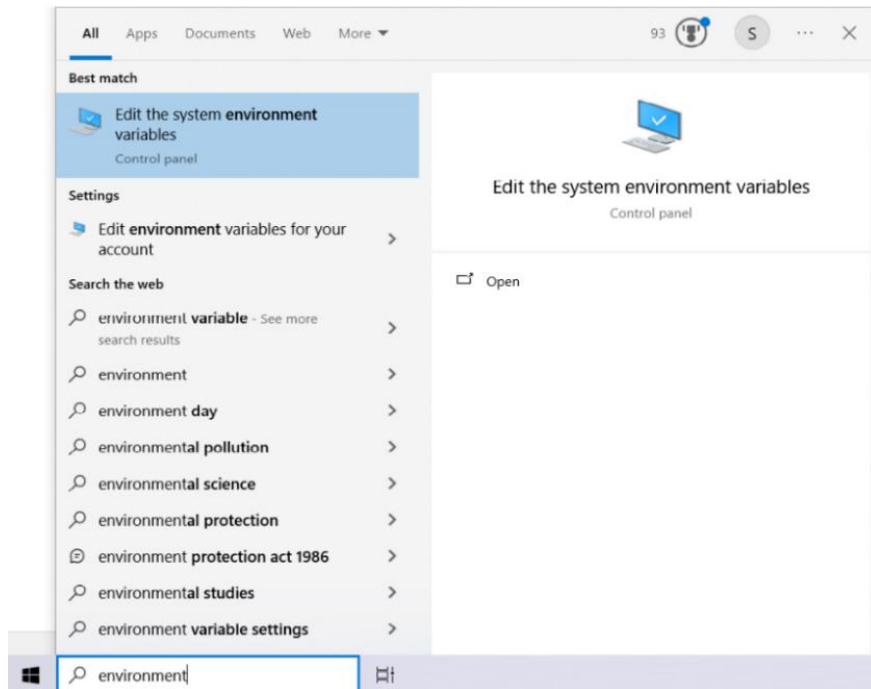
After installing Hadoop prerequisite and Hive binaries, we should configure two environment variables defining Hive installation path.

- **HIVE\_HOME**: This is the Hive installation directory path in the machine (*in our case, it is D:\Programs\Hive\apache-hive-4.0.0-bin*)
- **HADOOP\_USER\_CLASSPATH\_FIRST**: Set this variable value to `true` for Hive to use Hadoop user Class path first. This ensures `log4j2.x` and `jline` jars are loaded ahead of the jars pulled by Hadoop.

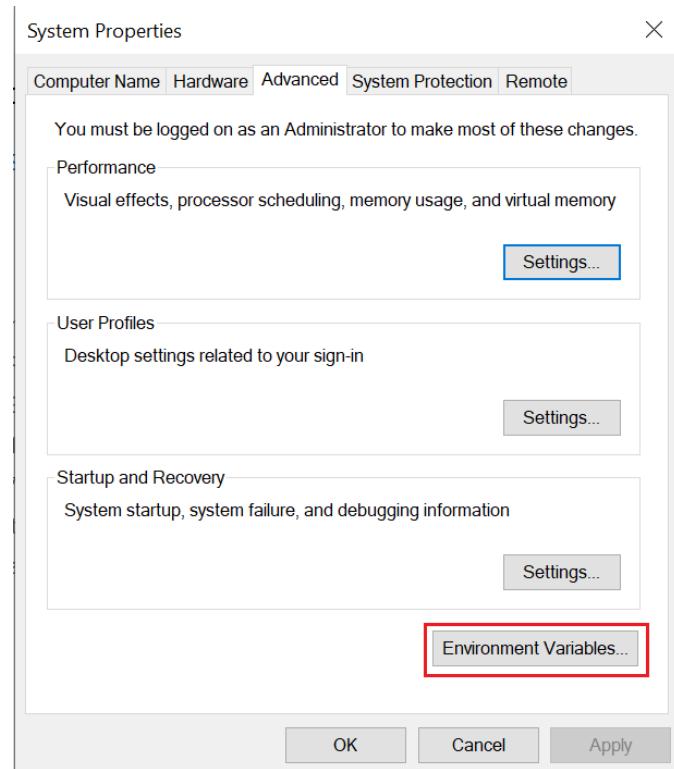
These variables need to be added to either **User environment variables** or **System environment variables** depending on Hive configuration needed **for a single user or for multiple users**.

In this tutorial, we will add User environment variables since we are configuring Hive for a single user. If you would like to configure Hive for multiple users, then define System environment variables.

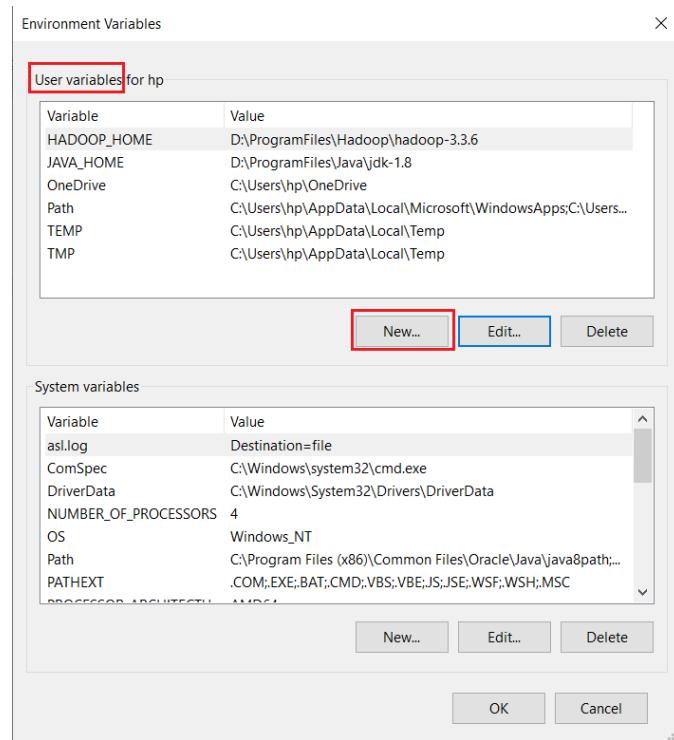
In the Windows search bar, start typing “environment variables” and select the first match which opens up **System Properties** dialog.



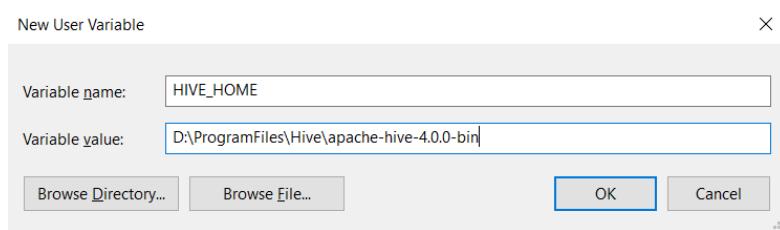
On the **System Properties** window, press **Environment Variables** button.



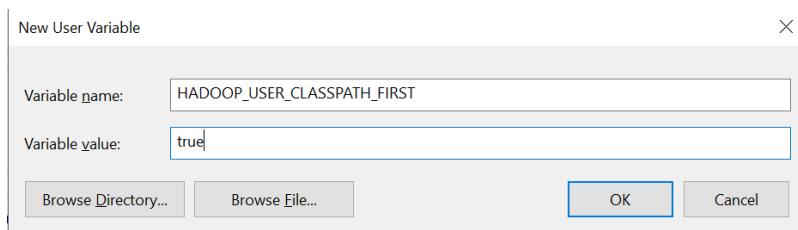
On the **Environment Variables** dialog, press **New** under **User variables** section.



Add **HIVE\_HOME** variable and press OK.

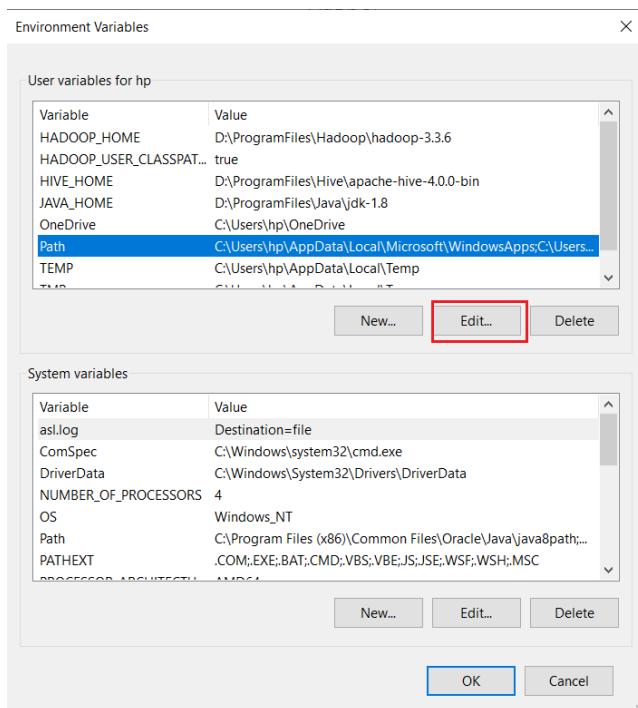


Press **New** again and add **HADOOP\_USER\_CLASSPATH\_FIRST** variable to **true** and press OK.

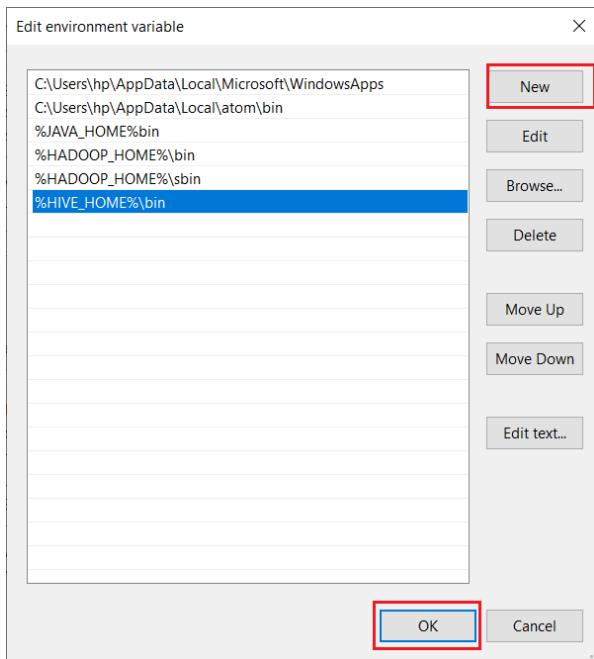


Now, we will update **PATH** variable to add Hive binary path

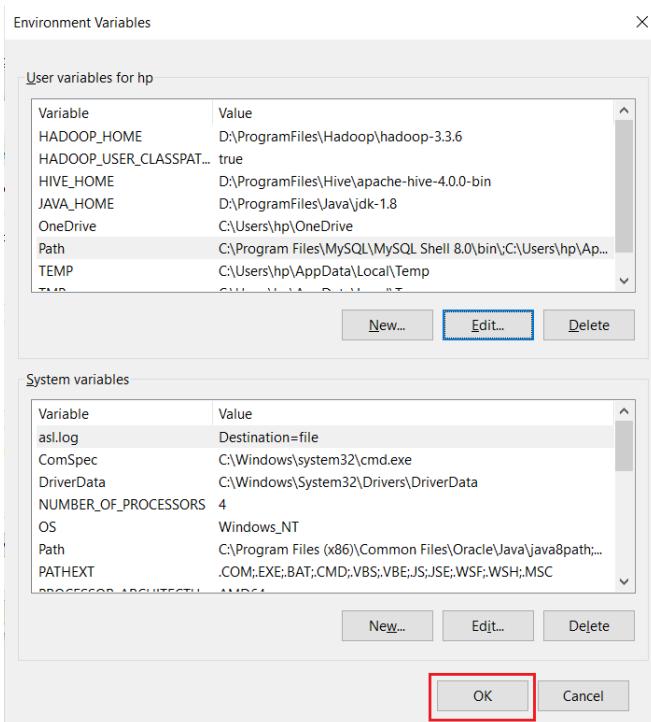
Select **PATH** variable under **User variables** and press **Edit** button.



Press **New** and add `%HIVE_HOME%\bin` path and press **OK**.



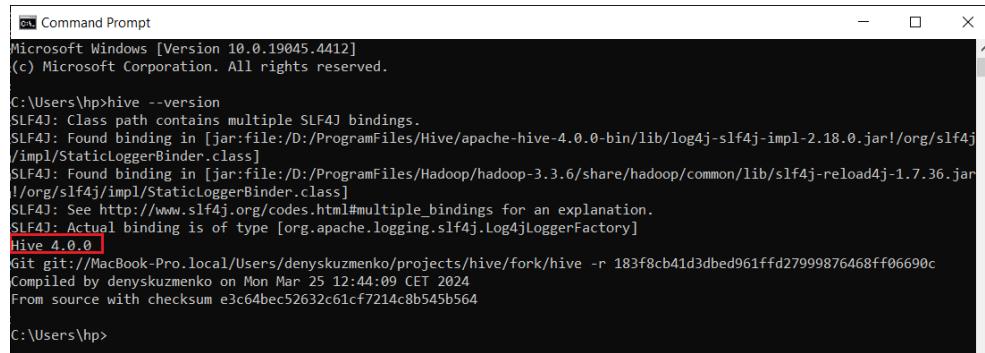
Press **OK** to apply environment variable changes and close window.



## 5. Verify Hive Installation:

Open **Windows PowerShell** or **Command Prompt** and run the following command to verify if Hive is installed properly:

```
hive --version
```



```
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>hive --version
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Hive 4.0.0
Compiled by denyskuzmenko on Mon Mar 25 12:44:09 CET 2024
From source with checksum e3c64bec52632c61cf7214c8b545b564

C:\Users\hp>
```

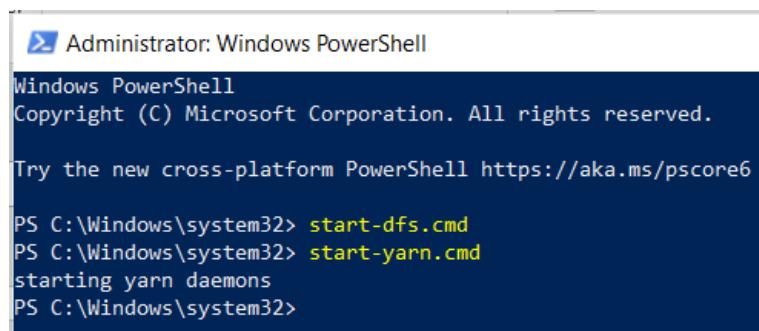
This may take couple of minutes to complete and displays **Hive 4.0.0** version installed.

## 6. Start Hadoop Services:

Before starting Hive, Hadoop services must be running since Hive runs on top of HDFS.

Open **Windows Command Prompt** or **Windows PowerShell** in **Administrator** mode and run the following commands to start Hadoop services.

```
start-dfs.cmd
start-yarn.cmd
```



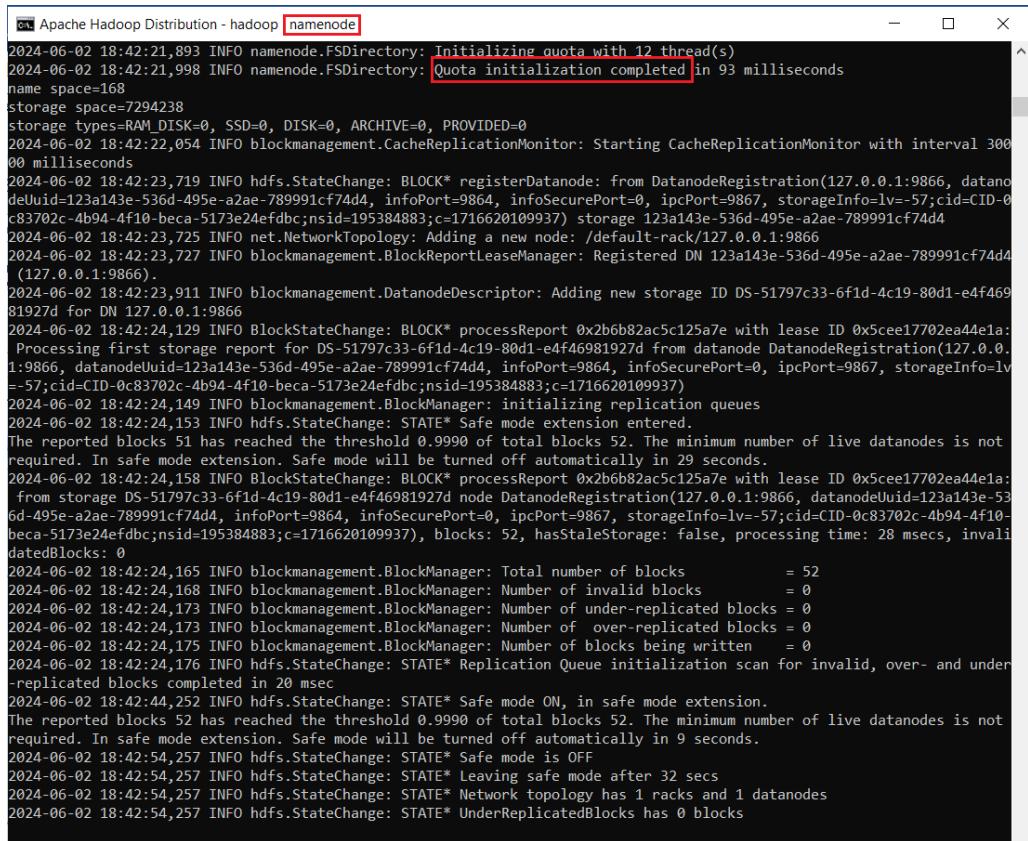
```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> start-dfs.cmd
PS C:\Windows\system32> start-yarn.cmd
starting yarn daemons
PS C:\Windows\system32>
```

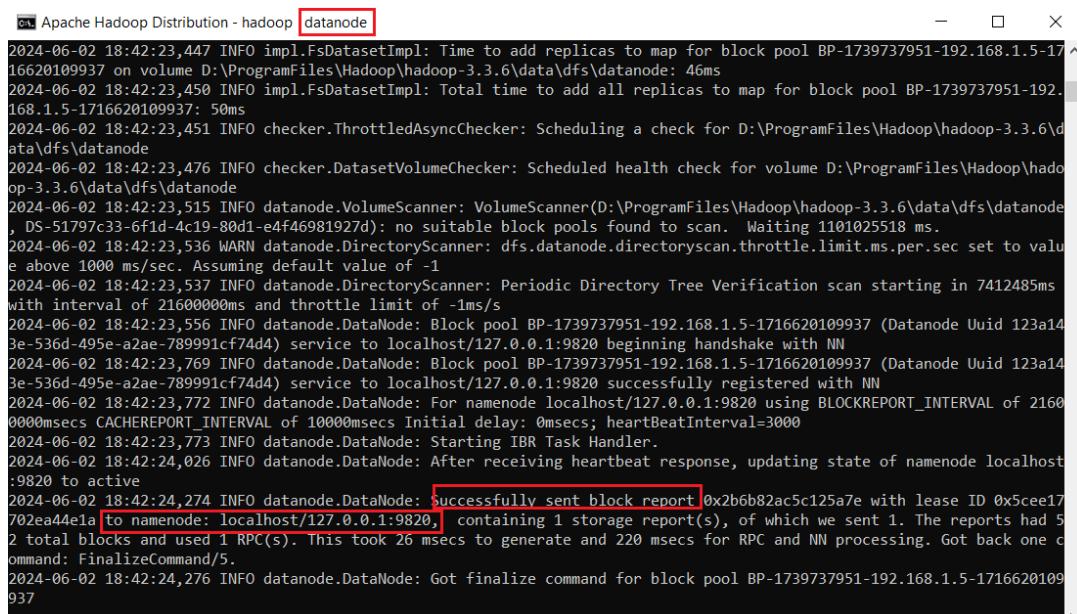
After executing the above commands, we can see four Windows command prompts opened for **namenode**, **datanode**, **resourcemanager** and **nodemanager** as below:

Wait for namenode service to say “Quota initialization completed”.



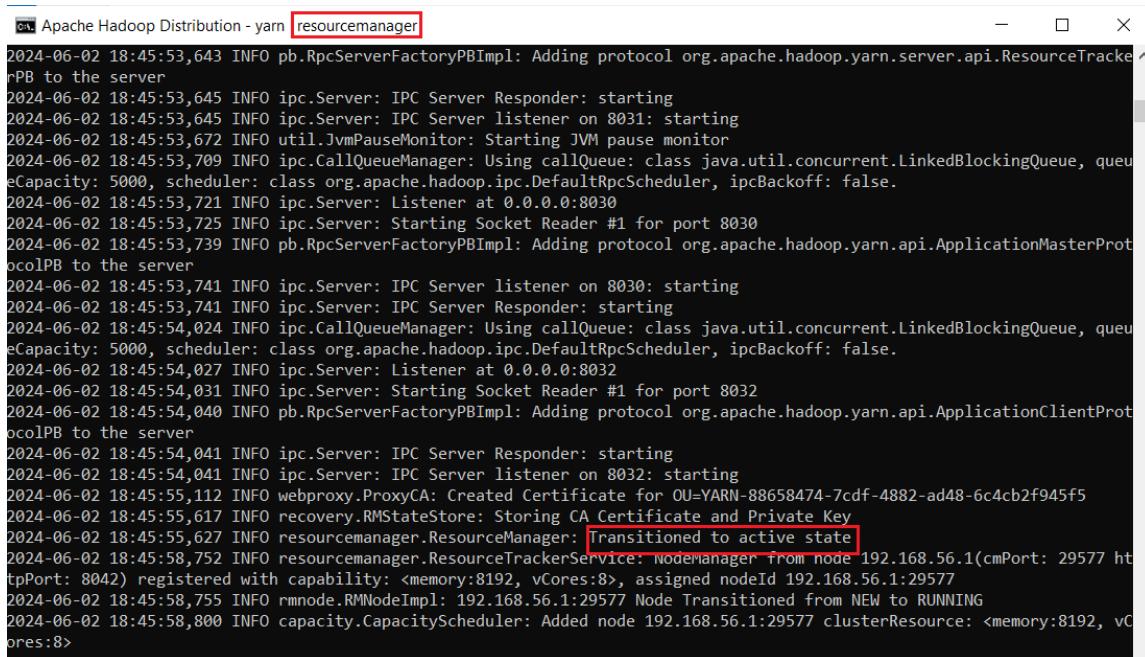
```
Apache Hadoop Distribution - hadoop [namenode]
2024-06-02 18:42:21,893 INFO namenode.FSDirectory: Initializing quota with 12 thread(s)
2024-06-02 18:42:21,998 INFO namenode.FSDirectory: Quota initialization completed in 93 milliseconds
name space=168
storage space=7294238
storage types=RAM_DISK=0, SSD=0, DISK=0, ARCHIVE=0, PROVIDED=0
2024-06-02 18:42:22,054 INFO blockmanagement.CacheReplicationMonitor: Starting CacheReplicationMonitor with interval 300
00 milliseconds
2024-06-02 18:42:23,719 INFO hdfs.StateChange: BLOCK* registerDatanode: from DatanodeRegistration(127.0.0.1:9866, datanodeUuid=123a143e-536d-495e-a2ae-789991cf74d4, infoPort=9864, infoSecurePort=0, ipcPort=9867, storageInfo=lv=-57;cid=CID-0c83702c-4b94-4f10-beca-5173e24efdbc;nsid=195384883;c=1716620109937) storage 123a143e-536d-495e-a2ae-789991cf74d4
2024-06-02 18:42:23,725 INFO net.NetworkTopology: Adding a new node: /default-rack/127.0.0.1:9866
2024-06-02 18:42:23,727 INFO blockmanagement.BlockReportLeaseManager: Registered DN 123a143e-536d-495e-a2ae-789991cf74d4
(127.0.0.1:9866).
2024-06-02 18:42:23,911 INFO blockmanagement.DatanodeDescriptor: Adding new storage ID DS-51797c33-6f1d-4c19-80d1-e4f46981927d for DN 127.0.0.1:9866
2024-06-02 18:42:24,129 INFO BlockStateChange: BLOCK* processReport 0x2b6b82ac5c125a7e with lease ID 0x5cee17702ea44e1a: Processing first storage report for DS-51797c33-6f1d-4c19-80d1-e4f46981927d from datanode DatanodeRegistration(127.0.0.1:9866, datanodeUuid=123a143e-536d-495e-a2ae-789991cf74d4, infoPort=9864, infoSecurePort=0, ipcPort=9867, storageInfo=lv=-57;cid=CID-0c83702c-4b94-4f10-beca-5173e24efdbc;nsid=195384883;c=1716620109937)
2024-06-02 18:42:24,149 INFO blockmanagement.BlockManager: initializing replication queues
2024-06-02 18:42:24,153 INFO hdfs.StateChange: STATE* Safe mode extension entered.
The reported blocks 51 has reached the threshold 0.9990 of total blocks 52. The minimum number of live datanodes is not required. In safe mode extension. Safe mode will be turned off automatically in 29 seconds.
2024-06-02 18:42:24,158 INFO BlockStateChange: BLOCK* processReport 0x2b6b82ac5c125a7e with lease ID 0x5cee17702ea44e1a: from storage DS-51797c33-6f1d-4c19-80d1-e4f46981927d node DatanodeRegistration(127.0.0.1:9866, datanodeUuid=123a143e-536d-495e-a2ae-789991cf74d4, infoPort=9864, infoSecurePort=0, ipcPort=9867, storageInfo=lv=-57;cid=CID-0c83702c-4b94-4f10-beca-5173e24efdbc;nsid=195384883;c=1716620109937), blocks: 52, hasStaleStorage: false, processing time: 28 msec, invalidatedBlocks: 0
2024-06-02 18:42:24,165 INFO blockmanagement.BlockManager: Total number of blocks = 52
2024-06-02 18:42:24,168 INFO blockmanagement.BlockManager: Number of invalid blocks = 0
2024-06-02 18:42:24,173 INFO blockmanagement.BlockManager: Number of under-replicated blocks = 0
2024-06-02 18:42:24,175 INFO blockmanagement.BlockManager: Number of over-replicated blocks = 0
2024-06-02 18:42:24,176 INFO hdfs.StateChange: STATE* Replication Queue initialization scan for invalid, over- and under-replicated blocks completed in 20 msec
2024-06-02 18:42:44,252 INFO hdfs.StateChange: STATE* Safe mode ON, in safe mode extension.
The reported blocks 52 has reached the threshold 0.9990 of total blocks 52. The minimum number of live datanodes is not required. In safe mode extension. Safe mode will be turned off automatically in 9 seconds.
2024-06-02 18:42:54,257 INFO hdfs.StateChange: STATE* Safe mode is OFF
2024-06-02 18:42:54,257 INFO hdfs.StateChange: STATE* Leaving safe mode after 32 secs
2024-06-02 18:42:54,257 INFO hdfs.StateChange: STATE* Network topology has 1 racks and 1 datanodes
2024-06-02 18:42:54,257 INFO hdfs.StateChange: STATE* UnderReplicatedBlocks has 0 blocks
```

Wait for datanode service to say “Successfully sent block report to namenode: localhost/127.0.0.1:9820”.



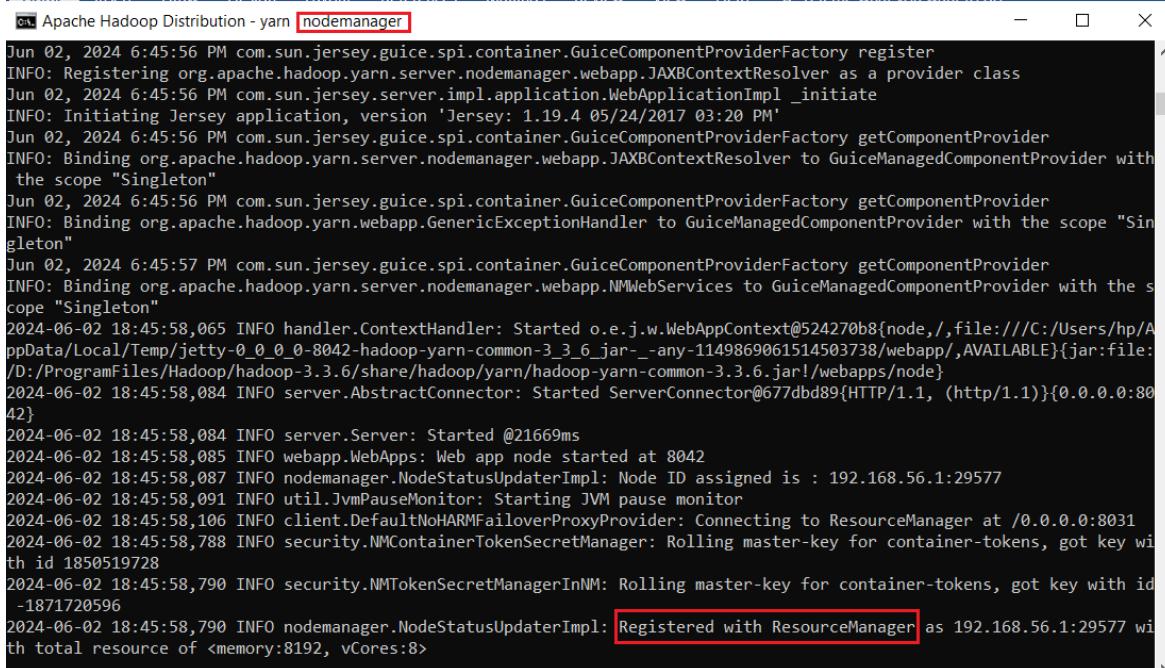
```
Apache Hadoop Distribution - hadoop [datanode]
2024-06-02 18:42:23,447 INFO impl.FsDatasetImpl: Time to add replicas to map for block pool BP-1739737951-192.168.1.5-1716620109937 on volume D:\ProgramFiles\Hadoop\hadoop-3.3.6\data\dfs\datanode: 46ms
2024-06-02 18:42:23,450 INFO impl.FsDatasetImpl: Total time to add all replicas to map for block pool BP-1739737951-192.168.1.5-1716620109937: 50ms
2024-06-02 18:42:23,451 INFO checker.ThrottledAsyncChecker: Scheduling a check for D:\ProgramFiles\Hadoop\hadoop-3.3.6\data\dfs\datanode
2024-06-02 18:42:23,476 INFO checker.DatasetVolumeChecker: Scheduled health check for volume D:\ProgramFiles\Hadoop\hadoop-3.3.6\data\dfs\datanode
2024-06-02 18:42:23,515 INFO datanode.VolumeScanner: VolumeScanner(D:\ProgramFiles\Hadoop\hadoop-3.3.6\data\dfs\datanode, DS-51797c33-6f1d-4c19-80d1-e4f46981927d): no suitable block pools found to scan. Waiting 1101025518 ms.
2024-06-02 18:42:23,536 WARN datanode.DirectoryScanner: dfs.datanode.directoryscan.throttle.limit.ms.per.sec set to value above 1000 ms/sec. Assuming default value of -1
2024-06-02 18:42:23,537 INFO datanode.DirectoryScanner: Periodic Directory Tree Verification scan starting in 7412485ms with interval of 21600000ms and throttle limit of -1ms
2024-06-02 18:42:23,556 INFO datanode.DataNode: Block pool BP-1739737951-192.168.1.5-1716620109937 (Datanode Uuid 123a143e-536d-495e-a2ae-789991cf74d4) service to localhost/127.0.0.1:9820 beginning handshake with NN
2024-06-02 18:42:23,769 INFO datanode.DataNode: Block pool BP-1739737951-192.168.1.5-1716620109937 (Datanode Uuid 123a143e-536d-495e-a2ae-789991cf74d4) service to localhost/127.0.0.1:9820 successfully registered with NN
2024-06-02 18:42:23,772 INFO datanode.DataNode: For namenode localhost/127.0.0.1:9820 using BLOCKREPORT_INTERVAL of 2160000ms CACHEREPORT_INTERVAL of 10000ms Initial delay: 0ms; heartBeatInterval=3000
2024-06-02 18:42:23,773 INFO datanode.DataNode: Starting IBR Task Handler.
2024-06-02 18:42:24,026 INFO datanode.DataNode: After receiving heartbeat response, updating state of namenode localhost:9820 to active
2024-06-02 18:42:24,274 INFO datanode.DataNode: Successfully sent block report 0x2b6b82ac5c125a7e with lease ID 0x5cee17702ea44e1a to namenode: localhost/127.0.0.1:9820, containing 1 storage report(s), of which we sent 1. The reports had 5 total blocks and used 1 RPC(s). This took 26 msec to generate and 220 msec for RPC and NN processing. Got back one command: FinalizeCommand/5.
2024-06-02 18:42:24,276 INFO datanode.DataNode: Got finalize command for block pool BP-1739737951-192.168.1.5-1716620109937
```

Wait for resourcemanager service to say “*Transitioned to active state*”.



```
Apache Hadoop Distribution - yarn resourcemanager
2024-06-02 18:45:53,643 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.hadoop.yarn.server.api.ResourceTrackerPB to the server
2024-06-02 18:45:53,645 INFO ipc.Server: IPC Server Responder: starting
2024-06-02 18:45:53,645 INFO ipc.Server: IPC Server listener on 8031: starting
2024-06-02 18:45:53,672 INFO util.JvmPauseMonitor: Starting JVM pause monitor
2024-06-02 18:45:53,709 INFO ipc.CallQueueManager: Using callQueue: class java.util.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.hadoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2024-06-02 18:45:53,721 INFO ipc.Server: Listener at 0.0.0.0:8030
2024-06-02 18:45:53,725 INFO ipc.Server: Starting Socket Reader #1 for port 8030
2024-06-02 18:45:53,739 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.hadoop.yarn.api.ApplicationMasterProtocolPB to the server
2024-06-02 18:45:53,741 INFO ipc.Server: IPC Server listener on 8030: starting
2024-06-02 18:45:53,741 INFO ipc.Server: IPC Server Responder: starting
2024-06-02 18:45:54,024 INFO ipc.CallQueueManager: Using callQueue: class java.util.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.hadoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2024-06-02 18:45:54,027 INFO ipc.Server: Listener at 0.0.0.0:8032
2024-06-02 18:45:54,031 INFO ipc.Server: Starting Socket Reader #1 for port 8032
2024-06-02 18:45:54,040 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.hadoop.yarn.api.ApplicationClientProtocolPB to the server
2024-06-02 18:45:54,041 INFO ipc.Server: IPC Server Responder: starting
2024-06-02 18:45:54,041 INFO ipc.Server: IPC Server listener on 8032: starting
2024-06-02 18:45:55,112 INFO webproxy.ProxyCA: Created Certificate for OU=YARN-88658474-7cdf-4882-ad48-6c4cb2f945f5
2024-06-02 18:45:55,617 INFO recovery.RMStateStore: Storing CA Certificate and Private Key
2024-06-02 18:45:55,627 INFO resourcemanager.ResourceManager: Transitioned to active state
2024-06-02 18:45:58,752 INFO resourcemanager.ResourceTrackerService: nodemanager from node 192.168.56.1(cmPort: 29577 httpPort: 8042) registered with capability: <memory:8192, vCores:8>, assigned nodeID 192.168.56.1:29577
2024-06-02 18:45:58,755 INFO rmnode.RMNodeImpl: 192.168.56.1:29577 Node Transitioned from NEW to RUNNING
2024-06-02 18:45:58,800 INFO capacity.CapacityScheduler: Added node 192.168.56.1:29577 clusterResource: <memory:8192, vCores:8>
```

Wait for nodemanager service to say “*Registered with ResourceManager*”.



```
Apache Hadoop Distribution - yarn nodemanager
Jun 02, 2024 6:45:56 PM com.sun.jersey.spi.container.GuiceComponentProviderFactory register
INFO: Registering org.apache.hadoop.yarn.server.nodemanager.webapp.JAXBContextResolver as a provider class
Jun 02, 2024 6:45:56 PM com.sun.jersey.impl.application.WebApplicationImpl _initiate
INFO: Initiating Jersey application, version 'Jersey: 1.19.4 05/24/2017 03:20 PM'
Jun 02, 2024 6:45:56 PM com.sun.jersey.spi.container.GuiceComponentProviderFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.nodemanager.webapp.JAXBContextResolver to GuiceManagedComponentProvider with the scope "Singleton"
Jun 02, 2024 6:45:56 PM com.sun.jersey.spi.container.GuiceComponentProviderFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.webapp.GenericExceptionHandler to GuiceManagedComponentProvider with the scope "Singleton"
Jun 02, 2024 6:45:57 PM com.sun.jersey.spi.container.GuiceComponentProviderFactory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.nodemanager.webapp.NMWebServices to GuiceManagedComponentProvider with the scope "Singleton"
2024-06-02 18:45:58,065 INFO handler.ContextHandler: Started o.e.j.w.WebAppContext@524270b8(node,,file:///C:/Users/hp/AppData/Local/Temp/jetty-0_0_0-8042-hadoop-yarn-common-3_3_6_jar-any-1149869061514503738/webapp/,AVAILABLE){jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/yarn/hadoop-yarn-common-3.3.6.jar!/webapps/node}
2024-06-02 18:45:58,084 INFO server.AbstractConnector: Started ServerConnector@677dbd89{HTTP/1.1, (http/1.1)}{0.0.0.0:8042}
2024-06-02 18:45:58,084 INFO server.Server: Started @21669ms
2024-06-02 18:45:58,085 INFO webapp.WebApps: Web app node started at 8042
2024-06-02 18:45:58,087 INFO nodemanager.NodeStatusUpdaterImpl: Node ID assigned is : 192.168.56.1:29577
2024-06-02 18:45:58,091 INFO util.JvmPauseMonitor: Starting JVM pause monitor
2024-06-02 18:45:58,106 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8031
2024-06-02 18:45:58,788 INFO security.NMContainerTokenSecretManager: Rolling master-key for container-tokens, got key with id 1850519728
2024-06-02 18:45:58,790 INFO security.NMTokenSecretManagerInNM: Rolling master-key for container-tokens, got key with id -1871720596
2024-06-02 18:45:58,790 INFO nodemanager.NodeStatusUpdaterImpl: Registered with ResourceManager as 192.168.56.1:29577 with total resource of <memory:8192, vCores:8>
```

## 7. Configure Embedded Derby Metastore:

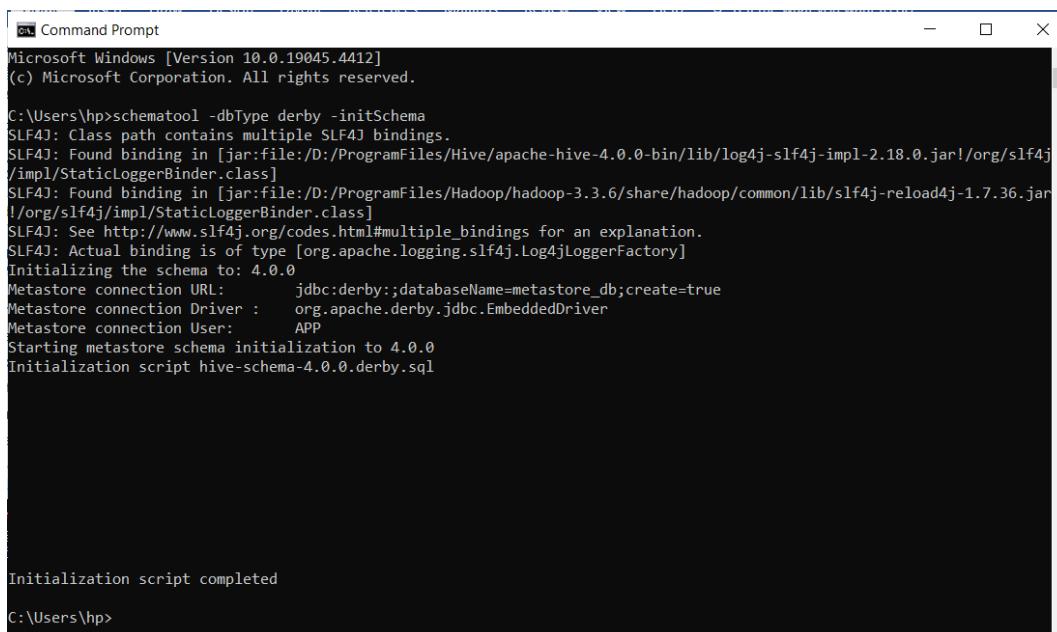
Hive by default runs with Embedded Metastore of Derby database on Hadoop File System.

### 7.1. Initialize Hive Metastore:

To start Hive, we need to initialize Hive Metastore (`metastore_db`) using `schematool` utility provided by Hive.

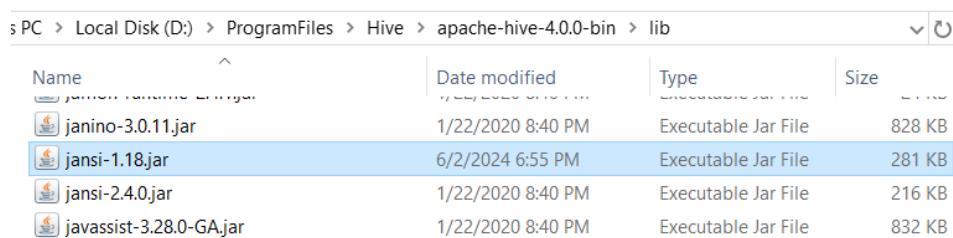
Execute the following command in Command Prompt or Windows PowerShell

```
schematool -dbType derby -initSchema
```



The screenshot shows a Microsoft Windows Command Prompt window titled "Command Prompt". The window displays the output of the `schematool -dbType derby -initSchema` command. The output includes logs from SLF4J about multiple bindings for org.apache.logging.slf4j.Log4jLoggerFactory, the initialization schema version (4.0.0), connection URL (jdbc:derby://;databaseName=metastore\_db;create=true), driver (org.apache.derby.jdbc.EmbeddedDriver), and user (APP). It also shows the starting of the metastore schema initialization and the execution of the initialization script `hive-schema-4.0.0.derby.sql`. The message "Initialization script completed" is displayed at the end, followed by the prompt `C:\Users\hp>`.

**Note:** You may encounter **Exception in thread "main" java.lang.NoSuchMethodError: org.fusesource.jansi.AnsiConsole.wrapOutputStream(Ljava/io/OutputStream;)Ljava/io/OutputStream**. To resolve this error, download `jansi-1.18.jar` file from the [official Jansi Download](#) website and copy to `HIVE_HOME\lib` directory.



The screenshot shows a Windows File Explorer window with the path `s PC > Local Disk (D:) > ProgramFiles > Hive > apache-hive-4.0.0-bin > lib`. The table below lists the files in the `lib` directory:

Name	Date modified	Type	Size
janino-3.0.11.jar	1/22/2020 8:40 PM	Executable Jar File	828 KB
jansi-1.18.jar	6/2/2024 6:55 PM	Executable Jar File	281 KB
jansi-2.4.0.jar	1/22/2020 8:40 PM	Executable Jar File	216 KB
javassist-3.28.0-GA.jar	1/22/2020 8:40 PM	Executable Jar File	832 KB

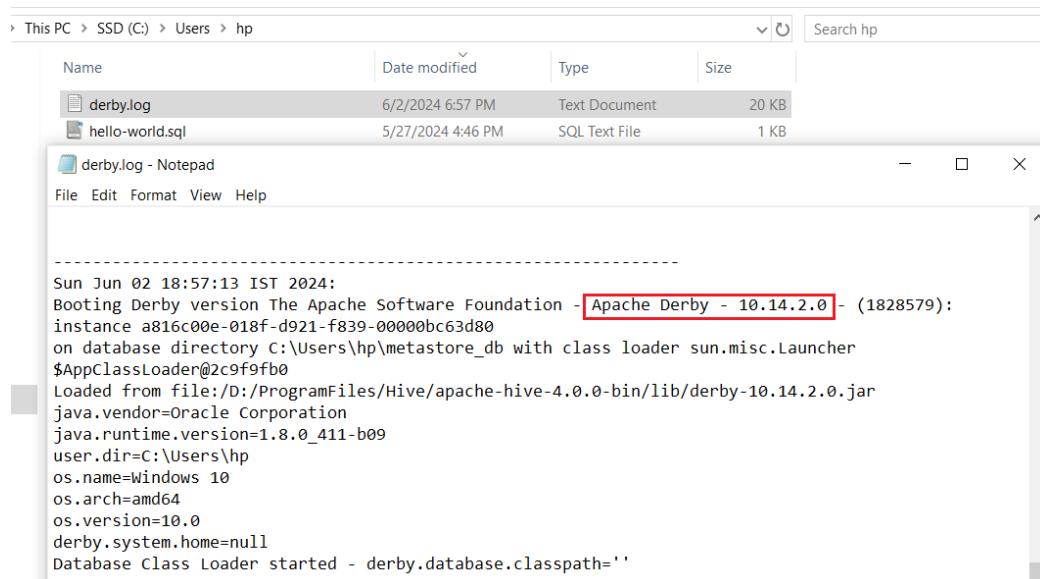
After executing the above command, it creates `metastore_db` folder, `beeline` folder and `derby.log` file in the location from where ever `schematool` utility got executed.

Name	Date modified	Type	Size
derby.log	6/2/2024 6:57 PM	Text Document	20 KB
hello-world.sql	5/27/2024 4:46 PM	SQL Text File	1 KB
beeline	6/2/2024 6:57 PM	File folder	
metastore_db	6/2/2024 6:57 PM	File folder	
Downloads	6/2/2024 6:55 PM	File folder	

- `metastore_db` folder contains the database files of Hive metastore.

Name	Date modified	Type	Size
log	6/2/2024 6:54 PM	File folder	
seg0	6/2/2024 6:57 PM	File folder	
tmp	6/2/2024 6:57 PM	File folder	
db.lck	6/2/2024 6:57 PM	LCK File	1 KB
README_DO_NOT_TOUCH_FILES.txt	6/2/2024 6:54 PM	Text Document	1 KB
service.properties	6/2/2024 6:54 PM	PROPERTIES File	1 KB

- Open `derby.log` file to see that Hive has booted Apache Derby database of `10.14.2.0` version.



```
Sun Jun 02 18:57:13 IST 2024:  
Booting Derby version The Apache Software Foundation - Apache Derby - 10.14.2.0 - (1828579):  
instance a816c00e-018f-d921-f839-00000bc63d80  
on database directory C:\Users\hp\metastore_db with class loader sun.misc.Launcher  
ClassLoader@2c9fffb0  
Loaded from file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/derby-10.14.2.0.jar  
java.vendor=Oracle Corporation  
java.runtime.version=1.8.0_411-b09  
user.dir=C:\Users\hp  
os.name=Windows 10  
os.arch=amd64  
os.version=10.0  
derby.system.home=null  
Database Class Loader started - derby.database.classpath=''
```

## 7.2. Start Beeline CLI:

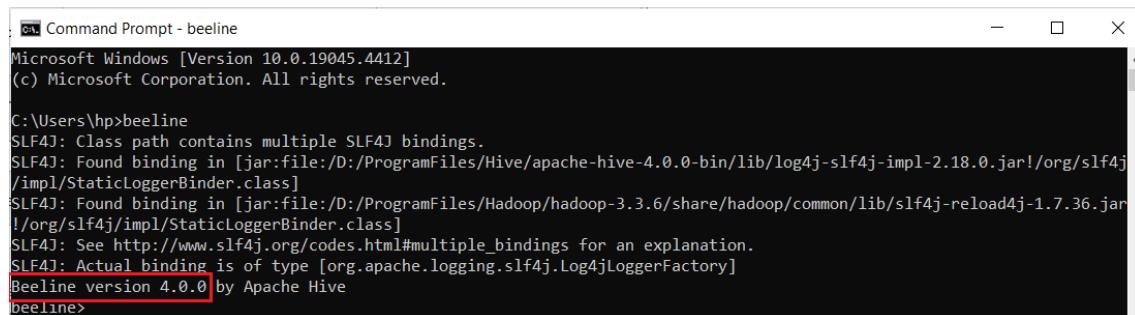
**Beeline** is the advanced version of Command Line Interface that comes with HiveServer2 service. Beeline is a JDBC client that is based on SQLLine CLI.

From Hive 4.x version, Hive CLI has been deprecated and Beeline is used instead.

Beeline works in both embedded mode and remote mode. In embedded mode, Beeline connects to an embedded HiveServer2 service and executes Hive commands.

To start Beeline, open command prompt and run this command.

```
beeline
```

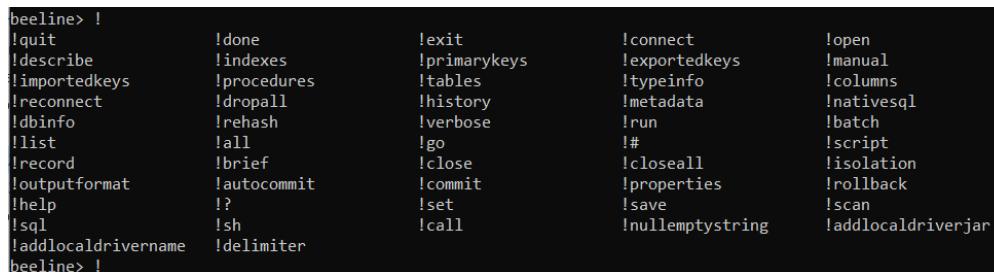


```
.\ Command Prompt - beeline
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>beeline
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Beeline version 4.0.0 by Apache Hive
beeline>
```

After executing the above command, it prints the **Beeline version 4.0.0 by Apache Hive** and provides us `beeline>` prompt.

To get a list of Beeline commands, type `!` on `beeline>` prompt and press Tab key. Go through [this Apache Hive documentation](#) to get more understanding on these commands.



```
beeline> !
!quit          !done           !exit          !connect        !open
!describe      !indexes        !primarykeys   !exportedkeys  !manual
!importedkeys  !procedures     !tables         !typeinfo      !columns
!reconnect     !dropall        !history       !metadata      !nativesql
!dbinfo        !rehash          !verbose       !run           !batch
!list          !all             !go             !#              !script
!record        !brief          !close          !closeall      !isolation
!outputformat  !autocommit    !commit         !properties    !rollback
!help          !?               !set            !save          !scan
!sql           !sh              !call           !nullemptystring !addlocaldriverjar
!addlocaldrivername !delimiter
beeline> !
```

To connect to embedded HiveServer2, enter the following command on `beeline>` prompt.

```
!connect jdbc:hive2://
```

It asks for username and password to connect. Enter the default username `scott` and password `tiger` which are provided by HiveServer2.

```
beeline> !connect jdbc:hive2://
Connecting to jdbc:hive2://
Enter username for jdbc:hive2://: scott
Enter password for jdbc:hive2://: *****
Hive Session ID = 89397bc8-4899-4e2d-bb70-d4ff609b02da
24/06/02 19:08:00 [main]: WARN hikari.HikariConfig: objectstore - leakDetectionThreshold is less than 2000ms or more than maxLifetime, disabling it.
24/06/02 19:08:05 [main]: WARN hikari.HikariConfig: objectstore-secondary - leakDetectionThreshold is less than 2000ms or more than maxLifetime, disabling it.
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 19:08:08 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.cpc.UnionSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 19:08:15 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.hll.UnionSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 19:08:15 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.IntersectsSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 19:08:15 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.EstimateSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 19:08:15 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.ExcludeSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 19:08:15 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.UnionSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 19:08:15 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.tuple.ArrayOfDoublesSketchToValuesUDTF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 19:08:16 [main]: WARN session.SessionState: Configuration hive.reloadable.aux.jars.path not specified
Connected to: Apache Hive (version 4.0.0)
Driver: Hive JDBC (version 4.0.0)
Transaction isolation: TRANSACTION_REPEATABLE_READ
@: jdbc:hive2://
```

We can also provide HiveServer2 credentials in the beeline connect command itself with this command

```
!connect jdbc:hive2:// -n scott -p tiger
or
!connect jdbc:hive2:// scott tiger
```

To start beeline connecting to HiveServer2 directly, use the following command:

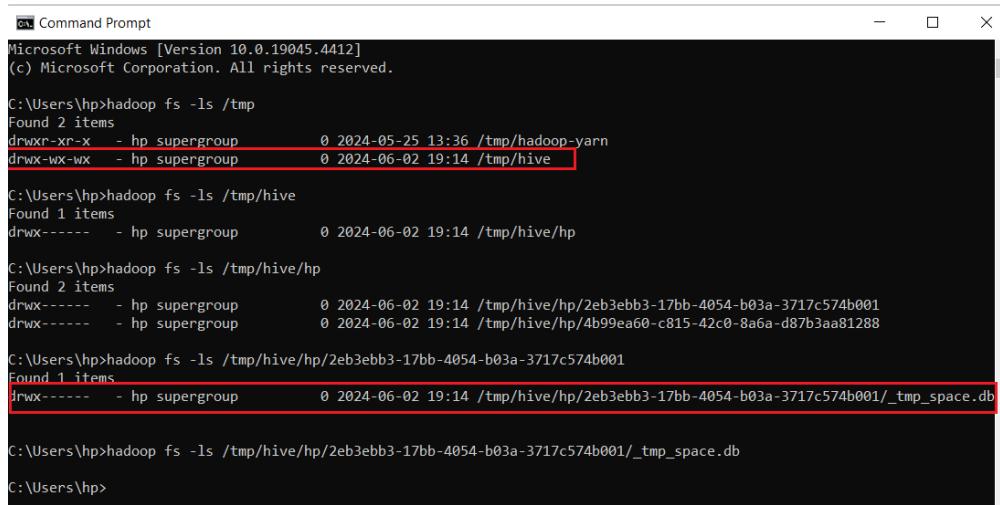
```
beeline -u jdbc:hive2:// -n scott -p tiger
or
beeline -u jdbc:hive2:// scott tiger
```

As soon as Beeline is connected, it creates a /tmp/hive folder on Hadoop File System.

Open another command prompt and execute this command to see /tmp/hive folder on HDFS

```
hadoop fs -ls /tmp
```

Hive uses /tmp HDFS directory for storing temp\_space database which is available at /tmp/hive/<userid>/<temp\_folder>/\_temp\_space.db



```
C:\ Command Prompt
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>hadoop fs -ls /tmp
Found 2 items
drwxr-xr-x - hp supergroup          0 2024-05-25 13:36 /tmp/hadoop-yarn
drwxrwxr-x - hp supergroup          0 2024-06-02 19:14 /tmp/hive

C:\Users\hp>hadoop fs -ls /tmp/hive
Found 1 items
drwx----- - hp supergroup          0 2024-06-02 19:14 /tmp/hive/hp

C:\Users\hp>hadoop fs -ls /tmp/hive/hp
Found 2 items
drwx----- - hp supergroup          0 2024-06-02 19:14 /tmp/hive/hp/2eb3ebb3-17bb-4054-b03a-3717c574b001
drwx----- - hp supergroup          0 2024-06-02 19:14 /tmp/hive/hp/4b99ea60-c815-42c0-8a6a-d87b3aa81288

C:\Users\hp>hadoop fs -ls /tmp/hive/hp/2eb3ebb3-17bb-4054-b03a-3717c574b001
Found 1 items
drwx----- - hp supergroup          0 2024-06-02 19:14 /tmp/hive/hp/2eb3ebb3-17bb-4054-b03a-3717c574b001/_temp_space.db

C:\Users\hp>hadoop fs -ls /tmp/hive/hp/2eb3ebb3-17bb-4054-b03a-3717c574b001/_temp_space.db
C:\Users\hp>
```

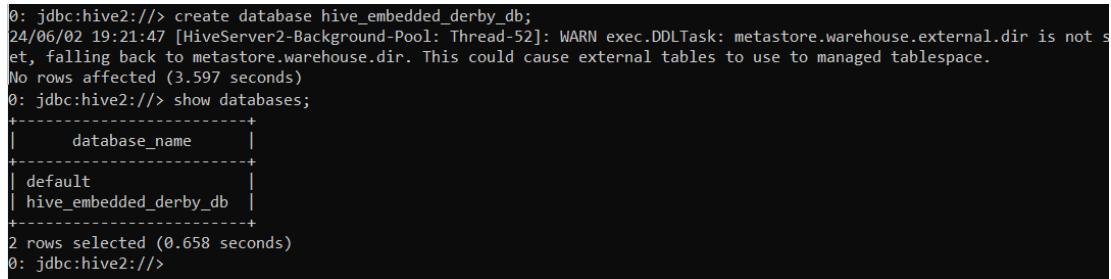
To come out of beeline> shell, use !quit command.

### 7.3. Run Queries on Beeline CLI:

Open Beeline CLI, connect to HiveServer2 and run the following queries to create a hive metadata database, create table, load data and select data.

- Create a database in Hive metastore:

```
create database hive_embedded_derby_db;
show databases;
```

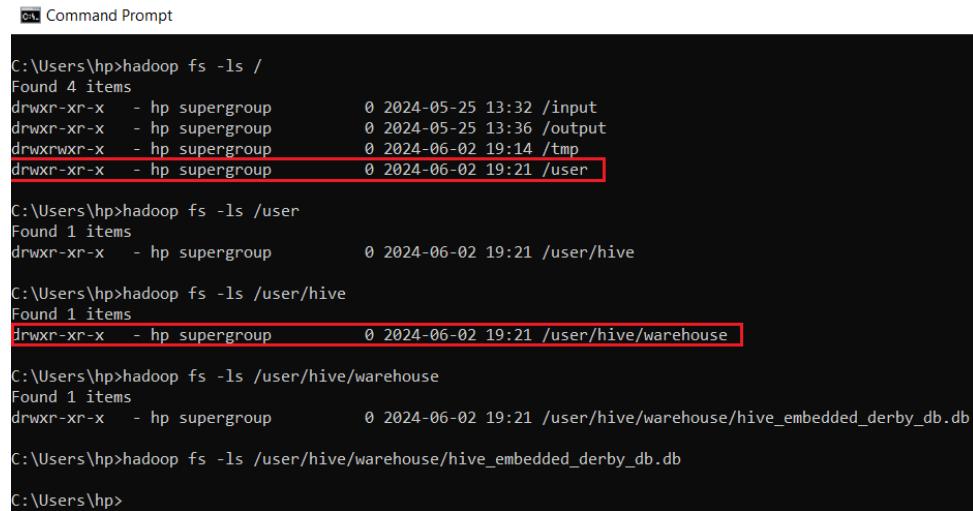


```
0: jdbc:hive2://> create database hive_embedded_derby_db;
24/06/02 19:21:47 [HiveServer2-Background-Pool: Thread-52]: WARN exec.DDLTask: metastore.warehouse.external.dir is not set, falling back to metastore.warehouse.dir. This could cause external tables to use to managed tablespace.
No rows affected (3.597 seconds)
0: jdbc:hive2://> show databases;
+-----+
| database_name |
+-----+
| default       |
| hive_embedded_derby_db |
+-----+
2 rows selected (0.658 seconds)
0: jdbc:hive2://>
```

As soon as the above query is executed, Hive creates the default warehouse directory /user/hive/warehouse in which creates a hive\_embedded\_derby\_db.db folder in Hadoop File System.

Run this command in another command prompt to validate in HDFS:

```
hadoop fs -ls /user/hive/warehouse
```



```
C:\Users\hp>hadoop fs -ls /
Found 4 items
drwxr-xr-x - hp supergroup          0 2024-05-25 13:32 /input
drwxr-xr-x - hp supergroup          0 2024-05-25 13:36 /output
drwxrwxr-x - hp supergroup          0 2024-06-02 19:14 /tmp
drwxr-xr-x - hp supergroup          0 2024-06-02 19:21 /user

C:\Users\hp>hadoop fs -ls /user
Found 1 items
drwxr-xr-x - hp supergroup          0 2024-06-02 19:21 /user/hive

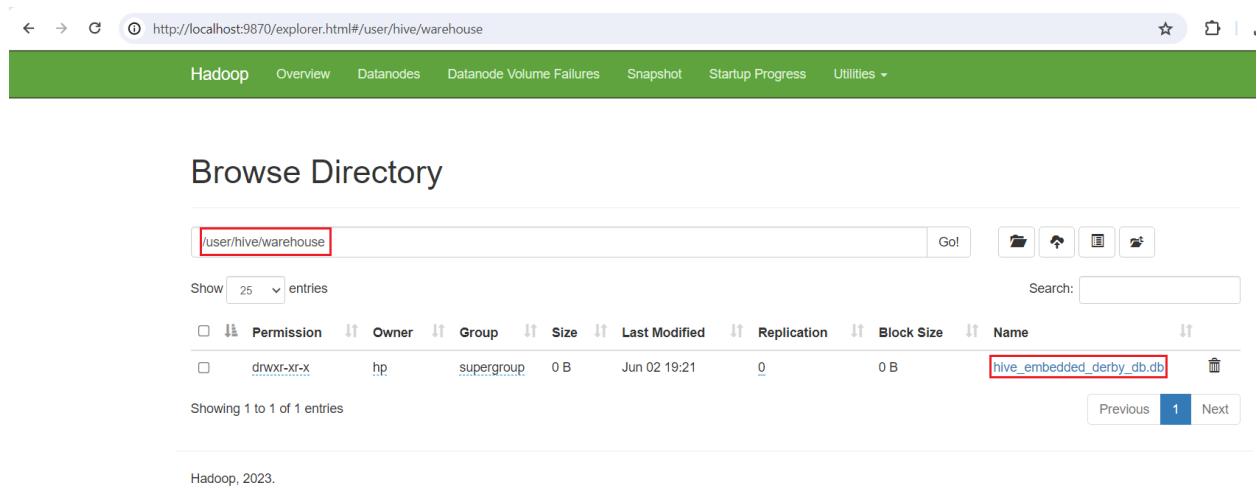
C:\Users\hp>hadoop fs -ls /user/hive
Found 1 items
drwxr-xr-x - hp supergroup          0 2024-06-02 19:21 /user/hive/warehouse

C:\Users\hp>hadoop fs -ls /user/hive/warehouse
Found 1 items
drwxr-xr-x - hp supergroup          0 2024-06-02 19:21 /user/hive/warehouse/hive_embedded_derby_db.db

C:\Users\hp>
```

We can verify the same in NameNode UI: <http://localhost:9870/dfshealth.html>

Open NameNode UI, go to **Utilities** tab and select **Browse the file system** option. Enter the directory name `/user/hive/warehouse` and you can see `hive_embedded_derby_db.db` folder available.



http://localhost:9870/explorer.html#/user/hive/warehouse

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities ▾

### Browse Directory

/user/hive/warehouse Go!

Show 25 entries Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hp	supergroup	0 B	Jun 02 19:21	0	0 B	hive_embedded_derby_db.db

Showing 1 to 1 of 1 entries Previous 1 Next

Hadoop, 2023.

- Create a table in Hive metastore:

```
use hive_embedded_derby_db;
create table employees(emp_id int, emp_name string, emp_salary int);
show tables;
```

```

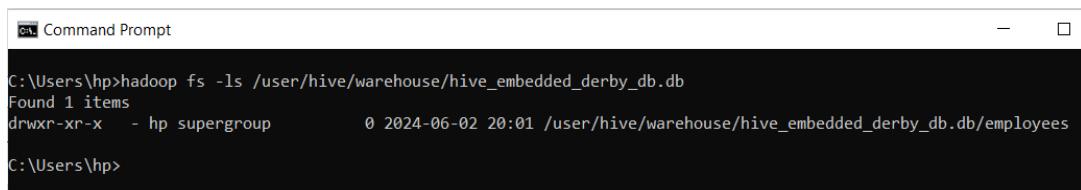
0: jdbc:hive2://> use hive_embedded_derby_db;
No rows affected (0.03 seconds)
0: jdbc:hive2://> create table employees(emp_id int, emp_name string, emp_salary int);
No rows affected (1.436 seconds)
0: jdbc:hive2://> show tables;
+-----+
| tab_name |
+-----+
| employees |
+-----+
1 row selected (0.282 seconds)
0: jdbc:hive2://>

```

We can see the above table is saved under  
`/user/hive/warehouse/hive_embedded_derby_db.db` HDFS location.

Run this command in another command prompt to validate in HDFS:

```
hadoop fs -ls /user/hive/warehouse/hive_embedded_derby_db.db
```



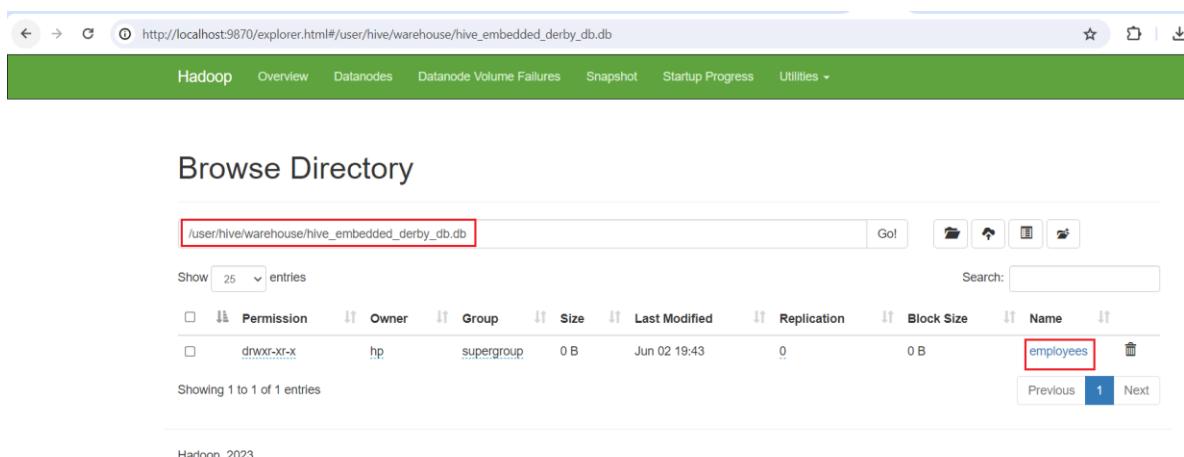
```

C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_embedded_derby_db.db
Found 1 items
drwxr-xr-x  - hp supergroup          0 2024-06-02 20:01 /user/hive/warehouse/hive_embedded_derby_db.db/employees

C:\Users\hp>

```

We can verify the same in NameNode UI. Click on `hive_embedded_derby_db.db` folder in `/user/hive/warehouse` directory and you can see `employees` folder available.



Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hp	supergroup	0 B	Jun 02 19:43	0	0 B	employees

- Insert data into the table:

```
insert into employees values (101, 'johnson',5000);
```

The above insert command submits the MapReduce job to get the record into table.

```
0: jdbc:hive2://> insert into employees values (101, 'johnson',5000);
24/06/02 20:00:08 [HiveServer2-Background-Pool: Thread-104]: WARN ql.Driver: Hive-on-MR is deprecated in Hive 2 and may
not be available in the future versions. Consider using a different execution engine (i.e. tez) or using Hive 1.X releases.
Query ID = hp_20240602200001_c8645d47-1b13-47a9-93cc-99b8d76d9f53
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
WARN : Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different e
xecution engine (i.e. tez) or using Hive 1.X releases.
24/06/02 20:00:11 [HiveServer2-Background-Pool: Thread-104]: WARN mapreduce.JobResourceUploader: Hadoop command-line opt
ion parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
Starting Job = job_1717334152951_0001, Tracking URL = http://DESKTOP-KGH2E2G:8088/proxy/application_1717334152951_0001/
Kill Command = D:\ProgramFiles\Hadoop\hadoop-3.3.6\bin\mapred job -kill job_1717334152951_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
24/06/02 20:00:34 [HiveServer2-Background-Pool: Thread-104]: WARN mapreduce.Counters: Group org.apache.hadoop.mapred.Tas
k$Counter is deprecated. Use org.apache.hadoop.mapreduce.TaskCounter instead
2024-06-02 20:00:34,557 Stage-1 map = 0%, reduce = 0%
2024-06-02 20:00:49,499 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 9.404 sec
2024-06-02 20:01:05,364 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 19.62 sec
MapReduce Total cumulative CPU time: 19 seconds 620 msec
Ended Job = job_1717334152951_0001
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://localhost:9820/user/hive/warehouse/hive_embedded_derby_db.db/employees/.hive-staging_hiv
e_2024-06-02_20-00-01_482_1593368538904413760-1/-ext-10000
Loading data to table hive_embedded_derby_db.employees
24/06/02 20:01:08 [HiveServer2-Background-Pool: Thread-104]: WARN metadata.Hive: Cannot get a table snapshot for employees
24/06/02 20:01:08 [HiveServer2-Background-Pool: Thread-104]: WARN metadata.Hive: Cannot get a table snapshot for employees
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 19.62 sec HDFS Read: 24647 HDFS Write: 306 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 19 seconds 620 msec
1 row affected (67.214 seconds)
```

We can track the job status on YARN UI: <http://localhost:8088/cluster>

In YARN UI, you can see an application name with `insert into employees...` that was executed.

ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime
application_1717334152951_0001	hp	Insert into employees....., Johnson',5000) (Stage-1)	MAPREDUCE	hp_20240602200001_c8645d47-1b13-47a9-93cc-99b8d76d9f53,userId=null	default	0	Sun Jun 2 20:00:14 +0550 2024	Sun Jun 2 20:00:16 +0550 2024	Sun Jun 2 20:01:05 +0550 2024

Click on the application ID to see the additional job details and logs.

The screenshot shows the Hadoop Application Overview page for application ID **application\_1717334152951\_0001**. The left sidebar has sections for Cluster (About, Nodes, Node Labels, Applications: NEW, NEW\_SAVING, SUBMITTED, ACCEPTED, RUNNING, FINISHED, FAILED, KILLED), Scheduler, and Tools. The main content area displays the following details:

User: hp	Name: insert into employees....., 'johnson',5000) (Stage-1)
Application Type: MAPREDUCE	
Application Tags: hp_20240602200001_c8645d47-1b13-47a9-93cc-99b8d76d9f53,userid=null	
Application Priority: 0 (Higher Integer value indicates higher priority)	
YarnApplicationState: FINISHED	
Queue: default	
FinalStatus Reported by AM: SUCCEEDED	
Started: Sun Jun 02 20:00:14 +0530 2024	
Launched: Sun Jun 02 20:00:16 +0530 2024	
Finished: Sun Jun 02 20:01:05 +0530 2024	
Elapsed: 51sec	
Tracking URL: History	
Log Aggregation Status: DISABLED	
Application Timeout (Remaining Time): Unlimited	
Diagnostics:	
Unmanaged Application: false	
Application Node Label expression: <Not set>	
AM container Node Label expression: <DEFAULT_PARTITION>	

Below this is the Application Metrics section:

Total Resource Preempted: <memory:0, vCores:0>
Total Number of Non-AM Containers Preempted: 0
Total Number of AM Containers Preempted: 0
Resource Preempted from Current Attempt: <memory:0, vCores:0>
Number of Non-AM Containers Preempted from Current Attempt: 0
Aggregate Resource Allocation: 143901 MB-seconds, 82 vcore-seconds
Aggregate Preempted Resource Allocation: 0 MB-seconds, 0 vcore-seconds

At the bottom is a table of logs:

Attempt ID	Started	Node	Logs	Nodes blacklisted by the app	Nodes blacklisted by the system
appattempt_1717334152951_0001_000001	Sun Jun 2 20:00:14 +0530 2024	http://192.168.56.1:8042	Logs	0	0

Showing 1 to 1 of 1 entries

After the above job is succeeded, we can see a file created under `/user/hive/warehouse/hive_embedded_derby_db.db/employees` HDFS location.

Open another Command Prompt and run this command:

```
hadoop fs -ls /user/hive/warehouse/hive_embedded_derby_db.db/employees
```

```
C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_embedded_derby_db.db/employees
Found 1 items
-rw-r--r-- 1 hp supergroup          17 2024-06-02 20:00 /user/hive/warehouse/hive_embedded_derby_db.db/employees/000000_0
C:\Users\hp>
```

In NameNode UI, click on `employees` folder in `/user/hive/warehouse/hive_embedded_derby_db.db` directory and you can see a file `000000_0` available.

Browse Directory

/user/hive/warehouse/hive\_embedded\_derby\_db.db/employees

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	hp	supergroup	17 B	Jun 02 20:00	1	128 MB	000000_0

Showing 1 to 1 of 1 entries

Search:

Go!

Show 25 entries

Hadoop, 2023.

- Select data from the Hive table:

```
select * from employees;
```

```
0: jdbc:hive2://> select * from employees;
24/06/02 20:17:45 [2eb3ebb3-17bb-4054-b03a-3717c574b001 main]: WARN optimizer.SimpleFetchOptimizer: Table hive_embedded_derby_db@employees is external table, falling back to filesystem scan.
+-----+-----+-----+
| employees.emp_id | employees.emp_name | employees.emp_salary |
+-----+-----+-----+
| 101             | johnson          | 5000                |
+-----+-----+-----+
1 row selected (0.561 seconds)
```

We can verify the above table output in HDFS using the following command

```
hadoop fs -cat /user/hive/warehouse/hive_embedded_derby_db.db/employees/000000_0
```

```
C:\ Command Prompt
C:\Users\hp>hadoop fs -cat /user/hive/warehouse/hive_embedded_derby_db.db/employees/000000_0
101@johnson@5000
C:\Users\hp>
```

In NameNode UI, click on **000000\_0** file and select **Head the file** or **Tail the file** to see the file contents. We can download this file by clicking on **Download** option.

The screenshot shows the Apache Hadoop File Browser interface. A modal window titled "File information - 000000\_0" is open. At the top of the modal, there are three buttons: "Download", "Head the file (first 32K)" (which is highlighted with a red box), and "Tail the file (last 32K)". Below this, a "Block information" section displays details about the file's first block: Block ID (1073742054), Block Pool ID (BP-1739737951-192.168.1.5-1716620109937), Generation Stamp (1230), Size (17), and Availability (192.168.56.1). The "File contents" section shows the first 32KB of the file, containing the single row: 101|johnson|5000.

### Note:

In embedded Hive metastore mode, we cannot start a second Beeline session while the current one is still active because embedded metastore allows only one connection at any time.

To test this, open another Command Prompt, start Beeline and connect to embedded HiveServer2 using the following command

```
beeline -u jdbc:hive2:// scott tiger
```

```
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>beeline -u jdbc:hive2:// scott tiger
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://
Hive Session ID = 9d9d7345-817d-4ad4-940e-a14f91750e99
24/06/02 20:21:04 [main]: WARN hikari.HikariConfig: objectstore - leakDetectionThreshold is less than 2000ms or more than maxLifetime, disabling it.
24/06/02 20:21:05 [main]: ERROR pool.HikariPool: objectstore - Exception during pool initialization.
java.sql.SQLException: Failed to start database 'metastore_db' with class loader sun.misc.Launcher$AppClassLoader@2c9f9fb0, see the next exception for details.
        at org.apache.derby.impl.jdbc.SQLExceptionFactory.getSQLException(Unknown Source) ~[derby-10.14.2.0.jar:?:?]
        at org.apache.derby.impl.jdbc.SQLExceptionFactory.getSQLException(Unknown Source) ~[derby-10.14.2.0.jar:?:?]
        at org.apache.derby.impl.jdbc.Util.getNextException(Unknown Source) ~[derby-10.14.2.0.jar:?:?]
        at org.apache.derby.impl.jdbc.EmbedConnection.bootDatabase(Unknown Source) ~[derby-10.14.2.0.jar:?:?]
        at org.apache.derby.impl.jdbc.EmbedConnection.<init>(Unknown Source) ~[derby-10.14.2.0.jar:?:?]
```

```

        at org.apache.derby.jdbc.InternalDriver$LoginCallable.call(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at java.util.concurrent.FutureTask.run(FutureTask.java:266) ~[:1.8.0_411]
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:1149) ~[:1.8.0_411]
        at java.lang.Thread.run(Thread.java:750) ~[:1.8.0_411]
Caused by: org.apache.derby.iapi.error.StandardException: Failed to start database 'metastore_db' with class loader sun.misc.Launcher$AppClassLoader@2c9f9fb0, see the next exception for details.
        at org.apache.derby.iapi.error.StandardException.newException(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at org.apache.derby.impl.jdbc.SQLExceptionFactory.wrapArgsForTransportAcrossDRDA(Unknown Source) ~[derby-10.14.2.0.jar:?]
        ... 15 more
Caused by: org.apache.derby.iapi.error.StandardException: Another instance of Derby may have already booted the database
C:\Users\hp\metastore_db.
        at org.apache.derby.iapi.error.StandardException.newException(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at org.apache.derby.iapi.error.StandardException.newException(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at org.apache.derby.impl.store.raw.data.BaseDataFileFactory.privGetJBMSLockOnDB(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at org.apache.derby.impl.store.raw.data.BaseDataFileFactory.run(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at java.security.AccessController.doPrivileged(Native Method) ~[:1.8.0_411]
        at org.apache.derby.impl.store.raw.data.BaseDataFileFactory.getJBMSLockOnDB(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at org.apache.derby.impl.store.raw.data.BaseDataFileFactory.boot(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at org.apache.derby.impl.services.monitor.BaseMonitor.boot(Unknown Source) ~[derby-10.14.2.0.jar:?]
        at org.apache.derby.impl.services.monitor.TopService.bootModule(Unknown Source) ~[derby-10.14.2.0.jar:?]

```

You can see that it has thrown errors “Failed to start database ‘metastore\_db’. Another instance of Derby may have already booted the database” because another Beeline session is active. Until the previous session is closed, we cannot start another Beeline session with Embedded metastore.

Close or exit out of all Beeline CLI and make sure that no sessions are active.

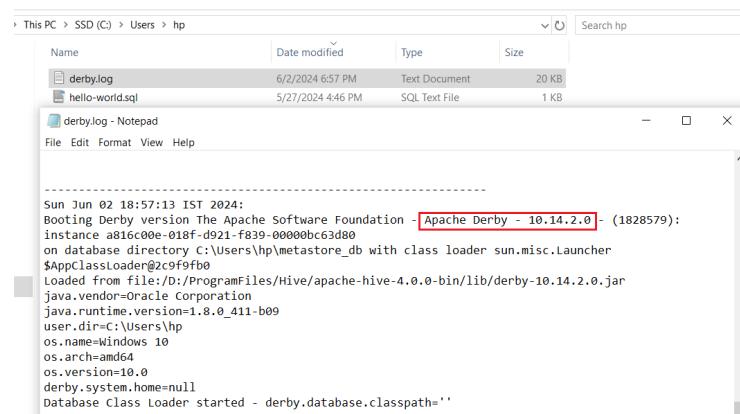
## 8. Configure Local Derby Metastore:

We can configure Hive metastore in local mode with Derby database by making Derby to run in network mode. For this, we should first download and install Apache Derby.

### 8.1. Install Apache Derby:

It is recommended to install Derby database of version that is shown for Embedded Derby installed with Hive.

Open `derby.log` file available in the location where `metastore_db` is created to see the embedded Derby database version.



The screenshot shows a Windows File Explorer window with the path `This PC > SSD (C) > Users > hp`. Inside the `hp` folder, there are two files: `derby.log` (Text Document, 20 KB) and `hello-world.sql` (SQL Text File, 1 KB). Below the File Explorer is a Notepad window titled `derby.log - Notepad`. The Notepad content displays the following log output:

```

-----
Sun Jun 02 18:57:13 IST 2024:
Booting Derby version The Apache Software Foundation - Apache Derby - 10.14.2.0 - (1828579):
instance ab16c00e-01bf-d921-f839-00000b0bc3d80
on database directory C:\Users\hp\metastore_db with class loader sun.misc.Launcher
$AppClassLoader@2c9f9fb0
Loaded from file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/derby-10.14.2.0.jar
java.vendor=Oracle Corporation
java.runtime.version=1.8.0_411-b09
user.dir=C:\Users\hp
os.name=Windows 10
os.arch=amd64
os.version=10.0
derby.system.home=null
Database Class Loader started - derby.database.classpath=''

```

Additionally, we can check the appropriate Derby version in [Apache Derby Downloads](#) page for supported versions of Java releases. Since we have Java 8 to run Hadoop and Hive, the latest Derby release for Java 8 is **Apache Derby 10.14.2.0** version.

The screenshot shows the Apache Derby Downloads page at [https://db.apache.org/derby/derby\\_downloads.html](https://db.apache.org/derby/derby_downloads.html). The page header includes the Apache Derby logo and the Apache DB Project logo. The main navigation menu has 'Download' selected. Under 'Download', 'Overview' is selected. The page title is 'Apache Derby: Downloads'. A sidebar on the left lists links like 'Search the site with goog' and 'Search'. The main content area lists supported Java versions with their corresponding Derby releases:

- For Java 21 and Higher (releases which handle the deprecation and removal of old Java apis)**: 10.17.1.0 (November 10, 2023 / SVN 1913217)
- For Java 17 and Higher (releases which no longer support the Java SecurityManager)**: 10.16.1.1 (May 19, 2022 / SVN 1901046)
- For Java 9 and Higher (jar files converted into JPMS modules)**: 10.15.2.0 (February 18, 2020 / SVN 1873585), 10.15.1.3 (March 5, 2019 / SVN 1853019)
- For Java 8 and Higher (releases which support lambda expressions)**: 10.14.2.0 (May 3, 2018 / SVN 1828579), 10.13.1.1 (October 25, 2016 / SVN 1766613)
- For Java 6 and Higher (releases which support JDBC 4.0 and driver autoloading)**

Download db-derby-10.14.2.0-bin.tar.gz file from the [Apache Derby 10.14.2.0 Release](#) website which gets downloaded to your **Downloads** folder.

The screenshot shows the Apache Derby 10.14.2.0 Release page at [https://db.apache.org/derby/releases/release-10\\_14\\_2\\_0.html](https://db.apache.org/derby/releases/release-10_14_2_0.html). The page header includes the Apache Derby logo and the Apache DB Project logo. The main navigation menu has 'Download' selected. Under 'Download', 'Release Notes for Apache Derby 10.14.2.0' is selected. The page title is 'Apache Derby 10.14.2.0 Release'. The left sidebar lists links for the Apache Software Foundation, Documentation, Blogs and Articles, Integration With Other Products, Eclipse Plug-ins, Papers and Presentations, and The Apache Software Foundation. The main content area has a section titled 'Distributions' with a sub-section 'Release Notes for Apache Derby 10.14.2.0'. It provides instructions to verify the integrity of distribution files and lists four distribution types:

- bin distribution - contains the documentation, Javadoc, and Jar files for Derby.
- lib distribution - contains only the Jar files for Derby.
- lib-debug distribution - contains Jar files for Derby with source line numbers.
- src distribution - contains the Derby source tree at the point which the binaries were built.

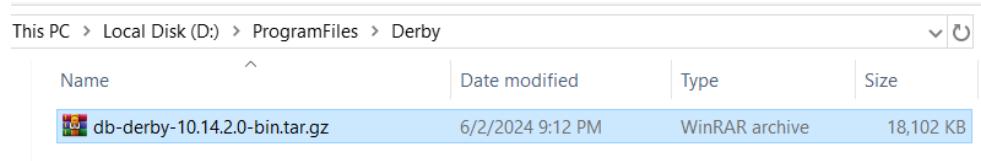
Links for each distribution type are provided with PGP and MD5 checksums:

- db-derby-10\_14\_2\_0-bin.zip [PGP] [MD5]
- db-derby-10\_14\_2\_0-bin.tar.gz [PGP] [MD5]
- db-derby-10\_14\_2\_0-lib.zip [PGP] [MD5]
- db-derby-10\_14\_2\_0-lib.tar.gz [PGP] [MD5]
- db-derby-10\_14\_2\_0-lib-debug.zip [PGP] [MD5]
- db-derby-10\_14\_2\_0-lib-debug.tar.gz [PGP] [MD5]
- db-derby-10\_14\_2\_0-src.zip [PGP] [MD5]
- db-derby-10\_14\_2\_0-src.tar.gz [PGP] [MD5]

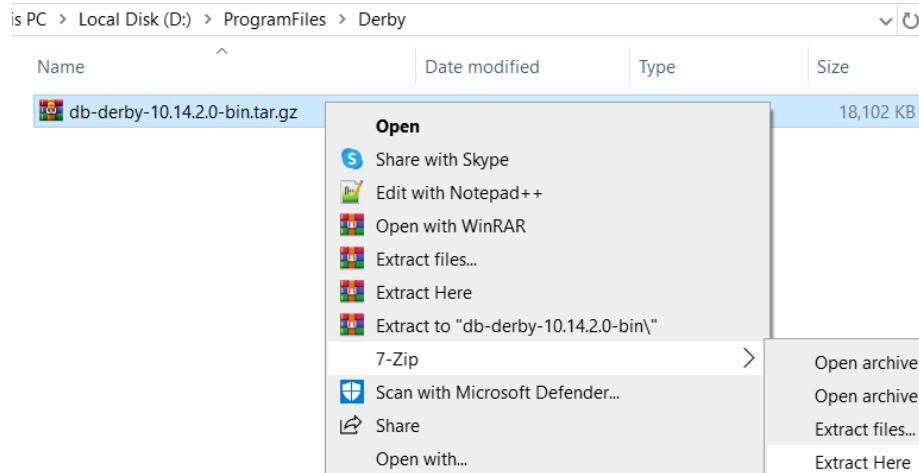
A note at the bottom states: '(Note that, due to long filenames, you will need gnu tar to unravel this tarball.)'

After the file is downloaded, unpack it using any file archiver (7zip or WinRAR) utility as below.

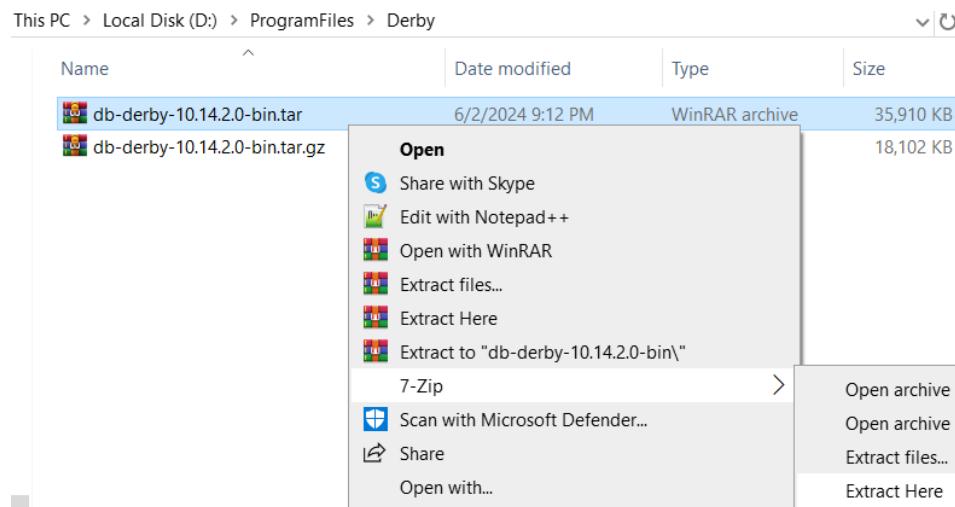
- Choose the installation directory in your machine and copy db-derby-10.14.2.0-bin.tar.gz file to that directory. Here, we are choosing Derby installation directory as D:\ProgramFiles\Derby.



- Right click on db-derby-10.14.2.0-bin.tar.gz and choose **7-Zip -> Extract Here** option which extracts a new packed file db-derby-10.14.2.0-bin.tar.



- Next, unpack db-derby-10.14.2.0-bin.tar file using 7zip utility.



- The tar file extraction may take few minutes to finish. After finishing, you see a folder named `db-derby-10.14.2.0-bin` which consists of Derby binaries and libraries.

Name	Date modified	Type	Size
bin	6/2/2024 9:15 PM	File folder	
demo	6/2/2024 9:15 PM	File folder	
docs	6/2/2024 9:15 PM	File folder	
javadoc	6/2/2024 9:15 PM	File folder	
lib	6/2/2024 9:15 PM	File folder	
test	6/2/2024 9:15 PM	File folder	
index.html	3/10/2018 10:01 PM	Chrome HTML Do...	5 KB
KEYS	4/7/2018 6:44 AM	File	46 KB
LICENSE	4/7/2018 6:44 AM	File	12 KB
NOTICE	4/7/2018 6:44 AM	File	13 KB
RELEASE-NOTES.html	4/7/2018 6:44 AM	Chrome HTML Do...	7 KB

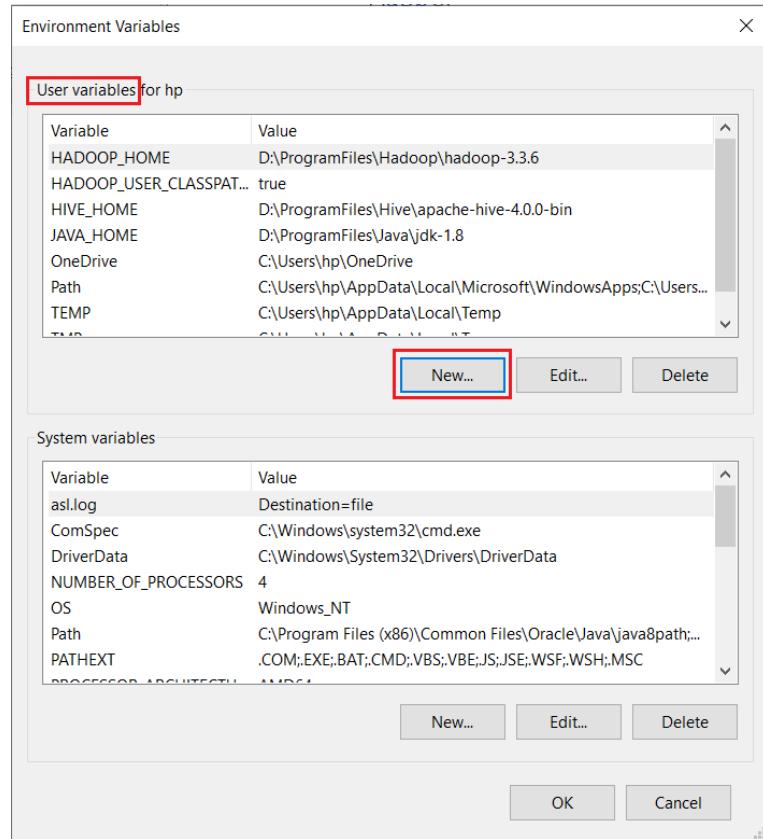
## 8.2. Set up Environment Variables:

After installing Derby, we should configure two environment variables defining Derby installation path.

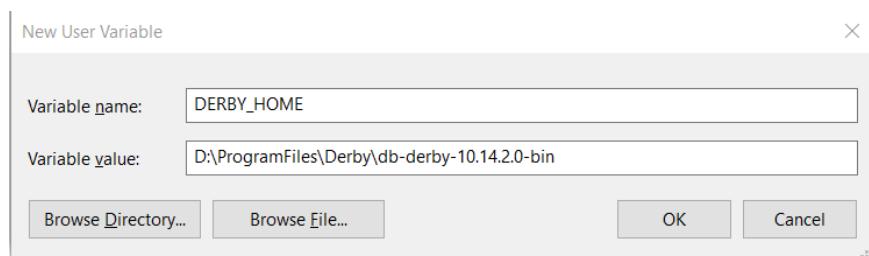
- DERBY\_HOME:** This is the Derby installation directory path in the machine (*in our case, it is D:\ProgramFiles\Derby\db-derby-10.14.2.0-bin*)
- DERBY\_OPTS:** Set it to `-Dderby.system.home=%DERBY_HOME%` location. This variable is optional but required if you want to create Hive metastore database in a custom location other than default location.

In the Windows search bar, start typing “environment variables” and select the first match which opens up **System Properties** dialog. On the **System Properties** window, press **Environment Variables** button.

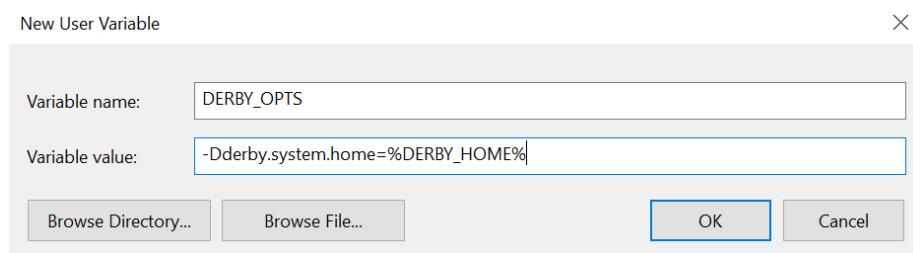
In the **Environment Variables** dialog, click on **New** under **User variables** section.



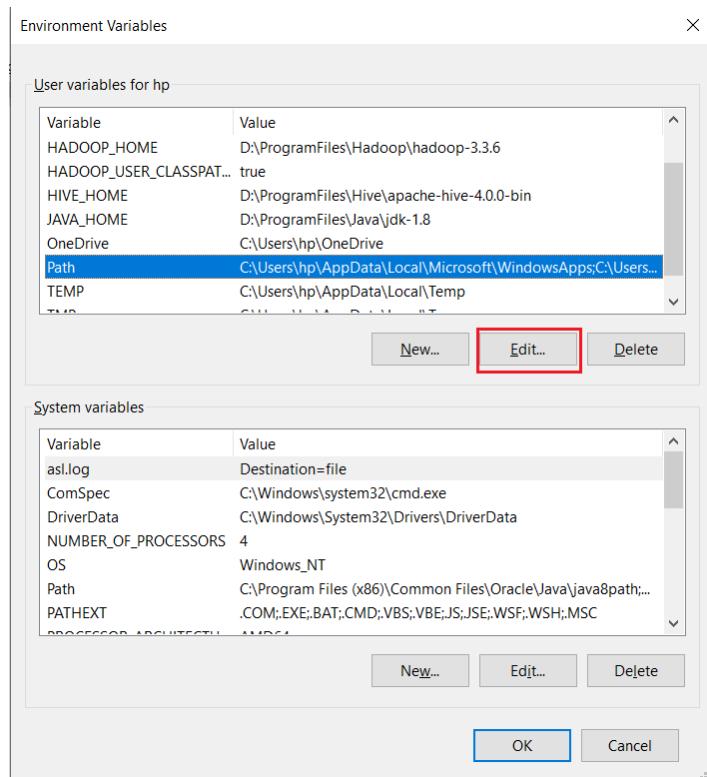
Add `DERBY_HOME` variable and press OK.



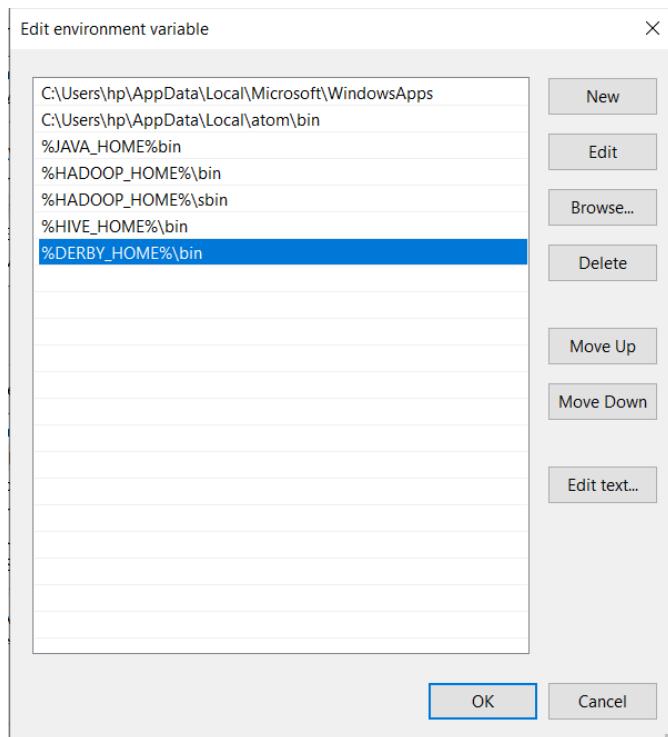
Click on **New** again and add `DERBY_OPTS` variable and press OK.



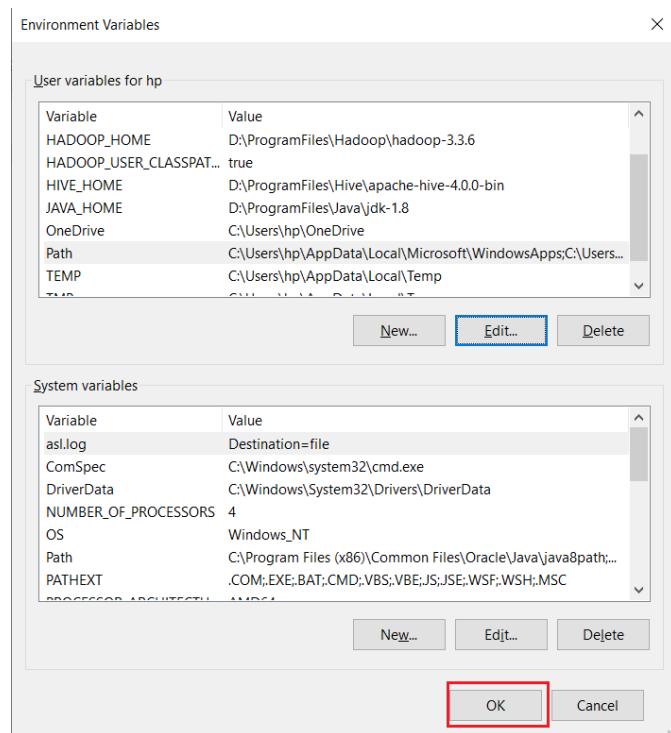
Select PATH variable and press **Edit** button



Press **New** and add **%DERBY\_HOME%\bin** value and press **OK**.



Press OK to apply environment variable changes and close window.



### 8.3. Start Derby Network Server:

Now, start the Derby network server on the local host.

Open **Command Prompt** or Windows PowerShell in **Administrator** mode and run the following command

```
startNetworkServer -h 0.0.0.0
```

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> startNetworkServer -h 0.0.0.0
Sun Jun 02 21:18:22 IST 2024 : Security manager installed using the Basic server security policy.
Sun Jun 02 21:18:23 IST 2024 : Apache Derby Network Server - 10.14.2.0 - (1828579) started and ready to accept connections on port 1527
```

Apache Derby Network server runs on **1527** port by default

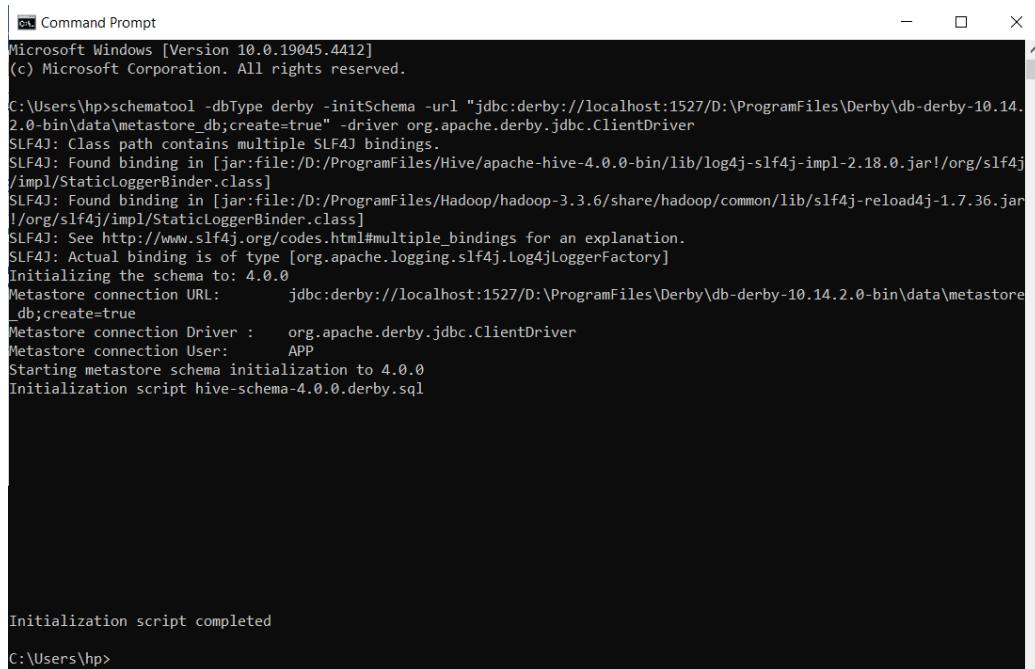
#### 8.4. Initialize Local Metastore:

Before starting Hive, we need to initialize Hive Metastore using `schematool` utility to create metastore database.

Run the following `schematool` command which connects to Derby Network Server and creates `metastore_db` in `DERBY_HOME\data` location (i.e.

`D:\ProgramFiles\Derby\db-derby-10.14.2.0-bin\data location)`

```
schematool -dbType derby -initSchema -url  
"jdbc:derby://localhost:1527/D:\ProgramFiles\Derby\db-derby-10.14.2.0-  
bin\data\metastore_db;create=true" -driver org.apache.derby.jdbc.ClientDriver
```

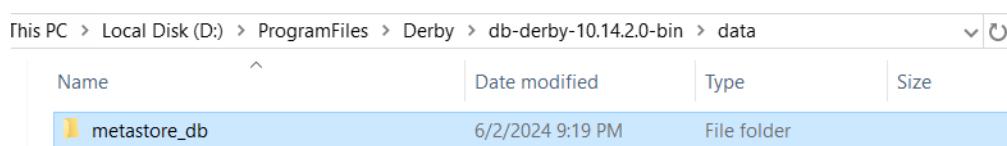


The screenshot shows a Microsoft Windows Command Prompt window titled "Command Prompt". The title bar includes standard window controls (minimize, maximize, close) and a maximize button. The window contains the following text output from the command execution:

```
Microsoft Windows [Version 10.0.19045.4412]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\hp>schematool -dbType derby -initSchema -url "jdbc:derby://localhost:1527/D:\ProgramFiles\Derby\db-derby-10.14.2.0-bin\data\metastore_db;create=true" -driver org.apache.derby.jdbc.ClientDriver  
SLF4J: Class path contains multiple SLF4J bindings.  
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]  
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]  
Initializing the schema to: 4.0.0  
Metastore connection URL: jdbc:derby://localhost:1527/D:\ProgramFiles\Derby\db-derby-10.14.2.0-bin\data\metastore_db;create=true  
Metastore connection Driver : org.apache.derby.jdbc.ClientDriver  
Metastore connection User: APP  
Starting metastore schema initialization to 4.0.0  
Initialization script hive-schema-4.0.0.derby.sql  
  
Initialization script completed  
C:\Users\hp>
```

**Note:** If you encounter Exception in thread "main" `java.lang.NoSuchMethodError: org.fusesource.jansi.AnsiConsole.wrapOutputStream(Ljava/io/OutputStream;)Ljava/io/OutputStream` error, download `jansi-1.18.jar` file from the [official Jansi Download](#) website and copy to `HIVE_HOME\lib` directory.

After executing the above command, it creates `metastore_db` folder in `DERBY_HOME\data` location.



## 8.5. Configure Hive Site:

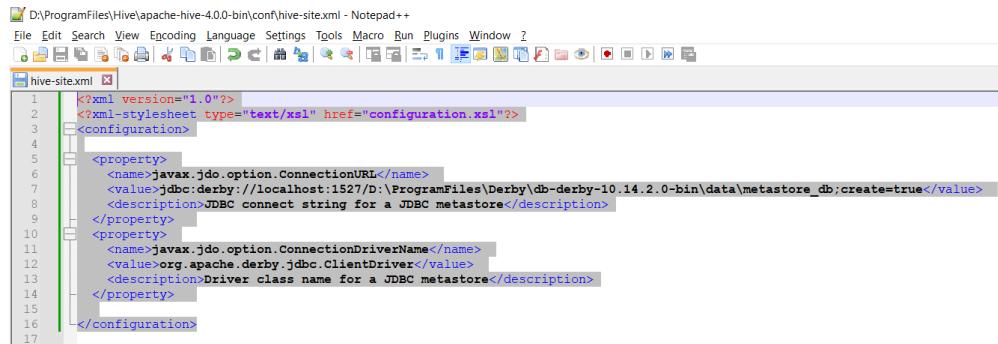
Now, add Derby Network database configuration settings in `hive-site.xml` file for Hive to connect.

Go to `%HIVE_HOME%\conf` directory, create a file named `hive-site.xml` and paste the following code in the file. This code is referring to `metastore_db` in `DERBY_HOME\data` location.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>

    <property>
        <name>javax.jdo.option.ConnectionURL</name>
        <value>jdbc:derby://localhost:1527/D:\ProgramFiles\Derby\db-derby-10.14.2.0-bin\data\metastore_db;create=true</value>
        <description>JDBC connect string for a JDBC metastore</description>
    </property>
    <property>
        <name>javax.jdo.option.ConnectionDriverName</name>
        <value>org.apache.derby.jdbc.ClientDriver</value>
        <description>Driver class name for a JDBC metastore</description>
    </property>

</configuration>
```



## 8.6. Copy Derby Libraries:

For Hive to communicate with local Derby, we need to copy `derbyclient.jar` and `derbytools.jar` files from `%DERBY_HOME%\lib` directory and paste them into `%HIVE_HOME%\lib` directory.

PC > Local Disk (D:) > ProgramFiles > Hive > apache-hive-4.0.0-bin > lib			
Name	Date modified	Type	Size
derby-10.14.2.0.jar	1/22/2020 8:40 PM	Executable Jar File	3,158 KB
derbyclient.jar	4/7/2018 6:40 AM	Executable Jar File	575 KB
derbytools.jar	4/7/2018 6:40 AM	Executable Jar File	226 KB
disruptor-3.3.7.jar	1/22/2020 8:40 PM	Executable Jar File	84 KB

## 8.7. Run Beeline CLI:

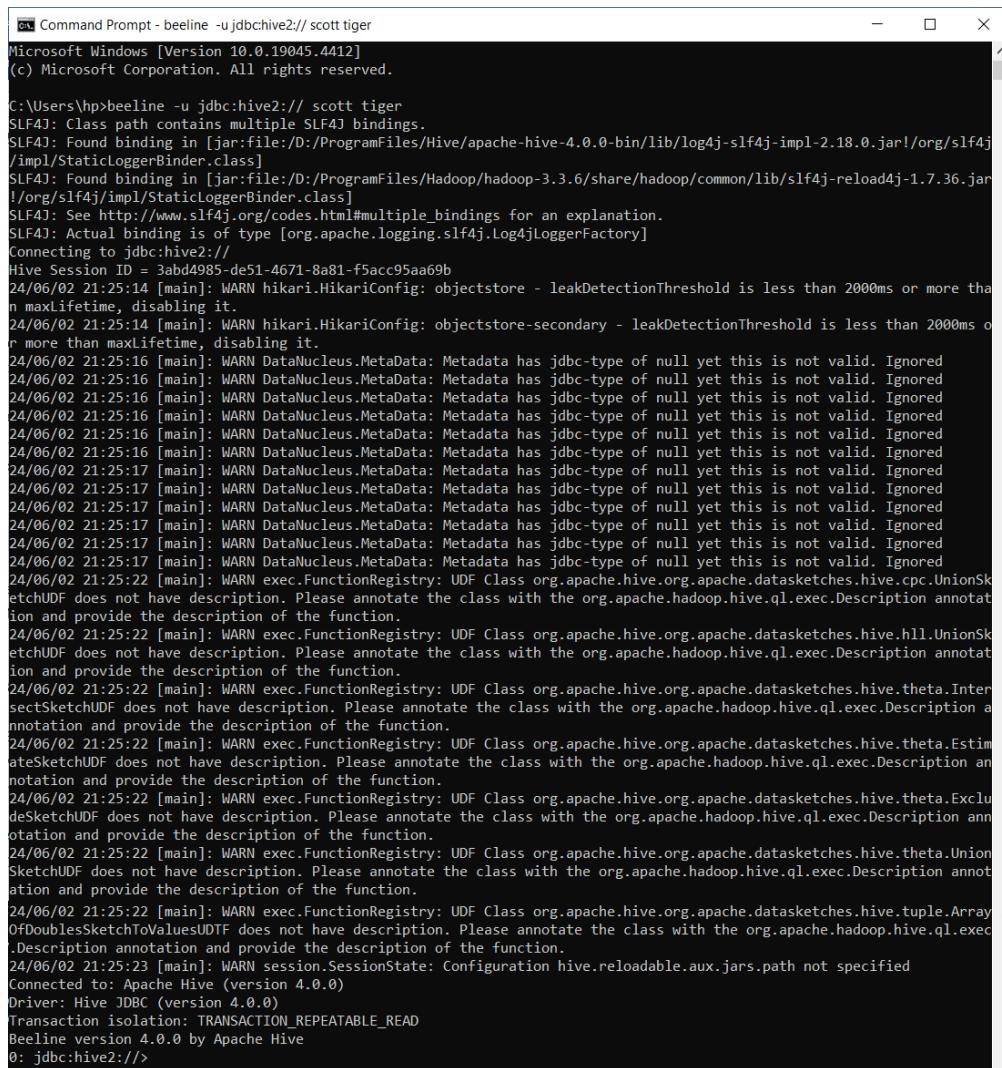
The Beeline works in embedded mode and remote mode. In embedded mode, Beeline connects to an embedded HiveServer2 and in remote mode, it connects to HiveServer2 service over Thrift. Using Beeline, we can connect to HiveServer2 running on Local or Remote server using hostname and port.

To start Beeline in embedded mode, open command prompt and run the following command.

```
beeline -u jdbc:hive2:// -n scott -p tiger
```

or

```
beeline -u jdbc:hive2:// scott tiger
```



```
C:\Users\hp\beeline -u jdbc:hive2:// scott tiger
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp\beeline -u jdbc:hive2:// scott tiger
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://
Hive Session ID = 3abd4985-de51-4671-8a81-f5acc95aa69b
24/06/02 21:25:14 [main]: WARN hikari.HikariConfig: objectstore - leakDetectionThreshold is less than 2000ms or more than maxLifetime, disabling it.
24/06/02 21:25:14 [main]: WARN hikari.HikariConfig: objectstore-secondary - leakDetectionThreshold is less than 2000ms or more than maxLifetime, disabling it.
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:16 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:25:17 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.cpc.UnionSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 21:25:22 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.hll.UnionSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 21:25:22 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.IntersectSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 21:25:22 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.EstimateSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 21:25:22 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.ExcludeSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 21:25:22 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.UnionSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 21:25:22 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.tuple.ArrayOfDoublesSketchToValuesUDTF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotation and provide the description of the function.
24/06/02 21:25:23 [main]: WARN session.SessionState: Configuration hive.reloadable.aux.jars.path not specified
Connected to: Apache Hive (version 4.0.0)
Driver: Hive JDBC (version 4.0.0)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 4.0.0 by Apache Hive
0: jdbc:hive2://>
```

**Note:** If you encounter “*org.apache.hadoop.hive.metastore.api.MetaException Version information not found in metastore*” error, then make sure `metastore_db` is initialized properly. If not done, stop the Derby Network server, delete the metastore database, reinitialize it using `schematool` utility, start Derby Network server and then start beeline.

After beeline is connected, run following queries to create database, table and insert data.

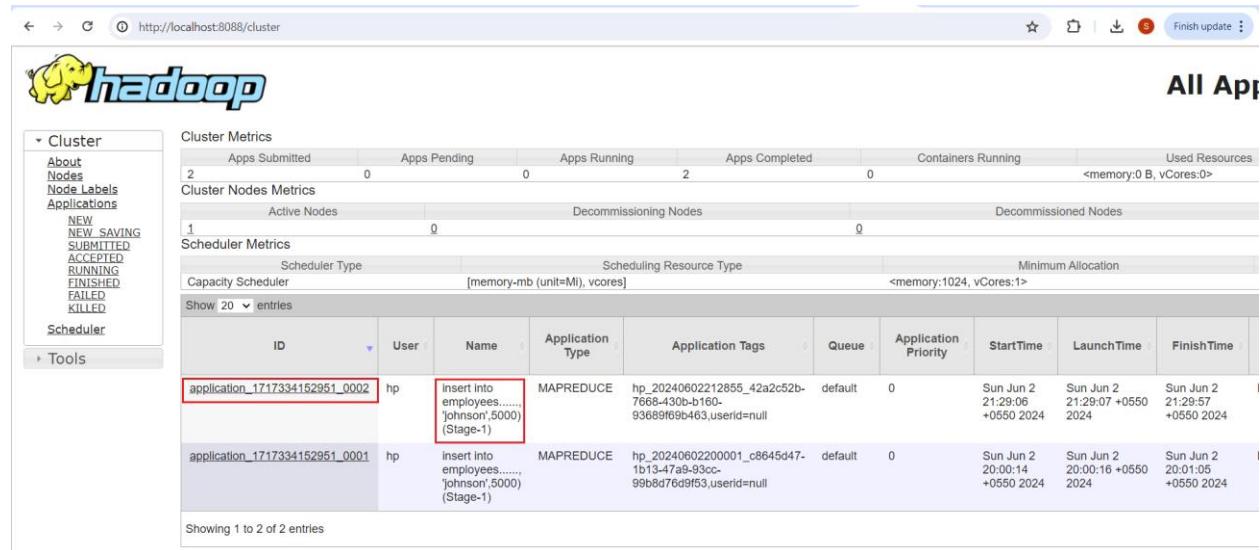
```
create database hive_local_derby_db;
show databases;
use hive_local_derby_db;
create table employees(emp_id int, emp_name string, emp_salary int);
show tables;
insert into employees values (101, 'johnson',5000);
```

The above insert statement submits MapReduce job

```
0: jdbc:hive2://> create database hive_local_derby_db;
0: jdbc:hive2://> show databases;
+-----+
| database_name |
+-----+
| default      |
| hive_local_derby_db |
+-----+
2 rows selected (0.919 seconds)
0: jdbc:hive2://> use hive_local_derby_db;
No rows affected (0.057 seconds)
0: jdbc:hive2://> create table employees(emp_id int, emp_name string, emp_salary int);
No rows affected (1.989 seconds)
0: jdbc:hive2://> show tables;
+-----+
| tab_name |
+-----+
| employees |
+-----+
1 row selected (0.202 seconds)
0: jdbc:hive2://> insert into employees values (101, 'johnson',5000);
24/06/02 21:29:01 [HiveServer2-Background-Pool: Thread-66]: WARN ql.Driver: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. tez) or using Hive 1.X releases.
Query ID = hp_20240602212855_42a2c52b-7668-430b-b160-93689f69b463
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
WARN : Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. tez) or using Hive 1.X releases.
24/06/02 21:29:03 [HiveServer2-Background-Pool: Thread-66]: WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
Starting Job = job_1717334152951_0002, Tracking URL = http://DESKTOP-KGH2E26:8088/proxy/application\_1717334152951\_0002/
Kill Command = D:\ProgramFiles\Hadoop\hadoop-3.3.6\bin\mapred job -kill job_1717334152951_0002
Hadoop job information for Stage-1: number of mappers: 1 number of reducers: 1
24/06/02 21:29:25 [HiveServer2-Background-Pool: Thread-66]: WARN mapreduce.Counters: Group org.apache.hadoop.mapred.Task$Counter is deprecated. Use org.apache.hadoop.mapreduce.TaskCounter instead
2024-06-02 21:29:25,701 Stage-1 map = 0%, reduce = 0%
2024-06-02 21:29:41,716 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 8.919 sec
2024-06-02 21:29:57,719 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 18.996 sec
MapReduce Total cumulative CPU time: 18 seconds 996 msec
Ended Job = job_1717334152951_0002
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://localhost:9800/user/hive/warehouse/hive_local_derby_db.db/employees/.hive-staging_hive_2024-06-02_21-28-55_427_60593639711147249-1-ext-10000
Loading data to table hive_local_derby_db.employees
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 18.996 sec HDFS Read: 24599 HDFS Write: 302 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 18 seconds 996 msec
24/06/02 21:30:00 [HiveServer2-Background-Pool: Thread-66]: ERROR operation.SQLOperation: Error running hive query
```

We can track the MapReduce job in YARN UI: <http://localhost:8088/cluster>

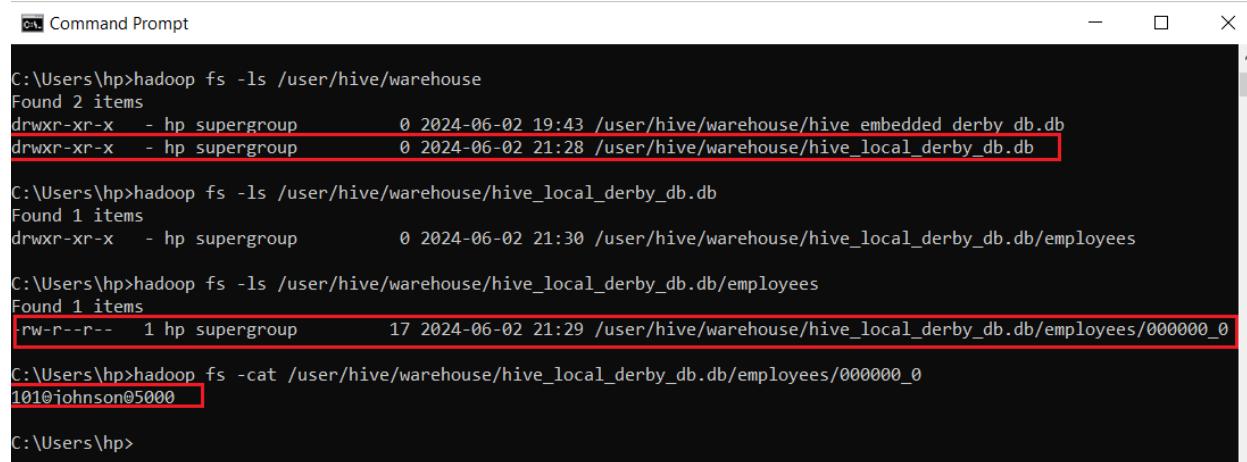
In YARN UI, you can see an application name with `insert into employees...` that was executed.



ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime
application_1717334152951_0002	hp	insert into employees..... [Johnson',5000) (Stage-1)	MAPREDUCE	hp_20240602212855_42a2c52b- 7668-430b-b160- 93689f69b463.userid=null	default	0	Sun Jun 2 21:29:06 +0550 2024	Sun Jun 2 21:29:07 +0550 2024	Sun Jun 2 21:29:57 +0550 2024
application_1717334152951_0001	hp	insert into employees..... [Johnson',5000) (Stage-1)	MAPREDUCE	hp_20240602200001_c8645d47- 1b13-47a9-93cc- 99b8d76d9f53.userid=null	default	0	Sun Jun 2 20:00:14 +0550 2024	Sun Jun 2 20:00:16 +0550 2024	Sun Jun 2 20:01:05 +0550 2024

After the above insert query is completed, we can verify the output in HDFS using the following commands:

```
hadoop fs -ls /user/hive/warehouse
hadoop fs -ls /user/hive/warehouse/hive_local_derby_db.db
hadoop fs -ls /user/hive/warehouse/hive_local_derby_db.db/employees
hadoop fs -cat /user/hive/warehouse/hive_local_derby_db.db/employees/000000_0
```



```
C:\Users\hp>hadoop fs -ls /user/hive/warehouse
Found 2 items
drwxr-xr-x - hp supergroup          0 2024-06-02 19:43 /user/hive/warehouse/hive_embedded_derby_db.db
drwxr-xr-x - hp supergroup          0 2024-06-02 21:28 /user/hive/warehouse/hive_local_derby_db.db

C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_local_derby_db.db
Found 1 items
drwxr-xr-x - hp supergroup          0 2024-06-02 21:30 /user/hive/warehouse/hive_local_derby_db.db/employees

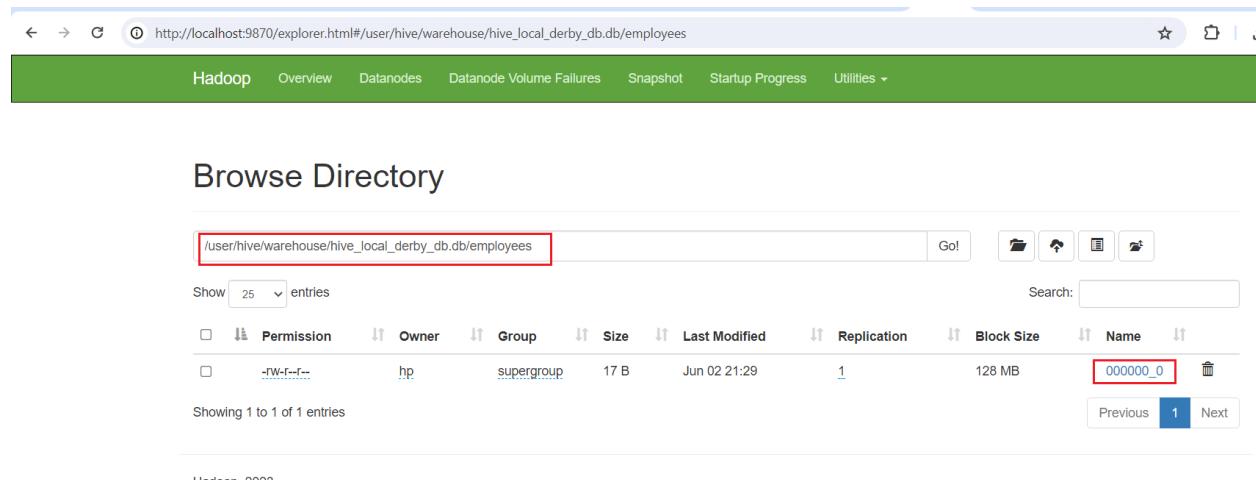
C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_local_derby_db.db/employees
Found 1 items
-rw-r--r--  1 hp supergroup         17 2024-06-02 21:29 /user/hive/warehouse/hive_local_derby_db.db/employees/000000_0

C:\Users\hp>hadoop fs -cat /user/hive/warehouse/hive_local_derby_db.db/employees/000000_0
1010johnson@5000

C:\Users\hp>
```

The same is visible in NameNode UI: <http://localhost:9870/dfshealth.html> also.

Open NameNode UI, go to **Utilities** tab and select **Browse the file system** option. Enter the directory name /user/hive/warehouse and you can see `hive_local_derby_db.db` folder available. Click on `hive_local_derby_db.db` folder to see `employees` folder in which `000000_0` file available. Click on the file and select **Head the file** or **Tail the file** to see the file contents. We can download this file by clicking on **Download** option.



The screenshot shows the HDFS NameNode UI's 'Browse Directory' interface. The address bar shows the URL [http://localhost:9870/explorer.html#/user/hive/warehouse/hive\\_local\\_derby\\_db.db/employees](http://localhost:9870/explorer.html#/user/hive/warehouse/hive_local_derby_db.db/employees). The main content area is titled 'Browse Directory' and shows a table of files in the '/user/hive/warehouse/hive\_local\_derby\_db.db/employees' directory. The table has columns: Permission, Owner, Group, Size, Last Modified, Replication, Block Size, and Name. One row is selected, showing the file '000000\_0'. The 'Name' column for this file is also highlighted with a red box. At the bottom of the table, there are navigation links for 'Previous', '1', and 'Next'.

Since we enabled Derby to run as Network server, we should be able to make multiple Hive connections. To test this, open another Command Prompt and start Hive shell and try to run a sample query (For example, `show databases;`).

```
beeline -u jdbc:hive2:// scott tiger
```

```
Command Prompt - beeline -u jdbc:hive2:// scott tiger
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>beeline -u jdbc:hive2:// scott tiger
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j
/sl4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.jar
!/org/slf4j/javax/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://
Hive Session ID = 43e9f936-6148-4cbd-a93c-d6be30cb2e3c
24/06/02 21:44:01 [main]: WARN hikari.HikariConfig: objectstore - leakDetectionThreshold is less than 2000ms or more than maxLifetime, disabling it.
24/06/02 21:44:02 [main]: WARN hikari.HikariConfig: objectstore-secondary - leakDetectionThreshold is less than 2000ms or more than maxLifetime, disabling it.
24/06/02 21:44:03 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:03 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:03 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:03 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:03 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:03 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:04 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:04 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:04 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:04 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:04 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:04 [main]: WARN DataNucleus.MetaData: Metadata has jdbc-type of null yet this is not valid. Ignored
24/06/02 21:44:04 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.cpc.UnionSk
etchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotat
ion and provide the description of the function.
24/06/02 21:44:08 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.hll.UnionSk
etchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annotat
ion and provide the description of the function.
24/06/02 21:44:08 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.Inter
sectSketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description a
nnotation and provide the description of the function.
24/06/02 21:44:08 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.Exclu
desketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description ann
otation and provide the description of the function.
24/06/02 21:44:08 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.theta.Union
SketchUDF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec.Description annot
ation and provide the description of the function.
24/06/02 21:44:08 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.tuple.Array
OfDoublesSketchToValuesUDTF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec
24/06/02 21:44:08 [main]: WARN exec.FunctionRegistry: UDF Class org.apache.hive.org.apache.datasetches.hive.tuple.Array
OfDoublesSketchToValuesUDTF does not have description. Please annotate the class with the org.apache.hadoop.hive.ql.exec
.Description annotation and provide the description of the function.
24/06/02 21:44:09 [main]: WARN session.SessionState: Configuration hive.reloadable.aux.jars.path not specified
Connected to: Apache Hive (version 4.0.0)
Driver: Hive JDBC (version 4.0.0)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 4.0.0 by Apache Hive
0: jdbc:hive2://> show databases;
+-----+
| database_name |
+-----+
| default      |
| hive_local_derby_db |
+-----+
2 rows selected (3.72 seconds)
0: jdbc:hive2://>
```

We are able to access metastore database without any issues while other Beeline session is still active and accessing the same database.

To come out of beeline> shell, use !quit command.

## 8.8. Start HiveServer2 Service:

Before starting Beeline in remote mode that connects to HiveServer2, make sure the HiveServer2 service is running.

Open command prompt and start the HiveServer2 service using the following command:

## hiveserver2

If you wish to see more logging of hiveserver2, then you need to set `HADOOP_CLIENT_OPTS` variable to display logging to console and trigger `hiveserver2` as below:

```
set HADOOP_CLIENT_OPTS=-Dhive.root.logger=console  
hiveserver2
```

```

2024-06-02T21:50:34,461 INFO [main] http.HttpServer: ASYNC_PROFILER_HOME env or -Dasync.profiler.home not specified. Disabling /prof endpoint..
2024-06-02T21:50:34,474 INFO [main] service.AbstractService: Service:OperationManager is started.
2024-06-02T21:50:34,475 INFO [main] service.AbstractService: Service:SessionManager is started.
2024-06-02T21:50:34,478 INFO [main] service.AbstractService: Service:CLIService is started.
2024-06-02T21:50:34,478 INFO [main] service.AbstractService: Service:ThriftBinaryCLIService is started.
2024-06-02T21:50:34,893 INFO [main] thrift.ThriftCLIService: Starting ThriftBinaryCLIService on port 10000 with 5...500 worker threads
2024-06-02T21:50:34,893 INFO [main] service.AbstractService: Service:HiveServer2 is started.
2024-06-02T21:50:34,901 INFO [main] server.Server: jetty-9.4.45.v20220203; built: 2022-02-03T09:14:34.105Z; git: 4a0c91cobe53805e3fcffcdcc9587d5301863db; jvm 1.8.0_411-b09
2024-06-02T21:50:35,334 INFO [main] server.session: DefaultSessionIdManager workerName=node0
2024-06-02T21:50:35,334 INFO [main] server.session: No SessionScavenger set, using defaults
2024-06-02T21:50:35,340 INFO [main] server.session: node0 Scavenging every 660000ms
2024-06-02T21:50:35,508 INFO [main] handler.ContextHandler: Started o.e.j.w.WebAppContext@18b58c77{hiveserver2,/file:///C:/Users/hp/AppData/Local/Temp/jetty-0_0_0-10002-hive-service-4_0_0_jar_-any-6173310191469174998/webapp/,AVAILABLE}{jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/hive-service-4.0.0.jar!/hive-webapps/hiveserver2}
2024-06-02T21:50:35,510 INFO [main] handler.ContextHandler: Started o.e.j.s.ServletContextHandler@36cf6377{static,/static,jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/hive-service-4.0.0.jar!/hive-webapps/static,AVAILABLE}
2024-06-02T21:50:35,540 INFO [main] server.AbstractConnector: Started ServerConnector@4ae5ddc0{HTTP/1.1}{{0.0.0.0:10002}}
2024-06-02T21:50:35,540 INFO [main] server.Server: Started @23259ms
2024-06-02T21:50:35,540 INFO [main] server.HiveServer2: Web UI has started on port 10002
2024-06-02T21:50:35,540 INFO [main] http.HttpServer: Started HttpServer[hiveserver2] on port 10002
2024-06-02T21:51:33,796 INFO [Scheduled Query Poller] HiveMetaStore.audit: ugi=hp ip=unknown-ip-addr cmd=scheduled_query_poll
2024-06-02T21:51:34,061 INFO [NotificationEventPoll 0] metastore.HiveMetaStoreClient: HMS client filtering is enabled.

```

We can see that HiveServer2 service started on port 10002 while ThriftBinaryCLIService started on port 10000.

## 8.9. Beeline Remote Connection:

In Remote mode, Beeline connects to HiveServer2 over Thrift CLI servers that runs on 10000 port. Beeline remote connection can be made as **anonymous** user or with **specific credentials**.

Use the following command to connect Beeline as anonymous user:

```
beeline -u jdbc:hive2://localhost:10000/
```

```
C:\Users\hp>beeline -u jdbc:hive2://localhost:10000/
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://localhost:10000/
24/06/02 21:55:21 [main]: WARN jdbc.HiveConnection: Failed to connect to localhost:10000
Error: Could not open client transport with JDBC Uri: jdbc:hive2://localhost:10000/: Failed to open new session: java.lang.RuntimeException: org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.authorize.AuthorizationException):
User: hp is not allowed to impersonate anonymous (state=08S01,code=0)
```

Here, we see an error that "**AuthorizationException: User xxx is not allowed to impersonate anonymous**". This is because HiveServer2 does not allow impersonation by default.

Use the following command to connect Beeline with default user scott:

```
beeline -u jdbc:hive2://localhost:10000/ -n scott -p tiger
```

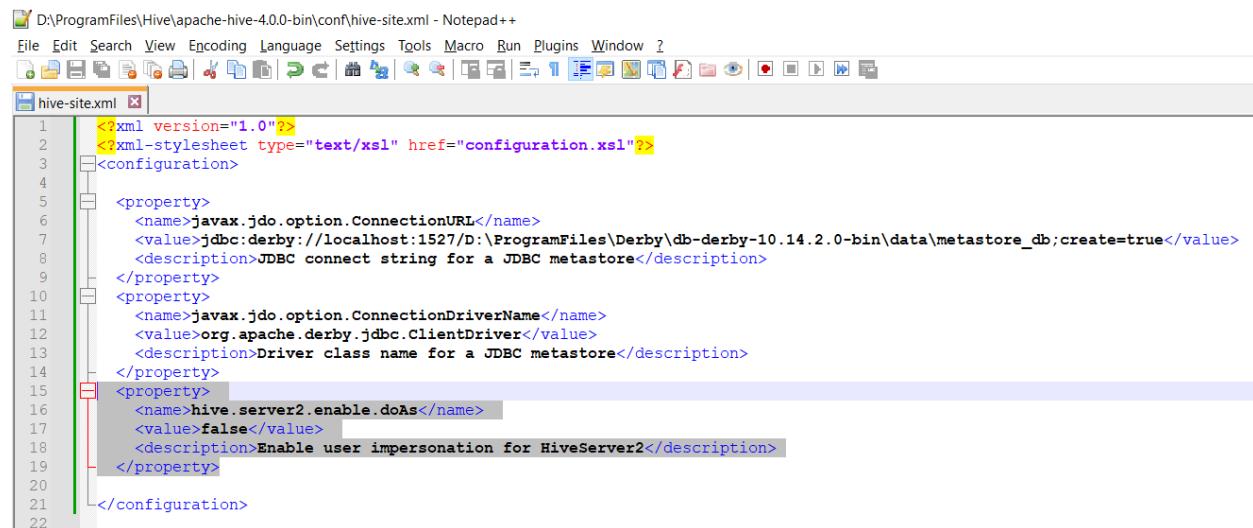
```
C:\Users\hp>beeline -u jdbc:hive2://localhost:10000/ -n scott -p tiger
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j
impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://localhost:10000
24/06/02 21:56:11 [main]: WARN jdbc.HiveConnection: Failed to connect to localhost:10000
Error: Could not open client transport with JDBC Uri: jdbc:hive2://localhost:10000/: Failed to open new session: java.lang.RuntimeException: org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.authorize.AuthorizationException):
User: hp is not allowed to impersonate scott (state=08S01,code=0)

C:\Users\hp>
```

Here, we see an error that "**AuthorizationException: User xxx is not allowed to impersonate anonymous**". This is because HiveServer2 does not allow impersonation by default.

To fix the above error, open `hive_site.xml` file in `HIVE_HOME\conf` location and add the following property between `<configuration>` and `</configuration>` tag.

```
<property>
    <name>hive.server2.enable.doAs</name>
    <value>false</value>
    <description>Enable user impersonation for HiveServer2</description>
</property>
```



Stop and restart HiveServer2 service using the following command

```
set HADOOP_CLIENT_OPTS=-Dhive.root.logger=console
hiveserver2
```

Now, we are able to connect to Beeline in remote mode and run queries without any issue.

```
beeline -u jdbc:hive2://localhost:10000/ -n scott -p tiger
```

```
show databases;
use hive_local_derby_db;
show tables;
select * from employees;
```

```
Command Prompt - beeline -u jdbc:hive2://localhost:10000/ -n scott -p tiger
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>beeline -u jdbc:hive2://localhost:10000/ -n scott -p tiger
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://localhost:10000/
Connected to: Apache Hive (version 4.0.0)
Driver: Hive JDBC (version 4.0.0)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 4.0.0 by Apache Hive
0: jdbc:hive2://localhost:10000/> show databases;
INFO : Compiling command(queryId=hp_20240602220048_b6c0cfde-47b7-4ec9-9e59-b8b93e4bcdf9): show databases
INFO : Semantic Analysis Completed (retryal = false)
INFO : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:database_name, type:string, comment:from deserializer)], properties:null)
INFO : Completed compiling command(queryId=hp_20240602220048_b6c0cfde-47b7-4ec9-9e59-b8b93e4bcdf9); Time taken: 2.631 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602220048_b6c0cfde-47b7-4ec9-9e59-b8b93e4bcdf9): show databases
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240602220048_b6c0cfde-47b7-4ec9-9e59-b8b93e4bcdf9); Time taken: 0.225 seconds
+-----+
| database_name |
+-----+
| default      |
| hive_local_derby_db |
+-----+
2 rows selected (3.957 seconds)
0: jdbc:hive2://localhost:10000/> use hive_local_derby_db;
```

## 9. Configure Remote MySQL Metastore:

Now, we will see how to use MySQL as the remote metastore database.

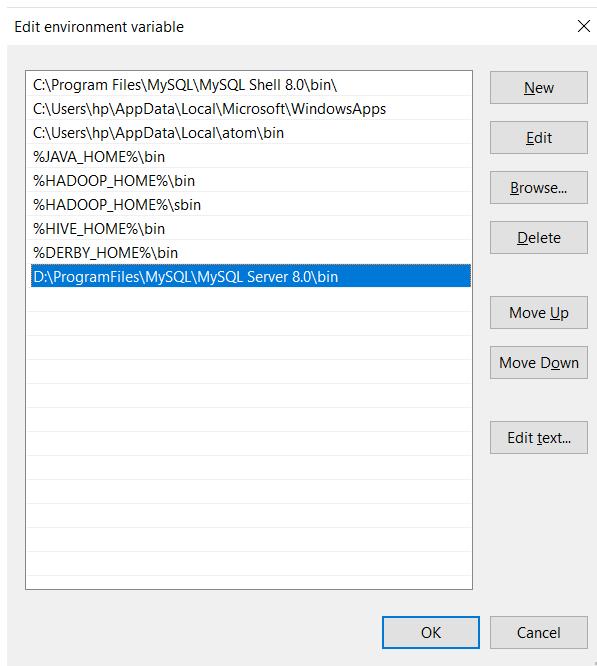
### 9.1. Install MySQL Server:

First, install MySQL server from the [official MySQL Downloads](#) website in the machine if was not already installed (*I installed the latest MySQL Server 8.0 version to D:\ProgramFiles\MySQL directory*)

Note that MySQL runs on port **3306** by default.

Name	Date modified	Type	Size
bin	5/29/2024 12:09 PM	File folder	
docs	5/29/2024 12:09 PM	File folder	
etc	5/29/2024 12:09 PM	File folder	
include	5/29/2024 12:09 PM	File folder	
lib	5/29/2024 12:09 PM	File folder	
share	5/29/2024 12:09 PM	File folder	
LICENSE	3/27/2024 7:22 PM	File	276 KB
LICENSE.router	3/27/2024 7:22 PM	ROUTER File	114 KB
README	3/27/2024 7:22 PM	File	1 KB
README.router	3/27/2024 7:22 PM	ROUTER File	1 KB

Open the **Environment Variables** dialog, select **PATH** variable under **User variables** section and press **Edit** button. Press **New** and add MySQL Install Path\bin value (*for example, D:\ProgramFiles\MySQL\MySQL Server 8.0\bin*) and press OK. Then press OK again to apply environment variable changes and close window.



## 9.2. Create Metastore DB in MySQL:

In MySQL Server, create a database for Hive metastore.

Open MySQL command prompt or Workbench.

To launch mysql> command prompt with root user, execute this command:

```
mysql -h localhost -u root -p
```

```
Command Prompt - mysql -h localhost -u root -p
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>mysql -h localhost -u root -p
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 1805
Server version: 8.0.37 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Run the following queries to create a database named `hive_metastore` and a new user for Hive named `hive` and password as `HiveAdmin` in MySQL server.

```
CREATE DATABASE hive_metastore;
USE hive_metastore;
CREATE USER 'hive'@'localhost' IDENTIFIED WITH mysql_native_password BY 'HiveAdmin';
GRANT ALL PRIVILEGES ON hive_metastore.* TO 'hive'@'localhost';
FLUSH PRIVILEGES;
```

**Note:**

Make sure that `hive_metastore` database and `hive` user are not already available in your MySQL server. If they are already available, use a different database name and user name to create. Otherwise, you may encounter below errors:

**ERROR 1007 (HY000): Can't create database 'hive\_metastore'; database exists**  
**ERROR 1396 (HY000): Operation CREATE USER failed for 'hive'@'localhost'**

```
mysql> CREATE DATABASE hive_metastore;
Query OK, 1 row affected (0.12 sec)

mysql> USE hive_metastore;
Database changed
mysql> CREATE USER 'hive'@'localhost' IDENTIFIED WITH mysql_native_password BY 'HiveAdmin';
Query OK, 0 rows affected (0.13 sec)

mysql> GRANT ALL PRIVILEGES ON hive_metastore.* TO 'hive'@'localhost';
Query OK, 0 rows affected (0.10 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.09 sec)

mysql>
```

### 9.3. Download MySQL JDBC Driver:

Since Hive does not provide the JDBC driver for MySQL by default, we need to explicitly get the driver and place it in `HIVE_HOME\lib` directory

Download **JDBC Driver for MySQL (Connector/J)** from the below link

<https://dev.mysql.com/downloads/connector/j/>

In the above link, choose **Platform Independent** Operating System and download `mysql-connector-j-* .zip` file as shown below:

The screenshot shows the MySQL Community Downloads website. At the top, there's a navigation bar with back, forward, and search icons, followed by the URL `dev.mysql.com/downloads/connector/j/`. Below the navigation is a breadcrumb trail: `MySQL Community Downloads > Connector/J`. A navigation bar at the top of the page includes tabs for "General Availability (GA) Releases" (which is selected), "Archives", and "Signature". The main content area is titled "Connector/J 8.4.0". It asks to "Select Operating System:" and has a dropdown menu set to "Platform Independent". Below this, two download options are listed:

Platform Independent (Architecture Independent), Compressed TAR Archive (mysql-connector-j-8.4.0.tar.gz)	8.4.0	4.1M	<a href="#">Download</a>
Platform Independent (Architecture Independent), ZIP Archive (mysql-connector-j-8.4.0.zip)	8.4.0	4.9M	<a href="#">Download</a>

Each download row includes a MD5 hash and a "Signature" link.

The latest version of zip file at the time of writing this document is `mysql-connector-j-8.4.0.zip`. You can download the file of whichever latest version that you could see.

After downloading the `zip` file, unzip it which creates `mysql-connector-j-*` directory. Open the directory and copy `mysql-connector-j-* .jar` file to `HIVE_HOME\lib` directory.

A screenshot of a Windows File Explorer window showing the contents of the `apache-hive-4.0.0-bin\lib` directory. The path in the address bar is `is PC > Local Disk (D:) > ProgramFiles > Hive > apache-hive-4.0.0-bin > lib`. The table lists four files:

Name	Date modified	Type	Size
minlog-1.3.1.jar	1/22/2020 8:40 PM	Executable Jar File	6 KB
mysql-connector-j-8.4.0.jar	3/13/2024 12:43 AM	Executable Jar File	2,475 KB
netty-3.10.5.Final.jar	1/22/2020 8:40 PM	Executable Jar File	1,300 KB

#### 9.4. Configure Hive Site File:

Now, we need to configure Metastore service to communicate with MySQL database.

Open `hive-site.xml` file in `HIVE_HOME\conf` directory and replace the existing properties with the following properties between `<configuration>` and `</configuration>` element.

```
<property>
    <name>javax.jdo.option.ConnectionURL</name>
    <value>jdbc:mysql://localhost:3306/hive_metastore</value>
    <description>JDBC connect string for a JDBC metastore</description>
</property>
<property>
    <name>javax.jdo.option.ConnectionDriverName</name>
    <value>com.mysql.jdbc.Driver</value>
    <description>Driver class name for a JDBC metastore</description>
</property>
<property>
    <name>javax.jdo.option.ConnectionUserName</name>
    <value>hive</value>
    <description>Username to use against metastore database</description>
</property>
<property>
    <name>javax.jdo.option.ConnectionPassword</name>
    <value>HiveAdmin</value>
    <description>password to use against metastore database</description>
</property>
<property>
    <name>hive.metastore.uris</name>
    <value>thrift://127.0.0.1:9083</value>
    <description>Thrift URI for the remote metastore. Used by metastore
client to connect to remote metastore</description>
</property>
<property>
    <name>hive.server2.enable.doAs</name>
    <value>false</value>
    <description>Enable user impersonation for HiveServer2</description>
</property>
```

```

1 <?xml version="1.0"?>
2 <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
3 <configuration>
4
5   <property>
6     <name>javax.jdo.option.ConnectionURL</name>
7     <value>jdbc:mysql://localhost:3306/hive_metastore</value>
8     <description>JDBC connect string for a JDBC metastore</description>
9   </property>
10  <property>
11    <name>javax.jdo.option.ConnectionDriverName</name>
12    <value>com.mysql.jdbc.Driver</value>
13    <description>Driver class name for a JDBC metastore</description>
14  </property>
15  <property>
16    <name>javax.jdo.option.ConnectionUserName</name>
17    <value>hive</value>
18    <description>Username to use against metastore database</description>
19  </property>
20  <property>
21    <name>javax.jdo.option.ConnectionPassword</name>
22    <value>HiveAdmin</value>
23    <description>password to use against metastore database</description>
24  </property>
25  <property>
26    <name>hive.metastore.uris</name>
27    <value>thrift://127.0.0.1:9083</value>
28    <description>Thrift URI for the remote metastore. Used by metastore client to connect to remote metastore</description>
29  </property>
30  <property>
31    <name>hive.server2.enable.doAs</name>
32    <value>false</value>
33    <description>Enable user impersonation for HiveServer2</description>
34  </property>
35
36 </configuration>

```

## 9.5. Initialize Metastore DB:

Now, run the `schematool` utility to create the initial DB structure in MySQL database using the following command

```
schematool -dbType mysql -initSchema
```

```

C:\Users\hp>schematool -dbType mysql -initSchema
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>schematool -dbType mysql -initSchema
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Initializing the schema to: 4.0.0
Metastore connection URL: jdbc:mysql://localhost:3306/hive_metastore
Metastore connection Driver: com.mysql.jdbc.Driver
Metastore connection User: hive
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SPI and manual loading of the driver class is generally unnecessary.
Starting metastore schema initialization to 4.0.0
Initialization script hive-schema-4.0.0.mysql.sql

Initialization script completed
C:\Users\hp>

```

## 9.6. Verify Metastore in MySQL:

Let us connect to MySQL server and verify Hive metastore created under `hive_metastore` database.

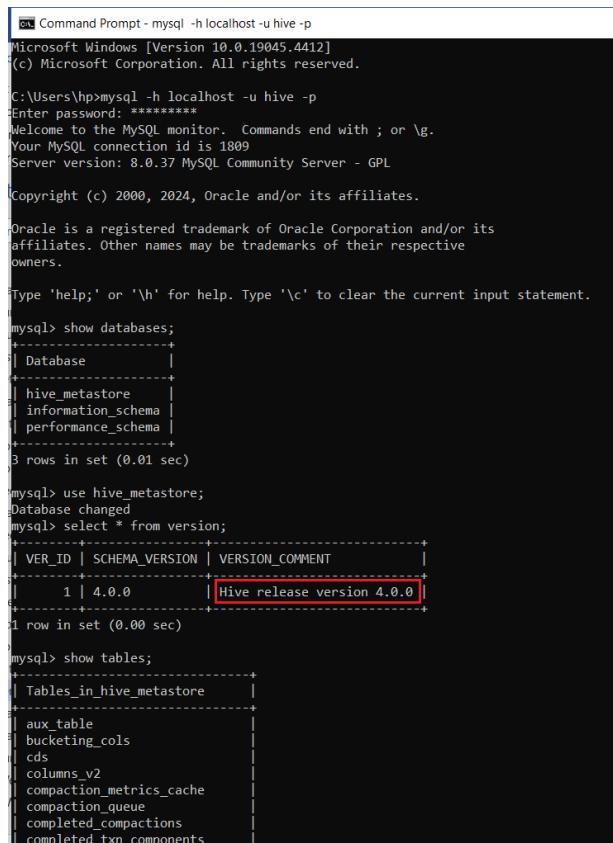
Launch MySQL command prompt using this command. Note that we are connecting with `hive` user credentials that we created earlier.

```
mysql -h localhost -u hive -p
```

Provide the password of `hive` user when asked.

In `mysql>` prompt, run the following queries:

```
show databases;
use hive_metastore;
select * from version;
show tables;
```



The screenshot shows a Command Prompt window titled "Command Prompt - mysql -h localhost -u hive -p". It displays the following MySQL session:

```
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>mysql -h localhost -u hive -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1809
Server version: 8.0.37 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help,' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database      |
+-----+
| hive_metastore |
| information_schema |
| performance_schema |
+-----+
3 rows in set (0.01 sec)

mysql> use hive_metastore;
Database changed
mysql> select * from version;
+-----+-----+-----+
| VER_ID | SCHEMA_VERSION | VERSION_COMMENT      |
+-----+-----+-----+
| 1      | 4.0.0          | Hive release version 4.0.0 |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_hive_metastore |
+-----+
| aux_table
| bucketing_cols
| cds
| columns_v2
| compaction_metrics_cache
| compaction_queue
| completed_compactions
| completed_txn_components |
+-----+
```

Here, we can see that Hive metastore has created tables and the version of Hive release **4.0.0**.

## 9.7. Start Hive Metastore service:

Hive will be able to connect to remote metastore in MySQL database using Thrift URIs.  
So, let us start the `metastore` service using the below command to connect to our MySQL server.

```
set HADOOP_CLIENT_OPTS=-Dhive.root.logger=console  
hive --service metastore
```

```
Command Prompt - hive --service metastore  
Microsoft Windows [Version 10.0.19045.4412]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\hp>set HADOOP_CLIENT_OPTS=-Dhive.root.logger=console  
  
C:\Users\hp>hive --service metastore  
"Starting Hive Metastore Server"  
SLF4J: Class path contains multiple SLF4J bindings.  
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]  
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]  
2024-06-02T22:10:45,794 INFO [main] conf.MetastoreConf: Found configuration file: file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/conf/hive-site.xml  
2024-06-02T22:10:45,980 INFO [main] conf.MetastoreConf: Unable to find config file: hivemetastore-site.xml  
2024-06-02T22:10:45,982 INFO [main] conf.MetastoreConf: Unable to find config file: metastore-site.xml  
2024-06-02T22:10:46,075 INFO [main] metastore.HiveMetaStore: STARTUP_MSG:  
*****  
STARTUP_MSG: Starting HiveMetaStore  
STARTUP_MSG: host = DESKTOP-KGH2E2G/192.168.56.1  
STARTUP_MSG: args = []  
STARTUP_MSG: version = 4.0.0  
STARTUP_MSG: classpath = D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\conf;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\accessors-smart-2.5.0.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\accumulo-core-1.10.1.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\accumulo-fate-1.10.1.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\accumulo-start-1.10.1.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\accumulo-trace-1.10.1.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\aircompressor-0.21.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\animal-sniffer-annotations-1.14.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\annotations-17.0.0.jar;D:\ProgramFiles\Hive\apache-hive-4.0.0-bin\lib\annotations-00ms or more than maxLifetime, disabling it.  
2024-06-02T22:10:49,152 INFO [main] hikari.HikariDataSource: objectstore-secondary - Starting...  
2024-06-02T22:10:49,199 INFO [main] hikari.HikariDataSource: objectstore-secondary - Start completed.  
2024-06-02T22:10:50,586 INFO [main] metastore.PersistenceManagerProvider: Setting MetaStore object pin classes with hive.metastore.cache.pinobjtypes="Table,StorageDescriptor,SerDeInfo,Partition,Database,Type,FieldSchema,Order"  
2024-06-02T22:10:50,587 INFO [main] metastore.ObjectStore: RawStore: org.apache.hadoop.hive.metastore.ObjectStore@ea9e141, with PersistenceManager: null will be shutdown  
2024-06-02T22:10:50,633 INFO [main] metastore.ObjectStore: RawStore: org.apache.hadoop.hive.metastore.ObjectStore@ea9e141, with PersistenceManager: org.datanucleus.api.jdo.JDOPersistenceManager@1f387978 created in the thread with id: 1  
2024-06-02T22:10:56,483 INFO [main] metastore.HMSHandler: Created RawStore: org.apache.hadoop.hive.metastore.ObjectStore@ea9e141  
2024-06-02T22:10:56,752 INFO [main] metastore.HMSHandler: Setting location of default catalog, as it hasn't been done after upgrade  
2024-06-02T22:10:59,113 INFO [main] metastore.HMSHandler: Started creating a default database with name: default  
2024-06-02T22:10:59,439 INFO [main] metastore.HMSHandler: Successfully created a default database with name: default  
2024-06-02T22:10:59,588 INFO [main] metastore.HMSHandler: Added admin role in metastore  
2024-06-02T22:10:59,665 INFO [main] metastore.HMSHandler: Added public role in metastore  
2024-06-02T22:10:59,853 INFO [main] metastore.HMSHandler: No user is added in admin role, since config is empty  
2024-06-02T22:10:59,865 INFO [main] metastore.HMSHandler: HMS server filtering is disabled by configuration  
2024-06-02T22:11:00,467 INFO [main] metastore.HiveMetaStore: Starting DB backed MetaStore Server with SetUGI enabled  
2024-06-02T22:11:00,488 INFO [main] metastore.HiveMetaStore: Direct SQL optimization = true  
2024-06-02T22:11:00,581 INFO [main] metastore.HMSHandler: Started the new metaserver on port [9083]...  
2024-06-02T22:11:00,596 INFO [main] metastore.HMSHandler: Options.minWorkerThreads = 200  
2024-06-02T22:11:00,609 INFO [main] metastore.HMSHandler: Options.maxWorkerThreads = 1000  
2024-06-02T22:11:00,622 INFO [main] metastore.HMSHandler: TCP keepalive = true  
2024-06-02T22:11:00,629 INFO [main] metastore.HMSHandler: Enable SSL = false  
2024-06-02T22:11:01,560 INFO [Metastore threads starter thread] leader.StaticLeaderElection: metastore.housekeeping.leader.hostname is empty. Start all the housekeeping threads.  
2024-06-02T22:11:01,562 INFO [Metastore threads starter thread] metastore.HiveMetaStore: Compaction HMS parameters:  
2024-06-02T22:11:01,564 INFO [Metastore threads starter thread] metastore.HiveMetaStore: metastore.comactor.initiator
```

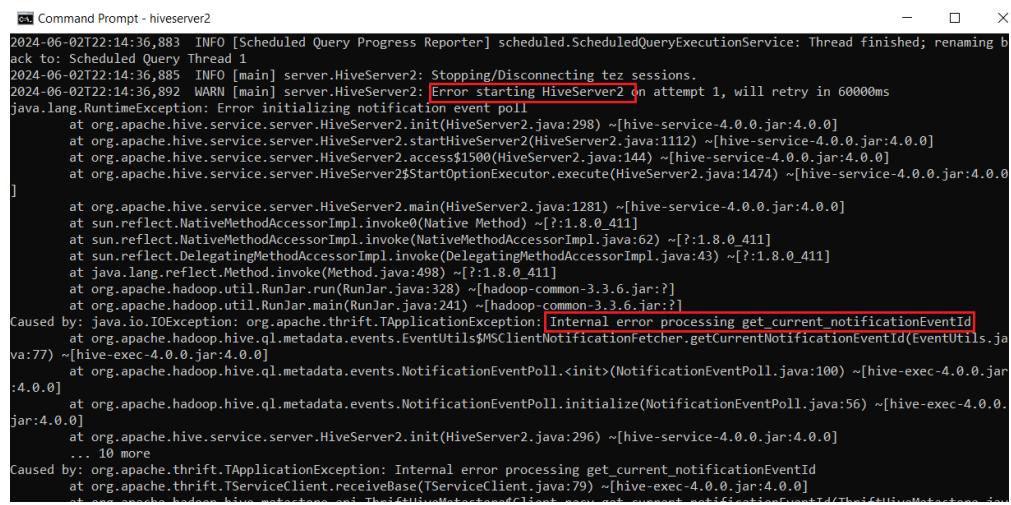
On the console, we can see that **metaserver** started on port **9083**

## 9.8. Start HiveServer2 Service:

To start Beeline in remote mode, HiveServer2 service must be running. If HiveServer2 is already running, stop it and start it freshly because we made configuration changes in `hive_site.xml` that should be loaded into HiveServer2.

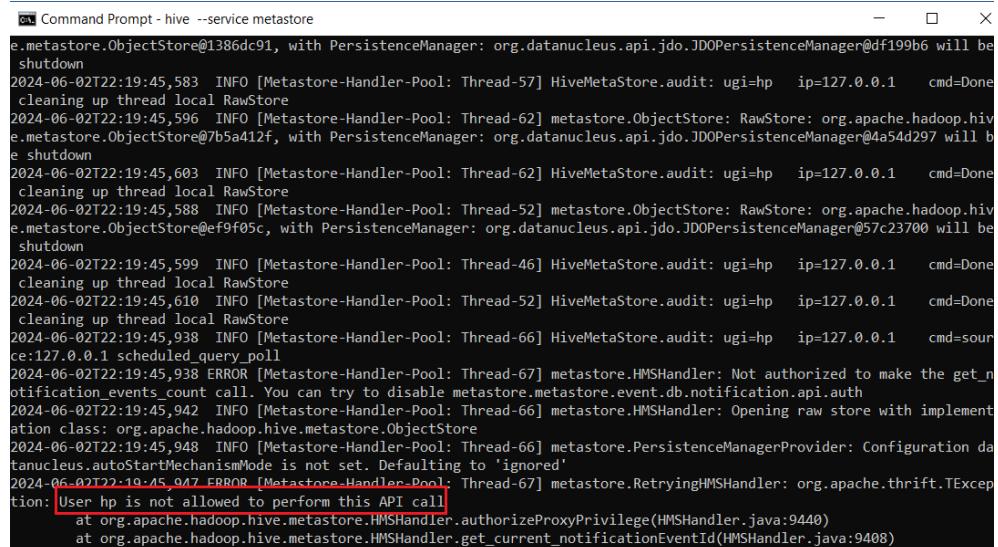
Open command prompt and start the HiveServer2 service using the following command

```
set HADOOP_CLIENT_OPTS=-Dhive.root.logger=console  
hiveserver2
```



```
2024-06-02T22:14:36,883 INFO [Scheduled Query Progress Reporter] scheduled.ScheduledQueryExecutionService: Thread finished; renaming back to: Scheduled Query Thread 1  
2024-06-02T22:14:36,885 INFO [main] server.HiveServer2: Stopping/Disconnecting tez sessions.  
2024-06-02T22:14:36,892 WARN [main] server.HiveServer2: [Error starting HiveServer2 on attempt 1, will retry in 60000ms  
java.lang.RuntimeException: Error initializing notification event poll  
    at org.apache.hive.service.server.HiveServer2.init(HiveServer2.java:298) ~[hive-service-4.0.0.jar:4.0.0]  
    at org.apache.hive.service.server.HiveServer2.startHiveServer2(HiveServer2.java:1112) ~[hive-service-4.0.0.jar:4.0.0]  
    at org.apache.hive.service.server.HiveServer2.access$1500(HiveServer2.java:144) ~[hive-service-4.0.0.jar:4.0.0]  
    at org.apache.hive.service.server.HiveServer2$StartOptionExecutor.execute(HiveServer2.java:1474) ~[hive-service-4.0.0.jar:4.0.0]  
]  
    at org.apache.hive.service.server.HiveServer2.main(HiveServer2.java:1281) ~[hive-service-4.0.0.jar:4.0.0]  
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method) ~[:1.8.0_411]  
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62) ~[:1.8.0_411]  
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) ~[:1.8.0_411]  
    at java.lang.reflect.Method.invoke(Method.java:498) ~[:1.8.0_411]  
    at org.apache.hadoop.util.RunJar.run(RunJar.java:328) ~[hadoop-common-3.3.6.jar:?:?]  
    at org.apache.hadoop.util.RunJar.main(RunJar.java:241) ~[hadoop-common-3.3.6.jar:?:?]  
Caused by: java.io.IOException: org.apache.thrift.TApplicationException: Internal error processing get_current_notificationEventId  
    at org.apache.hadoop.hive.ql.metadata.events.EventUtils$MSClientNotificationFetcher.getCurrentNotificationEventId(EventUtils.ja  
va:77) ~[hive-exec-4.0.0.jar:4.0.0]  
    at org.apache.hadoop.hive.ql.metadata.events.NotificationEventPoll.<init>(NotificationEventPoll.java:100) ~[hive-exec-4.0.0.ja  
rn:4.0.0]  
    at org.apache.hadoop.hive.ql.metadata.events.NotificationEventPoll.initialize(NotificationEventPoll.java:56) ~[hive-exec-4.0.0.  
jar:4.0.0]  
    at org.apache.hive.service.server.HiveServer2.init(HiveServer2.java:296) ~[hive-service-4.0.0.jar:4.0.0]  
... 10 more  
Caused by: org.apache.thrift.TApplicationException: Internal error processing get_current_notificationEventId  
    at org.apache.thrift.TServiceClient.receiveBase(TServiceClient.java:79) ~[hive-exec-4.0.0.jar:4.0.0]  
    at org.apache.hadoop.hive.metastore.HiveMetaStore$Client.recv_get_current_notificationEventId(HiveMetaStore.java:104)
```

During HiveServer2 startup, we will encounter error “**Internal error processing get\_current\_notificationEventId**” and in the metastore logs (*go to the window where metastore service was started*), we can see an additional error “**User xxx is not allowed to perform this API call**”.

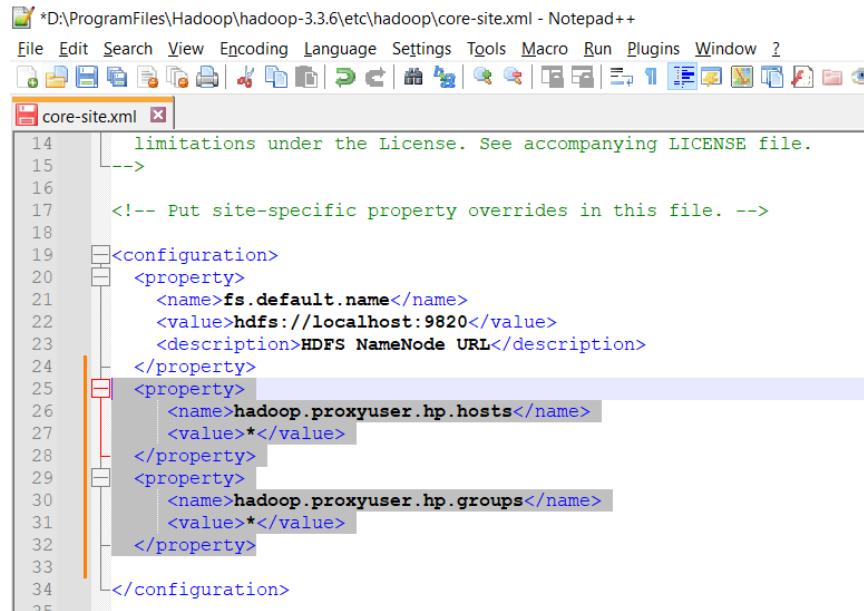


```
2024-06-02T22:19:45,583 INFO [Metastore-Handler-Pool: Thread-57] HiveMetaStore.audit: ugi=hp ip=127.0.0.1 cmd=Done cleaning up thread local RawStore  
2024-06-02T22:19:45,596 INFO [Metastore-Handler-Pool: Thread-62] metastore.ObjectStore: RawStore: org.apache.hadoop.hive.metastore.ObjectStore@7b5a412f, with PersistenceManager: org.datanucleus.api.jdo.JDOPersistenceManager@4a54d297 will be shutdown  
2024-06-02T22:19:45,603 INFO [Metastore-Handler-Pool: Thread-62] HiveMetaStore.audit: ugi=hp ip=127.0.0.1 cmd=Done cleaning up thread local RawStore  
2024-06-02T22:19:45,588 INFO [Metastore-Handler-Pool: Thread-52] metastore.ObjectStore: RawStore: org.apache.hadoop.hive.metastore.ObjectStore@eef9f05c, with PersistenceManager: org.datanucleus.api.jdo.JDOPersistenceManager@57c23700 will be shutdown  
2024-06-02T22:19:45,599 INFO [Metastore-Handler-Pool: Thread-46] HiveMetaStore.audit: ugi=hp ip=127.0.0.1 cmd=Done cleaning up thread local RawStore  
2024-06-02T22:19:45,610 INFO [Metastore-Handler-Pool: Thread-52] HiveMetaStore.audit: ugi=hp ip=127.0.0.1 cmd=Done cleaning up thread local RawStore  
2024-06-02T22:19:45,938 INFO [Metastore-Handler-Pool: Thread-66] HiveMetaStore.audit: ugi=hp ip=127.0.0.1 cmd=sources:127.0.0.1 scheduled_query_poll  
2024-06-02T22:19:45,938 ERROR [Metastore-Handler-Pool: Thread-67] metastore.HMSHandler: Not authorized to make the get_notification_events_count call. You can try to disable metastore.metastore.event.db.notification.api.auth  
2024-06-02T22:19:45,942 INFO [Metastore-Handler-Pool: Thread-66] metastore.HMSHandler: Opening raw store with implementation class: org.apache.hadoop.hive.metastore.ObjectStore  
2024-06-02T22:19:45,948 INFO [Metastore-Handler-Pool: Thread-66] metastore.PersistenceManagerProvider: Configuration datanucleus.autoStartMechanismMode is not set. Defaulting to 'ignored'  
2024-06-02T22:19:45,947 ERROR [Metastore-Handler-Pool: Thread-67] metastore.RetryingHMSHandler: org.apache.thrift.TException: User hp is not allowed to perform this API call  
    at org.apache.hadoop.hive.metastore.HMSHandler.authorizeProxyPrivilege(HMSHandler.java:9440)  
    at org.apache.hadoop.hive.metastore.HMSHandler.get_current_notificationEventId(HMSHandler.java:9408)
```

Those errors are raised because that the superuser does not have access to impersonate as Hive user. To resolve this issue, we need to configure proxy user in Hadoop's core-site.xml file as below:

- Open core-site.xml file available in %HADOOP\_HOME%\etc\hadoop location and add the following properties. In this configuration, replace \$superuser with your user name displayed in the above error (*in my case, I am running Hadoop with hp user and the same is being displayed in the above error*).

```
<property>
    <name>hadoop.proxyuser.$superuser.hosts</name>
    <value>*</value>
</property>
<property>
    <name>hadoop.proxyuser.$superuser.groups</name>
    <value>*</value>
</property>
```

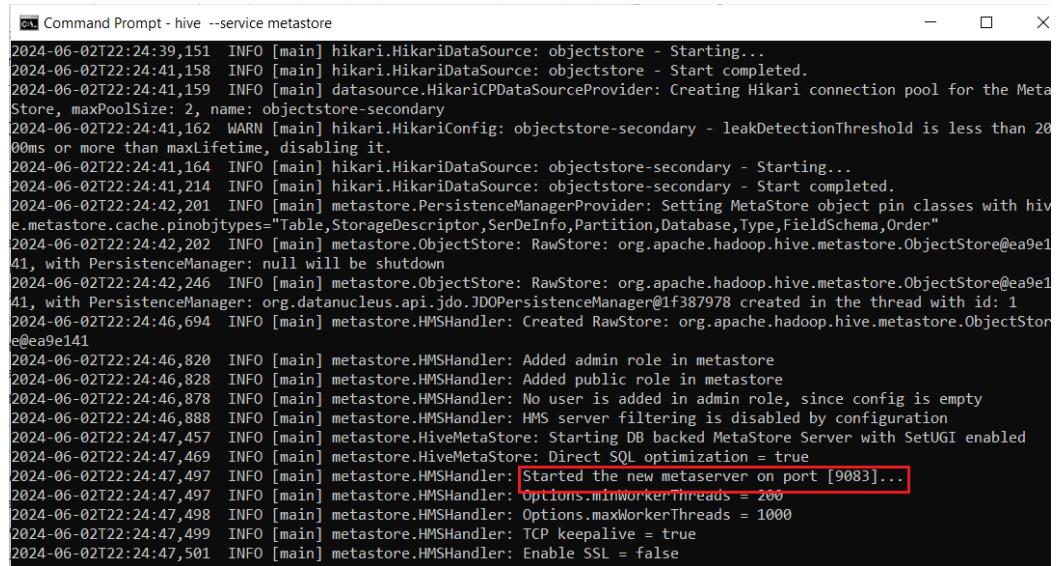


- After the above configuration change is done, restart Hadoop services first. Open **Windows Command Prompt** in **Administrator** mode and run the following commands.

```
stop-dfs.cmd
stop-yarn.cmd
start-dfs.cmd
start-yarn.cmd
```

- When Hadoop services are up and running, close and start Hive Metastore service using these commands.

```
set HADOOP_CLIENT_OPTS=-Dhive.root.logger=console
hive --service metastore
```



```
2024-06-02T22:24:39,151 INFO [main] hikari.HikariDataSource: objectstore - Starting...
2024-06-02T22:24:41,158 INFO [main] hikari.HikariDataSource: objectstore - Start completed.
2024-06-02T22:24:41,159 INFO [main] datasource.HikariCPDataSourceProvider: Creating Hikari connection pool for the MetaStore, maxPoolSize: 2, name: objectstore-secondary
2024-06-02T22:24:41,162 WARN [main] hikari.HikariConfig: objectstore-secondary - leakDetectionThreshold is less than 200ms or more than maxLifetime, disabling it.
2024-06-02T22:24:41,164 INFO [main] hikari.HikariDataSource: objectstore-secondary - Starting...
2024-06-02T22:24:41,214 INFO [main] hikari.HikariDataSource: objectstore-secondary - Start completed.
2024-06-02T22:24:42,201 INFO [main] metastore.PersistenceManagerProvider: Setting MetaStore object pin classes with hive.metastore.cache.pinobjtypes="Table,StorageDescriptor,SerDeInfo,Partition,Database,Type,FieldSchema,Order"
2024-06-02T22:24:42,202 INFO [main] metastore.ObjectStore: RawStore: org.apache.hadoop.hive.metastore.ObjectStore@ea9e141, with PersistenceManager: null will be shutdown
2024-06-02T22:24:42,246 INFO [main] metastore.ObjectStore: RawStore: org.apache.hadoop.hive.metastore.ObjectStore@ea9e141, with PersistenceManager: org.datanucleus.api.jdo.JDOPersistenceManager@1f387978 created in the thread with id: 1
2024-06-02T22:24:46,694 INFO [main] metastore.HMSHandler: Created RawStore: org.apache.hadoop.hive.metastore.ObjectStore@ea9e141
2024-06-02T22:24:46,820 INFO [main] metastore.HMSHandler: Added admin role in metastore
2024-06-02T22:24:46,828 INFO [main] metastore.HMSHandler: Added public role in metastore
2024-06-02T22:24:46,878 INFO [main] metastore.HMSHandler: No user is added in admin role, since config is empty
2024-06-02T22:24:46,888 INFO [main] metastore.HMSHandler: HMS server filtering is disabled by configuration
2024-06-02T22:24:47,457 INFO [main] metastore.HiveMetaStore: Starting DB backed MetaStore Server with SetUGI enabled
2024-06-02T22:24:47,469 INFO [main] metastore.HiveMetaStore: Direct SQL optimization = true
2024-06-02T22:24:47,497 INFO [main] metastore.HMSHandler: Started the new metaserver on port [9083]...
2024-06-02T22:24:47,497 INFO [main] metastore.HMSHandler: Options.minWorkerThreads = 200
2024-06-02T22:24:47,498 INFO [main] metastore.HMSHandler: Options.maxWorkerThreads = 1000
2024-06-02T22:24:47,499 INFO [main] metastore.HMSHandler: TCP keepalive = true
2024-06-02T22:24:47,501 INFO [main] metastore.HMSHandler: Enable SSL = false
```

Here, we can see that `metastore` service started on port 9083.

- After Hive Metastore is running, close and start HiveServer2 using these commands.

```
set HADOOP_CLIENT_OPTS=-Dhive.root.logger=console
hiveserver2
```

```

Command Prompt - hiveserver2
2024-06-02T22:33:25,266 INFO [Metastore-RuntimeStats-Loader-1] metastore.HiveMetaStoreClient: Resolved metastore uris: [thrift://127.0.0.1:9083]
2024-06-02T22:33:25,266 INFO [Metastore-RuntimeStats-Loader-1] metastore.HiveMetaStoreClient: Trying to connect to metastore with URI (thrift://127.0.0.1:9083) in binary transport mode
2024-06-02T22:33:25,270 INFO [Metastore-RuntimeStats-Loader-1] metastore.HiveMetaStoreClient: Opened a connection to metastore, URI (thrift://127.0.0.1:9083) current connections: 3
2024-06-02T22:33:25,285 INFO [Metastore-RuntimeStats-Loader-1] metastore.RetryingMetaStoreClient: RetryingMetaStoreClient proxy=class org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient ugi=hp (auth:SIMPLE) retries=1 delay=1 lifetime=0
2024-06-02T22:33:25,830 INFO [HiveMaterializedViewsRegistry-0] metadata.HiveMaterializedViewsRegistry: Materialized views registry has been initialized
2024-06-02T22:33:25,852 INFO [main] metadata.HiveMetaStoreClientWithLocalCache: Local cache initialized in HiveMetaStoreClient: com.github.bennanies.caffeine.cache.BoundedLocalCache$BoundedLocalManualCache@1aed6f0b
2024-06-02T22:33:25,929 INFO [main] events.NotificationEventPoll: Initializing lastCheckedEventId to 0
2024-06-02T22:33:25,935 INFO [main] server.HiveServer2: Compaction HS2 parameters:
2024-06-02T22:33:25,936 INFO [main] server.HiveServer2: hive.metastore.runworker.in = hs2
2024-06-02T22:33:25,937 INFO [main] server.HiveServer2: metastore.compactor.worker.threads = 0
2024-06-02T22:33:25,937 WARN [main] server.HiveServer2: Invalid number of Compactor Worker threads(0) on HS2
2024-06-02T22:33:25,942 INFO [main] server.HiveServer2: Initializing the compaction pools with using the global worker limit: 0
2024-06-02T22:33:25,943 WARN [main] server.HiveServer2: No default compaction pool configured, all non-labeled compaction requests will remain unprocessed!
2024-06-02T22:33:25,944 INFO [main] server.HiveServer2: This HS2 instance will act as Compactor with the following worker pool configuration:
Global pool size: 0

2024-06-02T22:33:25,946 INFO [main] server.HiveServer2: Starting Web UI on port 10002
2024-06-02T22:33:26,051 INFO [main] util.log: Logging initialized @17356ms to org.eclipse.jetty.util.log.Slf4jLog
2024-06-02T22:33:26,285 INFO [main] http.HttpServer: ASYNC_PROFILER_HOME env or -Dasync.profiler.home not specified. Disabling /prof endpoint..
2024-06-02T22:33:26,294 INFO [main] service.AbstractService: Service:OperationManager is started.
2024-06-02T22:33:26,294 INFO [main] service.AbstractService: Service:SessionManager is started.
2024-06-02T22:33:26,298 INFO [main] service.AbstractService: Service:CLIService is started.
2024-06-02T22:33:26,298 INFO [main] service.AbstractService: Service:ThriftBinaryCLIService is started.
2024-06-02T22:33:26,683 INFO [main] thrift.ThriftCLIService: Starting ThriftBinaryCLIService on port 10000 with 5...500 worker threads
2024-06-02T22:33:26,683 INFO [main] service.AbstractService: Service:HiveServer2 is started.
2024-06-02T22:33:26,688 INFO [main] server.Server: jetty-9.4.45.v20220203; built: 2022-02-03T09:14:34.105Z; git: 4a0c91c0be53805e3cffdc5cc9587d5301863db; jvm 1.8.0_411-b09
2024-06-02T22:33:26,950 INFO [main] server.session: DefaultSessionIdManager workerName=node0
2024-06-02T22:33:26,950 INFO [main] server.session: No SessionScavenger set, using defaults
2024-06-02T22:33:26,955 INFO [main] server.session: node0 Scavenging every 660000ms
2024-06-02T22:33:27,045 INFO [main] handler.ContextHandler: Started o.e.j.w.WebAppContext@349f3ff7{hiveserver2/,file:///C:/Users/hp/AppData/Local/Temp/jetty-0_0_0-10002-hive-service-4_0_0-jar-_any-307521053838814726/webapp/,AVAILABLE}{jar:/file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/hive-service-4.0.0.jar!/hive-webapps/hiveserver2}
2024-06-02T22:33:27,047 INFO [main] handler.ContextHandler: Started o.e.j.s.ServletContextHandler@598778cc{static,/static,jar:/file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/hive-service-4.0.0.jar!/hive-webapps/static,AVAILABLE}
2024-06-02T22:33:27,065 INFO [main] server.AbstractConnector: Started ServerConnector@671da0f9{HTTP/1.1, (http/1.1)}{0.0.0.0:10002}
2024-06-02T22:33:26,683 INFO [main] service.AbstractService: Service:HiveServer2 is started.
2024-06-02T22:33:26,688 INFO [main] server.Server: jetty-9.4.45.v20220203; built: 2022-02-03T09:14:34.105Z; git: 4a0c91c0be53805e3cffdc5cc9587d5301863db; jvm 1.8.0_411-b09
2024-06-02T22:33:26,950 INFO [main] server.session: DefaultSessionIdManager workerName=node0
2024-06-02T22:33:26,950 INFO [main] server.session: No SessionScavenger set, using defaults
2024-06-02T22:33:26,955 INFO [main] server.session: node0 Scavenging every 660000ms
2024-06-02T22:33:27,045 INFO [main] handler.ContextHandler: Started o.e.j.w.WebAppContext@349f3ff7{hiveserver2/,file:///C:/Users/hp/AppData/Local/Temp/jetty-0_0_0-10002-hive-service-4_0_0-jar-_any-307521053838814726/webapp/,AVAILABLE}{jar:/file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/hive-service-4.0.0.jar!/hive-webapps/hiveserver2}
2024-06-02T22:33:27,047 INFO [main] handler.ContextHandler: Started o.e.j.s.ServletContextHandler@598778cc{static,/static,jar:/file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/hive-service-4.0.0.jar!/hive-webapps/static,AVAILABLE}
2024-06-02T22:33:27,065 INFO [main] server.AbstractConnector: Started ServerConnector@671da0f9{HTTP/1.1, (http/1.1)}{0.0.0.0:10002}
2024-06-02T22:33:27,066 INFO [main] server.HiveServer2: Web UI has started on port 10002
2024-06-02T22:33:27,066 INFO [main] server.Server: Started @18373ms
2024-06-02T22:33:27,066 INFO [main] http.HttpServer: Started HttpServer[hiveserver2] on port 10002

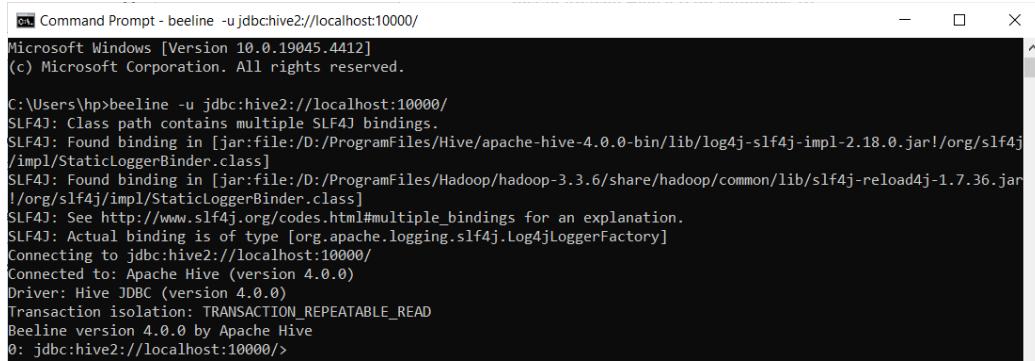
```

Here, we can see that HiveServer2 service connected to metastore running over Thrift URI `thrift://127.0.0.1:9083` and started on port 10002 and ThriftBinaryCLIService started on port 10000.

## 9.9. Run Queries in Beeline CLI:

Open new command prompt and connect remote Beeline CLI as anonymous user with this command

```
beeline -u jdbc:hive2://localhost:10000/
```



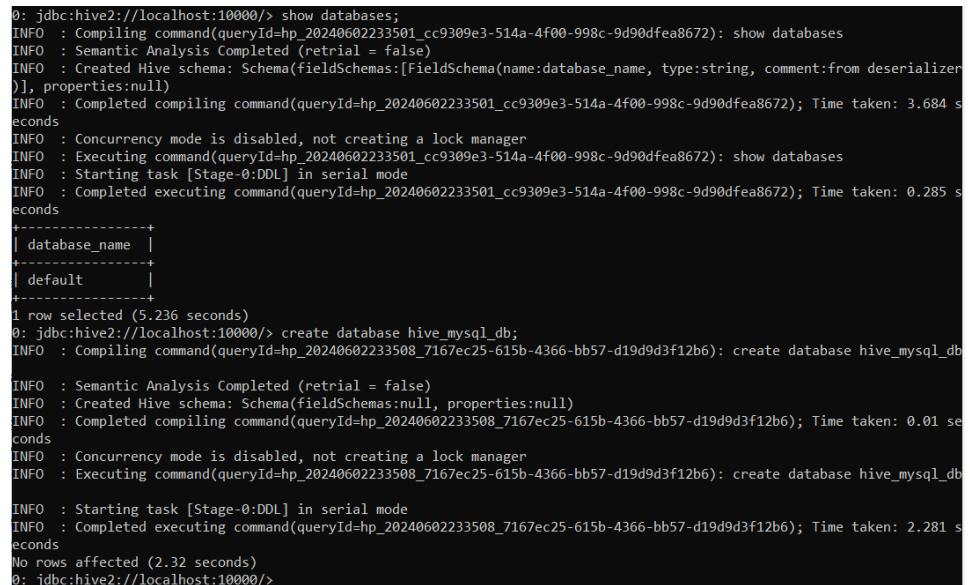
```
0: Command Prompt - beeline -u jdbc:hive2://localhost:10000/
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>beeline -u jdbc:hive2://localhost:10000/
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j
/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://localhost:10000/
Connected to: Apache Hive (version 4.0.0)
Driver: Hive JDBC (version 4.0.0)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 4.0.0 by Apache Hive
0: jdbc:hive2://localhost:10000/
```

After beeline is connected, run these queries to create a database, table and load data into it.

- Create a database named `hive_mysql_db`

```
show databases;
create database hive_mysql_db;
```



```
0: jdbc:hive2://localhost:10000/> show databases;
INFO : Compiling command(queryId=hp_20240602233501_cc9309e3-514a-4f00-998c-9d90dfa8672): show databases
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:database_name, type:string, comment:from deserializer
)], properties:null)
INFO : Completed compiling command(queryId=hp_20240602233501_cc9309e3-514a-4f00-998c-9d90dfa8672); Time taken: 3.684 s
econds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602233501_cc9309e3-514a-4f00-998c-9d90dfa8672): show databases
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240602233501_cc9309e3-514a-4f00-998c-9d90dfa8672); Time taken: 0.285 s
econds
+-----+
| database_name |
+-----+
| default      |
+-----+
1 row selected (5.236 seconds)
0: jdbc:hive2://localhost:10000/> create database hive_mysql_db;
INFO : Compiling command(queryId=hp_20240602233508_7167ec25-615b-4366-bb57-d19d9d3f12b6): create database hive_mysql_db
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldschemas:null, properties:null)
INFO : Completed compiling command(queryId=hp_20240602233508_7167ec25-615b-4366-bb57-d19d9d3f12b6); Time taken: 0.01 se
conds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602233508_7167ec25-615b-4366-bb57-d19d9d3f12b6): create database hive_mysql_db
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240602233508_7167ec25-615b-4366-bb57-d19d9d3f12b6); Time taken: 2.281 s
econds
No rows affected (2.32 seconds)
0: jdbc:hive2://localhost:10000/
```

As soon as we run the above query, Hive has created `hive_mysql_db`.db database in HDFS/`/user/hive/warehouse` location.

```
hadoop fs -ls /user/hive/warehouse
```

```

Command Prompt
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp>hadoop fs -ls /user/hive/warehouse
Found 3 items
drwxr-xr-x - hp supergroup          0 2024-06-02 19:43 /user/hive/warehouse/hive_embedded_derby_db.db
drwxr-xr-x - hp supergroup          0 2024-06-02 21:28 /user/hive/warehouse/hive_local_derby_db.db
drwxr-xr-x - hp supergroup          0 2024-06-02 23:35 /user/hive/warehouse/hive_mysql_db.db

C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_mysql_db.db

C:\Users\hp>

```

The same is visible in NameNode UI: <http://localhost:9870/dfshealth.html>

Open NameNode UI, go to **Utilities** tab and select **Browse the file system** option. Enter the directory name /user/hive/warehouse and you can see **hive\_mysql\_db.db** folder available.

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hp	supergroup	0 B	Jun 02 19:43	0	0 B	hive_embedded_derby_db.db
drwxr-xr-x	hp	supergroup	0 B	Jun 02 21:28	0	0 B	hive_local_derby_db.db
drwxr-xr-x	hp	supergroup	0 B	Jun 02 23:35	0	0 B	hive_mysql_db.db

- Create a table named employees in **hive\_mysql\_db** database. This table is created based on the columns data in the CSV file that we are going to load.

```

use hive_mysql_db;

create table employees_tmp(employee_id int, first_name string, last_name
string, email string, phone_number string, hire_date string, job_id string,
salary int, commission_pct int, manager_id int, department_id int) row format
delimited fields terminated by ','
tblproperties('skip.header.line.count'=1');

```

```

0: jdbc:hive2://localhost:10000> use hive_mysql_db;
INFO : Compiling command(queryId=hp_20240602233907_d87a5ab4-c0cd-4f77-b27c-555ec0f10023): use hive_mysql_db
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:null, properties:null)
INFO : Completed compiling command(queryId=hp_20240602233907_d87a5ab4-c0cd-4f77-b27c-555ec0f10023); Time taken: 0.081 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602233907_d87a5ab4-c0cd-4f77-b27c-555ec0f10023): use hive_mysql_db
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240602233907_d87a5ab4-c0cd-4f77-b27c-555ec0f10023); Time taken: 0.039 seconds
No rows affected (0.174 seconds)
0: jdbc:hive2://localhost:10000> create table employees_tmp(employee_id int, first_name string, last_name string, email string, phone_number string, hire_date string, job_id string, salary int, commission_pct int, manager_id int, department_id int) row format delimited fields terminated by ',' tblproperties ('skip.header.line.count'=1');
INFO : Compiling command(queryId=hp_20240602233913_8893b8f9-d44d-476c-9e15-eb2a1511c1bc): create table employees_tmp(employee_id int, first_name string, last_name string, email string, phone_number string, hire_date string, job_id string, salary int, commission_pct int, manager_id int, department_id int) row format delimited fields terminated by ',' tblproperties ('skip.header.line.count'=1')
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:null, properties:null)
INFO : Completed compiling command(queryId=hp_20240602233913_8893b8f9-d44d-476c-9e15-eb2a1511c1bc); Time taken: 0.149 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602233913_8893b8f9-d44d-476c-9e15-eb2a1511c1bc): create table employees_tmp(employee_id int, first_name string, last_name string, email string, phone_number string, hire_date string, job_id string, salary int, commission_pct int, manager_id int, department_id int) row format delimited fields terminated by ',' tblproperties ('skip.header.line.count'=1')
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240602233913_8893b8f9-d44d-476c-9e15-eb2a1511c1bc); Time taken: 1.443 seconds
No rows affected (1.626 seconds)
0: jdbc:hive2://localhost:10000>

```

On HDFS, employees\_tmp folder is created in /user/hive/warehouse/hive\_mysql\_db.db location.

```

C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_mysql_db.db
Found 1 items
drwxr-xr-x - hp supergroup          0 2024-06-02 23:39 /user/hive/warehouse/hive_mysql_db.db/employees_tmp

C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_mysql_db.db/employees_tmp

```

On NameNode UI, click on hive\_mysql\_db.db folder to see employees\_tmp folder.

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hp	supergroup	0 B	Jun 02 23:39	0	0 B	employees_tmp

- Verify the format of the employees\_tmp table created.

```
describe formatted employees_tmp;
```

```
0: jdbc:hive2://localhost:10000> describe formatted employees_tmp;
Error: Error while compiling statement: FAILED: ParseException line 1:0 cannot recognize input near 'describle' 'formatted'
ed' 'employees_tmp' (state=42000,code=40000)
0: jdbc:hive2://localhost:10000> describe formatted employees_tmp;
INFO : Compiling command(queryId=hp_20240603083811_1c535ef3-50fc-4f7e-b903-d07ef2bc5da1): describe formatted employees_
tmp
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:col_name, type:string, comment:from deserializer), Fi
eldSchema(name:data_type, type:string, comment:from deserializer), FieldSchema(name:comment, type:string, comment:from d
eserializer)], properties:null)
INFO : Completed compiling command(queryId=hp_20240603083811_1c535ef3-50fc-4f7e-b903-d07ef2bc5da1); Time taken: 0.052 s
seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240603083811_1c535ef3-50fc-4f7e-b903-d07ef2bc5da1): describe formatted employees_
tmp
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240603083811_1c535ef3-50fc-4f7e-b903-d07ef2bc5da1); Time taken: 0.112 s
seconds
+-----+-----+-----+
| col_name | data_type | comment |
+-----+-----+-----+
| employee_id | int | NULL |
| first_name | string | NULL |
| last_name | string | NULL |
| email | string | NULL |
| phone_number | string | NULL |
| hire_date | string | NULL |
| job_id | string | NULL |
| salary | int | NULL |
| commission_pct | int | NULL |
| manager_id | int | NULL |
| department_id | int | NULL |
| # Detailed Table Information | NULL | NULL |
| Database: | hive_mysql_db | NULL |
| OwnerType: | USER | NULL |
| Owner: | hp | NULL |
| CreateTime: | Sun Jun 02 23:39:14 IST 2024 | NULL |
| LastAccessTime: | UNKNOWN | NULL |
| Retention: | 0 | NULL |
| Location: | hdfs://localhost:9820/user/hive/warehouse/hive_mysql_db.db/employees_tmp | NULL |
| Table Type: | EXTERNAL_TABLE | NULL |
| Table Parameters: | NULL | NULL |
| | EXTERNAL | TRUE |
| | TRANSLATED_TO_EXTERNAL | TRUE |
| | bucketing_version | 2 |
| | external.table.purge | TRUE |
| | numFiles | 1 |
| | numRows | 0 |
| | rawDataSize | 0 |
| | skip.header.line.count | 1 |
| | totalSize | 3778 |
| | transient_lastDdlTime | 1717351971 |
| | NULL | NULL |
| | NULL | NULL |
| | org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe | NULL |
| InputFormat: | org.apache.hadoop.mapred.TextInputFormat | NULL |
| OutputFormat: | org.apache.hadoop.hive.io.HiveIgnoreKeyTextOutputFormat | NULL |
| Compressed: | No | NULL |
| Num Buckets: | -1 | NULL |
| Bucket Columns: | [] | NULL |
| Sort Columns: | [] | NULL |
| Storage Desc Params: | NULL | NULL |
| | field_delim | , |
| | serialization.format | , |
+-----+-----+-----+
44 rows selected (0.474 seconds)
0: jdbc:hive2://localhost:10000>
```

- Load data from employees.csv file located in the local directory D:\Datasets into the hive table employees\_tmp.

**Note:** This CSV file is available in [this location](#) that you can download and copy to D:\Datasets folder in your machine.

```
load data local inpath 'file:///D:\Datasets\employees.csv' into table
employees_tmp;
```

```
0: jdbc:hive2://localhost:10000> load data local inpath 'file:///D:\Datasets\employees.csv' into table employees_tmp;
INFO : Compiling command(queryId=hp_20240602234249_ea7bc217-eb92-4426-ba03-19207cbe6f2e): load data local inpath 'file:///D:\Datasets\employees.csv' into table employees_tmp
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:null, properties:null)
INFO : Completed compiling command(queryId=hp_20240602234249_ea7bc217-eb92-4426-ba03-19207cbe6f2e); Time taken: 0.161 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602234249_ea7bc217-eb92-4426-ba03-19207cbe6f2e): load data local inpath 'file:///D:\Datasets\employees.csv' into table employees_tmp
INFO : Starting task [Stage-0:MOVE] in serial mode
INFO : Loading data to table hive_mysql_db.employees_tmp from file:/D:/Datasets/employees.csv
INFO : Starting task [Stage-1:STATS] in serial mode
INFO : Executing stats task
INFO : Table hive_mysql_db.employees_tmp stats: [numFiles=1, numRows=0, totalSize=3778, rawDataSize=0, numFilesErasureEncoded=0]
INFO : Completed executing command(queryId=hp_20240602234249_ea7bc217-eb92-4426-ba03-19207cbe6f2e); Time taken: 2.123 seconds
0 rows affected (2.328 seconds)
0: jdbc:hive2://localhost:10000>
```

In HDFS, there is a `employees.csv` file created in `/user/hive/warehouse/hive_mysql_db.db/employees_tmp` location.

```
C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_mysql_db.db/employees_tmp
Found 1 items
-rw-r--r-- 1 hp supergroup 3778 2024-06-02 23:42 /user/hive/warehouse/hive_mysql_db.db/employees_tmp/employees.csv

C:\Users\hp>
```

On NameNode UI, click on `employees_tmp` folder and you can see `employees.csv` file is created.

	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
<input type="checkbox"/>	-rw-r--r--	hp	supergroup	3.69 KB	Jun 02 23:42	1	128 MB	employees.csv

- Select top 5 records in `employees_tmp` table from Hive CLI.

```
set hive.cli.print.header=true;
select * from employees_tmp limit 5;
```

```

0: jdbc:hive2://localhost:10000> set hive.cli.print.header=true;
No rows affected (0.043 seconds)
0: jdbc:hive2://localhost:10000> select * from employees_tmp limit 5;
INFO : Compiling command(queryId=hp_20240602234441_8d5c05fb-e7fe-474c-9719-3695b7232254): select * from employees_tmp l
imit 5
INFO : No Stats for hive_mysql_db@employees_tmp, Columns: commission_pct, manager_id, department_id, job_id, employee_i
d, last_name, phone_number, hire_date, salary, first_name, email
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:employees_tmp.employee_id, type:int, comment:null), F
ieldSchema(name:employees_tmp.first_name, type:string, comment:null), FieldSchema(name:employees_tmp.last_name, type:str
ing, comment:null), FieldSchema(name:employees_tmp.email, type:string, comment:null), FieldSchema(name:employees_tmp.pho
ne_number, type:string, comment:null), FieldSchema(name:employees_tmp.hire_date, type:string, comment:null), FieldSchema(
name:employees_tmp.job_id, type:string, comment:null), FieldSchema(name:employees_tmp.salary, type:int, comment:null),
FieldSchema(name:employees_tmp.commission_pct, type:int, comment:null), FieldSchema(name:employees_tmp.manager_id, type:
int, comment:null), FieldSchema(name:employees_tmp.department_id, type:int, comment:null)], properties:null)
INFO : Completed compiling command(queryId=hp_20240602234441_8d5c05fb-e7fe-474c-9719-3695b7232254); Time taken: 4.038 s
econds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602234441_8d5c05fb-e7fe-474c-9719-3695b7232254): select * from employees_tmp l
imit 5
INFO : Completed executing command(queryId=hp_20240602234441_8d5c05fb-e7fe-474c-9719-3695b7232254); Time taken: 0.001 s
econds
+-----+-----+-----+-----+
| employees_tmp.employee_id | employees_tmp.first_name | employees_tmp.last_name | employees_tmp.email | employees_t
mp.phone_number | employees_tmp.hire_date | employees_tmp.job_id | employees_tmp.salary | employees_tmp.commission_p
ct | employees_tmp.manager_id | employees_tmp.department_id |
+-----+-----+-----+-----+
| 198 | Donald | O'Connell | DOCONNEL | NULL | 650.507.983
| 124 | 50 | SH_CLERK | | 2600 | DGRANT | NULL |
| 199 | Douglas | SH_CLERK | Grant | 2600 | DWHALEN | NULL |
| 124 | 50 | Jennifer | Whalen | 4400 | MHARTSTE | NULL |
| 200 | 17-SEP-03 | AD_ASST | Hartstein | 13000 | PFAY | NULL |
| 101 | Michael | MK_MAN | | | 515.123.444
| 201 | 17-FEB-04 | 20 | Fay | 6000 | 515.123.555
| 100 | Pat | MK_REP | | | 603.123.666
| 202 | 17-AUG-05 | 20 | | | 515.123.555
| 201 | 20 | | | | |
+-----+-----+-----+-----+
5 rows selected (4.315 seconds)
0: jdbc:hive2://localhost:10000>
```

We can verify the same in HDFS using hadoop cat command.

```

hadoop fs -cat
/usr/hive/warehouse/hive_mysql_db.db/employees_tmp/employees.csv
```

```
C:\Users\hp>hadoop fs -cat /user/hive/warehouse/hive_mysql_db.db/employees_tmp/employees.csv
EMPLOYEE_ID,FIRST_NAME,LAST_NAME,EMAIL,PHONE_NUMBER,HIRE_DATE,JOB_ID,SALARY,COMMISSION_PCT,MANAGER_ID,DEPARTMENT_ID
198,Donald,OConnell,DOCONNEL,650.507.9833,21-JUN-07,SH_CLERK,2600, -,124,50
199,Douglas,Grant,DGRANT,650.507.9844,13-JAN-08,SH_CLERK,2600, -,124,50
200,Jennifer,Whalen,JWHALEN,515.123.4444,17-SEP-03,AD_ASST,4400, -,101,10
201,Michael,Hartstein,MHARTSTE,515.123.5555,17-FEB-04,MK_MAN,13000, -,100,20
202,Pat,Fay,PFAY,603.123.6666,17-AUG-05,MK_REP,6000, -,201,20
203,Susan,Mavris,SMAVRIS,515.123.7777,07-JUN-02,HR_REP,6500, -,101,40
204,Hermann,Baer,HBAER,515.123.8888,07-JUN-02,PR_REP,10000, -,101,70
205,Shelley,Higgins,SHIGGINS,515.123.8080,07-JUN-02,AC_MGR,12008, -,101,110
206,William,Gietz,WGIETZ,515.123.8181,07-JUN-02,AC_ACCOUNT,8300, -,205,110
100,Steven,King,SKING,515.123.4567,17-JUN-03,AD_PRES,24000, -, -,90
101,Neena,Kochhar,NKOCHHAR,515.123.4568,21-SEP-05,AD_VP,17000, -,100,90
102,Lex,De Haan,LDEHAAN,515.123.4569,13-JAN-01,AD_VP,17000, -,100,90
103,Alexander,Hunold,AHUNOLD,590.423.4567,03-JAN-06,IT_PROG,9000, -,102,60
104,Bruce,Ernst,BERNST,590.423.4568,21-MAY-07,IT_PROG,6000, -,103,60
105,David,Austin,DAUSTIN,590.423.4569,25-JUN-05,IT_PROG,4800, -,103,60
106,Valli,Pataballa,VPATABAL,590.423.4560,05-FEB-06,IT_PROG,4800, -,103,60
```

In the NameNode UI, click on `employees.csv` file and go to **Head the file** tab to see first few records or **Tail the file** tab to see last few records.

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
198	Donald	OConnell	DOCONNEL	650.507.9833	21-JUN-07	SH_CLERK	2600	-	124	50
199	Douglas	Grant	DGRANT	650.507.9844	13-JAN-08	SH_CLERK	2600	-	124	50
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-SEP-03	AD_ASST	4400	-	101	10
201	Michael	Hartstein	MHARTSTE	515.123.5555	17-FEB-04	MK_MAN	13000	-	100	20
202	Pat	Fay	PFAY	603.123.6666	17-AUG-05	MK_REP	6000	-	201	20
203	Susan	Mavris	SMAVRIS	515.123.7777	07-JUN-02	HR_REP	6500	-	101	40
204	Hermann	Baer	HBAER	515.123.8888	07-JUN-02	PR_REP	10000	-	101	70
205	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-02	AC_MGR	12008	-	101	110
206	William	Gietz	WGIETZ	515.123.8181	07-JUN-02	AC_ACCOUNT	8300	-	205	110
100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	24000	-	-	90
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	-	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	-	100	90
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	-	102	60
104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	-	103	60
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	-	103	60
106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	-	103	60

- Now, let us create a new table named `employees` similar to `employees_tmp` table except that data type of `hire_date` column would be `date` instead of `string`.

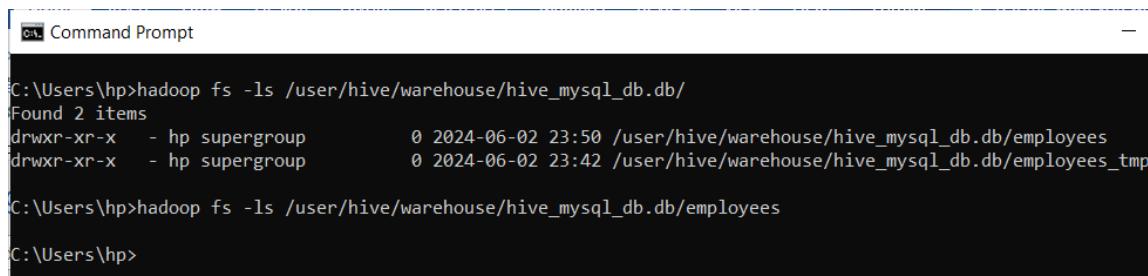
```
create table employees(employee_id int, first_name string, last_name string,
email string, phone_number string, hire_date date, job_id string, salary int,
commission_pct int, manager_id int, department_id int)
tblproperties('skip.header.line.count'=1');
```

```

0: jdbc:hive2://localhost:10000> create table employees(employee_id int, first_name string, last_name string, email string, phone_number string, hire_date date, job_id string, salary int, commission_pct int, manager_id int, department_id int)tblproperties("skip.header.line.count"="1");
INFO : Compiling command(queryId=hp_20240603085427_f936390c-3327-476a-93e8-03be269792c2): create table employees(employee_id int, first_name string, last_name string, email string, phone_number string, hire_date date, job_id string, salary int, commission_pct int, manager_id int, department_id int)tblproperties("skip.header.line.count"="1")
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldsSchemas:null, properties:null)
INFO : Completed compiling command(queryId=hp_20240603085427_f936390c-3327-476a-93e8-03be269792c2); Time taken: 0.015 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240603085427_f936390c-3327-476a-93e8-03be269792c2): create table employees(employee_id int, first_name string, last_name string, email string, phone_number string, hire_date date, job_id string, salary int, commission_pct int, manager_id int, department_id int)tblproperties("skip.header.line.count"="1")
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240603085427_f936390c-3327-476a-93e8-03be269792c2); Time taken: 0.26 seconds
No rows affected (0.315 seconds)
0: jdbc:hive2://localhost:10000>

```

On HDFS, employees folder is created in  
`/user/hive/warehouse/hive_mysql_db.db` location.



```

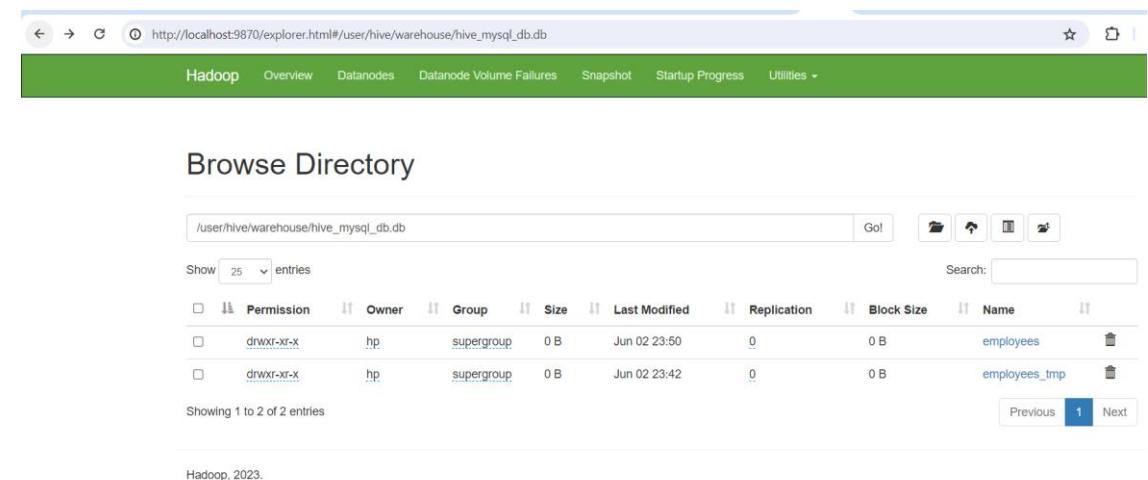
C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_mysql_db.db/
Found 2 items
drwxr-xr-x  - hp supergroup          0 2024-06-02 23:50 /user/hive/warehouse/hive_mysql_db.db/employees
drwxr-xr-x  - hp supergroup          0 2024-06-02 23:42 /user/hive/warehouse/hive_mysql_db.db/employees_tmp

C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_mysql_db.db/employees

C:\Users\hp>

```

On NameNode UI, go to `/user/hive/warehouse/hive_mysql_db.db` folder to see employees folder.



	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hp	supergroup	0 B	Jun 02 23:50	0	0 B	employees	
drwxr-xr-x	hp	supergroup	0 B	Jun 02 23:42	0	0 B	employees_tmp	

- Load data into employees table from employees\_tmp table by transforming hire\_date column in employees\_tmp table into the Hive accepted date format.

```

insert into employees select employee_id, first_name, last_name, email,
phone_number, from_unixtime(unix_timestamp(hire_date,'dd-MMM-yy'), 'yyyy-MM-

```

```
dd') as hire_date, job_id, salary, commission_pct, manager_id, department_id
from employees_tmp;
```

```
0: jdbc:hive2://localhost:10000> insert into employees select employee_id, first_name, last_name, email, phone_number, from_unixtime(unix_timestamp(hire_date,'dd-MMM-yy'), 'yyyy-MM-dd') as hire_date, job_id, salary, commission_pct, manager_id, department_id from employees_tmp;
INFO : Compiling command(queryId=hp_20240602235347_2081c26a-bad9-4115-8c59-aeed40b89a3e): insert into employees select employee_id, first_name, last_name, email, phone_number, from_unixtime(unix_timestamp(hire_date,'dd-MMM-yy'), 'yyyy-MM-dd') as hire_date, job_id, salary, commission_pct, manager_id, department_id from employees_tmp
INFO : No Stats for hive_mysql_db@employees_tmp, Columns: commission_pct, manager_id, department_id, job_id, employee_id, last_name, phone_number, hire_date, salary, first_name, email
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:_col0, type:int, comment:null), FieldSchema(name:_col1, type:string, comment:null), FieldSchema(name:_col2, type:string, comment:null), FieldSchema(name:_col3, type:string, comment:null), FieldSchema(name:_col4, type:string, comment:null), FieldSchema(name:_col5, type:date, comment:null), FieldSchema(name:_col6, type:string, comment:null), FieldSchema(name:_col7, type:int, comment:null), FieldSchema(name:_col8, type:int, comment:null), FieldSchema(name:_col9, type:int, comment:null), FieldSchema(name:_col10, type:int, comment:null)], properties:null)
INFO : Completed compiling command(queryId=hp_20240602235347_2081c26a-bad9-4115-8c59-aeed40b89a3e); Time taken: 1.61 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240602235347_2081c26a-bad9-4115-8c59-aeed40b89a3e): insert into employees select employee_id, first_name, last_name, email, phone_number, from_unixtime(unix_timestamp(hire_date,'dd-MMM-yy'), 'yyyy-MM-dd') as hire_date, job_id, salary, commission_pct, manager_id, department_id from employees_tmp
WARN : Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. tez) or using Hive 1.X releases.
INFO : Query ID = hp_20240602235347_2081c26a-bad9-4115-8c59-aeed40b89a3e
INFO : Total jobs = 3
INFO : Launching Job 1 out of 3
INFO : Starting task [Stage-1:MAPRED] in serial mode
INFO : Number of reduce tasks determined at compile time: 1
INFO : In order to change the average load for a reducer (in bytes):
INFO :   set hive.exec.reducers.bytes.per.reducer=<number>
INFO : In order to limit the maximum number of reducers:
INFO :   set hive.exec.reducers.max=<number>
INFO : In order to set a constant number of reducers:
INFO :   set mapreduce.job.reduces=<number>
INFO : number of splits:1
INFO : Submitting tokens for job: job_1717349440214_0001
INFO : Executing with tokens: []
INFO : The url to track the job: http://DESKTOP-KGH2E2G:8088/proxy/application_1717349440214_0001/
INFO : Starting Job = job_1717349440214_0001, Tracking URL = http://DESKTOP-KGH2E2G:8088/proxy/application_1717349440214_0001/
INFO : Kill Command = D:\ProgramFiles\Hadoop\hadoop-3.3.6\bin\mapred job -kill job_1717349440214_0001
INFO : Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
INFO : 2024-06-02 23:54:17,177 Stage-1 map = 0%, reduce = 0%
INFO : 2024-06-02 23:54:28,873 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 6.514 sec
INFO : 2024-06-02 23:54:44,697 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 16.434 sec
INFO : MapReduce Total cumulative CPU time: 16 seconds 434 msec
INFO : Ended Job = job_1717349440214_0001
INFO : Starting task [Stage-7:CONDITIONAL] in serial mode
INFO : Stage-4 is selected by condition resolver.
INFO : Stage-3 is filtered out by condition resolver.
INFO : Stage-5 is filtered out by condition resolver.
INFO : Starting task [Stage-4:MOVE] in serial mode
INFO : Moving data to directory hdfs://localhost:9820/user/hive/warehouse/hive_mysql_db.db/employees/.hive-staging_hive_2024-06-02_23-53-47_068_2888876849100680296-1/-ext-10000
INFO : Starting task [Stage-0:MOVE] in serial mode
INFO : Loading data to table hive_mysql_db.employees from hdfs://localhost:9820/user/hive/warehouse/hive_mysql_db.db/employees/.hive-staging_hive_2024-06-02_23-53-47_068_2888876849100680296-1/-ext-10000
INFO : Starting task [Stage-2:STATS] in serial mode
INFO : Executing stats task
INFO : Table hive_mysql_db.employees stats: [numFiles=1, numRows=51, totalSize=3726, rawDataSize=3675, numFilesErasureCoded=0]
INFO : MapReduce Jobs Launched:
INFO : Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 16.434 sec HDFS Read: 65633 HDFS Write: 7841 HDFS EC Read: 0 SUCCESS
INFO : Total MapReduce CPU Time Spent: 16 seconds 434 msec
INFO : Completed executing command(queryId=hp_20240602235347_2081c26a-bad9-4115-8c59-aeed40b89a3e); Time taken: 59.85 seconds
51 rows affected (61.543 seconds)
0: jdbc:hive2://localhost:10000/
```

The above query submitted a MapReduce job to Hadoop YARN which we can track using the application tracking URL provided.

Open Hadoop YARN UI: <http://localhost:8088/cluster> where we can see the application has been submitted which is running the application “*insert into employees .. from employees\_tmp (Stage-1)*”

The screenshot shows the Apache Hadoop Cluster UI at <http://localhost:8088/cluster>. The 'All Applications' tab is selected. On the left, there's a sidebar with 'Cluster Metrics', 'Nodes Metrics', 'Scheduler Metrics', and 'Tools'. The main area displays a table of applications. One row is highlighted with a red box, showing the application ID 'application\_1717349440214\_0001' submitted by 'hp' with the name 'Insert into employees,...rom employees\_tmp (Stage-1)'. The application type is 'MAPREDUCE' and it has tags like 'hp\_20240602235347\_2081c26a-ba99-4115-8c59-aed40b8a3e', 'userid=anonymous', and 'default'. The application was submitted on Sun Jun 2 23:53:56 +0550 2024 and finished on Sun Jun 2 23:53:59 +0550 2024.

After the above query is completed, we can see a file named `000000_0` created in `/user/hive/warehouse/hive_mysql_db.db/employees` HDFS location.

```
C:\Users\hp>hadoop fs -ls /user/hive/warehouse/hive_mysql_db.db/employees
Found 1 items
-rw-r--r-- 1 hp supergroup      3726 2024-06-02 23:54 /user/hive/warehouse/hive_mysql_db.db/employees/000000_0
C:\Users\hp>
```

On NameNode UI, click on `employees` folder to see `000000_0` file.

http://localhost:9870/explorer.html#/user/hive/warehouse/hive\_mysql\_db.db/employees. It shows a 'Browse Directory' interface with a table of files. The file '000000\_0' is highlighted with a red box in the 'Name' column."/>

The screenshot shows the Hadoop NameNode UI at [http://localhost:9870/explorer.html#/user/hive/warehouse/hive\\_mysql\\_db.db/employees](http://localhost:9870/explorer.html#/user/hive/warehouse/hive_mysql_db.db/employees). The top navigation bar includes 'Hadoop', 'Overview', 'Datanodes', 'Datanode Volume Failures', 'Snapshot', 'Startup Progress', and 'Utilities'. Below, a 'Browse Directory' section shows a table of files under '/user/hive/warehouse/hive\_mysql\_db.db/employees'. The table columns are: Permission, Owner, Group, Size, Last Modified, Replication, Block Size, and Name. A single file, '000000\_0', is listed with the following details: -rw-r--r--, hp, supergroup, 3.64 KB, Jun 02 23:54, 1, 128 MB, and '000000\_0'. The 'Name' column for this file is highlighted with a red box.

- Select top 5 records in `employees` table in Hive CLI.

```
select * from employees limit 5;
```

```

0: jdbc:hive2://localhost:10000> select * from employees limit 5;
INFO : Compiling command(queryId=hp_20240603085819_84cb37ad-1486-448c-b970-3250fb91b9cb): select * from employees limit 5
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:employees.employee_id, type:int, comment:null), FieldSchema(name:employees.first_name, type:string, comment:null), FieldSchema(name:employees.last_name, type:string, comment:null), FieldSchema(name:employees.email, type:string, comment:null), FieldSchema(name:employees.phone_number, type:string, comment:null), FieldSchema(name:employees.hire_date, type:int, comment:null), FieldSchema(name:employees.job_id, type:string, comment:null), FieldSchema(name:employees.commission_pct, type:int, comment:null), FieldSchema(name:employees.salary, type:int, comment:null), FieldSchema(name:employees.department_id, type:int, comment:null)], properties:null)
INFO : Completed compiling command(queryId=hp_20240603085819_84cb37ad-1486-448c-b970-3250fb91b9cb); Time taken: 0.315 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240603085819_84cb37ad-1486-448c-b970-3250fb91b9cb): select * from employees limit 5
INFO : Completed executing command(queryId=hp_20240603085819_84cb37ad-1486-448c-b970-3250fb91b9cb); Time taken: 0.001 seconds
+-----+-----+-----+-----+-----+-----+
| employees.employee_id | employees.first_name | employees.last_name | employees.email | employees.phone_number | employees.hire_date |
| employees.job_id | employees.salary | employees.commission_pct | employees.manager_id | employees.department_id |
+-----+-----+-----+-----+-----+-----+
| 198 | Donald | OConnell | DOCONNEL | 650.507.9833 | 2007-06-21 | |
| 199 | SH_CLERK | 2600 | NULL | DGRANT | 650.507.9844 | 2008-01-13 |
| 200 | Douglas | Grant | 124 | 50 | 124 | 50 |
| 201 | SH_CLERK | 2600 | NULL | JWHALEN | 515.123.4444 | 2003-09-17 |
| 202 | Jennifer | Whalen | 101 | 10 | 101 | 10 |
| 203 | AD_ASST | 4400 | NULL | MHARTSTE | 515.123.5555 | 2004-02-17 |
| 204 | Michael | Hartstein | 100 | 20 | 100 | 20 |
| 205 | MK_MAN | 13000 | NULL | PFAY | 603.123.6666 | 2005-08-17 |
| 206 | Pat | Fay | 201 | 20 | 201 | 20 |
+-----+-----+-----+-----+-----+-----+
5 rows selected (0.403 seconds)
0: jdbc:hive2://localhost:10000/>
```

We can verify the same in HDFS using hadoop cat command.

```
hadoop fs -cat /user/hive/warehouse/hive_mysql_db.db/employees/000000_0
```

```
Found 1 items
-rw-r--r-- 1 hp supergroup      3726 2024-06-02 23:54 /user/hive/warehouse/hive_mysql_db.db/employees/000000_0

C:\Users\hp\hadoop fs -cat /user/hive/warehouse/hive_mysql_db.db/employees/000000_0
\N0FIRST_NAME\LAST_NAME\EMAIL\PHONE_NUMBER\NJOB_ID\N\N\N
1980Donald@O'Connell@DOCONNEL@650.507.9833@2007-06-21@SH_CLERK@2600@\N@124050
1990Douglas@Grant@GRANT0650.507.9844@2008-01-13@SH_CLERK@2600@\N@124050
2000Jennifer@Whalen@JWHALEN@515.123.4444@2003-09-17@AD_ASST@4400@\N@101010
2010Michael@Hartstein@MHARTSTE@515.123.5555@2004-02-17@MK_MAN@13000@\N@100020
2020Patricia@Fay@PFAY@603.123.6666@2005-08-17@MK_REP@60000@\N@201020
2030Susan@Mavris@SMARVIS@515.123.7777@2002-06-07@HR_REP@6500@\N@101040
2040Hermann@Baer@HBAER@515.123.8888@2002-06-07@PR_REP@10000@\N@101070
2050Shelley@Higgins@SHIGGINS@515.123.8080@2002-06-07@AC_MGR@12008@\N@1010110
2060William@Gietz@WGIETZ@515.123.8181@2002-06-07@AC_ACCOUNT@8300@\N@2050110
```

In the NameNode UI, click on `000000_0` file and go to **Head** the file tab to see first few records.

- Verify if the count of records between employees\_tmp and employees tables are matching.

#### **Count records in employees\_tmp table:**

```
select count(*) from employees_tmp;
```

```
0: jdbc:hive2://localhost:10000/ select count(*) from employees_tmp;
INFO : Compiling command(queryId=hp_20240603000857_607aaabc-442d-4c4e-ab3a-48e20e5fd651): select count(*) from employees_tmp
INFO : Semantic Analysis Completed (retry = false)
INFO : Created Hive schema: Schema(fieldschemas:[FieldSchema(name:_c0, type:bigint, comment:null)]), properties:null
INFO : Completed compiling command(queryId=hp_20240603000857_607aaabc-442d-4c4e-ab3a-48e20e5fd651); Time taken: 0.332 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240603000857_607aaabc-442d-4c4e-ab3a-48e20e5fd651): select count(*) from employees_tmp
WARN : Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. tez) or using Hive 1.X releases.
INFO : Query ID = hp_20240603000857_607aaabc-442d-4c4e-ab3a-48e20e5fd651
INFO : Total jobs = 1
INFO : Launching Job 1 out of 1
INFO : Starting task [Stage-1:MAPRED] in serial mode
INFO : Number of reduce tasks determined at compile time: 1
INFO : In order to change the average load for a reducer (in bytes):
INFO :   set hive.exec.reducers.bytes.per.reducer=<number>
INFO : In order to limit the maximum number of reducers:
INFO :   set hive.exec.reducers.max=<number>
INFO : In order to set a constant number of reducers:
INFO :   set mapreduce.job.reduces=<number>
INFO : number of splits:1
INFO : Submitting tokens for job: job_1717349440214_0002
INFO : Executing with tokens: []
INFO : The url to track the job: http://DESKTOP-KGH2E2G:8088/proxy/application_1717349440214_0002/
INFO : Starting Job = job_1717349440214_0002, tracking URL = http://DESKTOP-KGH2E2G:8088/proxy/application_1717349440214_0002
INFO : Kill Command = D:\Program Files\Hadoop\hadoop-3.3.6\bin\mapred job -kill job_1717349440214_0002
INFO : Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
INFO : 2024-06-03 00:09:21.198 Stage-1 map = 0%, reduce = 0%
INFO : 2024-06-03 00:09:32.931 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 6.247 sec
INFO : 2024-06-03 00:09:47.571 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 13.244 sec
INFO : MapReduce Total cumulative CPU time: 13 seconds 244 msec
INFO : Ended Job = job_1717349440214_0002
INFO : MapReduce Jobs Launched:
INFO : Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 13.244 sec HDFS Read: 20486 HDFS Write: 102 HDFS EC Read: 0 SUCCESS
INFO : Total MapReduce CPU Time Spent: 13 seconds 244 msec
[ _c0
  51
]
1 row selected (52.793 seconds)
INFO : Completed executing command(queryId=hp_20240603000857_607aaabc-442d-4c4e-ab3a-48e20e5fd651); Time taken: 52.374 seconds
0: jdbc:hive2://localhost:10000/
```

It displayed the count as 51. Note that the actual records in the table are 50 but it is counting header also as one record and displaying 51.

The above query submitted another job to Hadoop YARN and provided us a tracking URL which we can see in YARN UI.

ID	User	Command	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime
application_1717349440214_0002	hp	select count(*) from employeesTmp (Stage-1)	MAPREDUCE	hp_20240603000857_607aaabc-442d-4c4e-ab3a-48e20ef651.userid=anonymous	default	0	Mon Jun 3 00:09:04 +0550 2024	Mon Jun 3 00:09:05 +0550 2024	Mon Jun 3 00:09:48 +0550 2024
application_1717349440214_0001	hp	insert into employees.....rom employeesTmp (Stage-1)	MAPREDUCE	hp_20240602235347_2081c26a-bad9-4115-8c59-aee40b89a3e.userid=anonymous	default	0	Sun Jun 2 23:53:56 +0550 2024	Sun Jun 2 23:53:59 +0550 2024	Sun Jun 2 23:54:45 +0550 2024

### **Count records in employees table:**

```
select count(*) from employees;
```

```
9. jdbc:hive2://localhost:10000/> select count(*) from employees;
INFO : Compiling command[queryId=hp_20240603001137_20553304-724f-4503-8234-3ee892a607ae]: select count(*) from employees
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema[fieldSchemas:[FieldSchema(name:_c0, type:bigint, comment:null)], properties:null]
INFO : Completed compiling command[queryId=hp_20240603001137_20553304-724f-4503-8234-3ee892a607ae]; Time taken: 0.285 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command[queryId=hp_20240603001137_20553304-724f-4503-8234-3ee892a607ae]: select count(*) from employees
WARN : Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. tez) or using Hive 1.X releases.
INFO : Query ID = hp_20240603001137_20553304-724f-4503-8234-3ee892a607ae
INFO : Total jobs = 1
INFO : Launching Job 1 out of 1
INFO : Starting task [Stage-1:MAPRED] in serial mode
INFO : Number of reduce tasks determined at compile time: 1
INFO : In order to change the average load for a reducer (in bytes):
INFO :   set hive.exec.reducers.bytes.per.reducer=<number>
INFO : In order to limit the maximum number of reducers:
INFO :   set hive.exec.reducers.max=<number>
INFO : In order to set a constant number of reducers:
INFO :   set mapreduce.job.reduces=<number>
INFO : number of splits:1
INFO : Submitting tokens for job: job_1717349440214_0003
INFO : Executing with tokens: []
INFO : The url to track the job: http://DESKTOP-KGH2E2G:8088/proxy/application_1717349440214_0003/
INFO : Starting Job = job_1717349440214_0003, Tracking URL: http://DESKTOP-KGH2E2G:8088/proxy/application_1717349440214_0003
INFO : Kill Command = D:\Program Files\Hadoop\hadoop-3.3.6\bin\mapred job -kill job_1717349440214_0003
INFO : Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
INFO : 2024-06-03 00:11:54,948 Stage-1 map = 0%, reduce = 0%
INFO : 2024-06-03 00:12:05,368 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 6.233 sec
INFO : 2024-06-03 00:12:16,807 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 12.56 sec
INFO : MapReduce Total cumulative CPU time: 12 seconds 560 msec
INFO : Ended Job = job_1717349440214_0003
INFO : MapReduce Jobs Launched:
INFO : Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 12.56 sec HDFS Read: 21001 HDFS Write: 102 HDFS EC Read: 0 SUCCESS
INFO : Total MapReduce CPU Time Spent: 12 seconds 560 msec
INFO : Completed executing command(queryId=hive_20240603001137_20553304-724f-4503-8234-3ee892a607ae); Time taken: 41.305 seconds
-----+
| _c0 |
| 51 |
-----+
1 row selected (41.664 seconds)
9. jdbc:hive2://localhost:10000/>
```

It displayed the count as 51. Note that the actual records in the table are 50 but it is counting header also as one record and displaying 51.

The above query submitted another job to Hadoop YARN and provided us a tracking URL which we can see in YARN UI. It displayed the count as 51.

ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime
application_1717349440214_0003	hp	select count(*) from employees (Stage-1)	MAPREDUCE	hp_20240603001137_20553304-724f4503-8234-3ee892a607ae.userid=anonymous	default	0	Mon Jun 3 00:11:40 +0550 2024	Mon Jun 3 00:11:40 +0550 2024	Mon Jun 3 00:12:17 +0550 2024
application_1717349440214_0002	hp	select count(*) from employees_tmp (Stage-1)	MAPREDUCE	hp_20240603000857_607aaabc-442d-4c4e-ab3a-48e20e5f651.userid=anonymous	default	0	Mon Jun 3 00:09:04 +0550 2024	Mon Jun 3 00:09:05 +0550 2024	Mon Jun 3 00:09:48 +0550 2024
application_1717349440214_0001	hp	insert into employees.....rom employees_tmp (Stage-1)	MAPREDUCE	hp_20240602235347_2081c26a-bad9-4115-8c59-aeed40b85a3e.userid=anonymous	default	0	Sun Jun 2 23:53:56 +0550 2024	Sun Jun 2 23:53:59 +0550 2024	Sun Jun 2 23:54:45 +0550 2024

## 9.10. Verify Metadata in MySQL:

Let us connect to MySQL server and verify Hive metadata of database and tables created in MySQL.

Launch MySQL command prompt using this command. Note that we are connecting with `hive` user credentials that we created earlier.

```
mysql -h localhost -u hive -p
```

Provide the password of `hive` user when asked.

In `mysql>` prompt, run the following queries:

- Get the metadata of `hive_mysql_db` database.

```
use hive_metastore;
select * from dbs;
```

```
mysql> use hive_metastore;
Database changed
mysql> select * from dbs;
+-----+-----+-----+-----+-----+-----+-----+-----+
| DB_ID | DESC          | DB_LOCATION_URI   | NAME      | OWNER_NAME |
| OWNER_TYPE | CTLG_NAME | CREATE_TIME | DB_MANAGED_LOCATION_URI | TYPE    | DATACONNECTOR_NAME | REMOTE_DBNAME |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | Default Hive database | hdfs://localhost:9820/user/hive/warehouse | default | public
| ROLE | hive | 1717346459 | NULL | NATIVE | NULL | NULL |
| 31 | NULL | hdfs://localhost:9820/user/hive/warehouse/hive_mysql_db.db | hive_mysql_db | hp |
| USER | hive | 1717351510 | NULL | NATIVE | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

It shows the DB location URI as

```
hdfs://localhost:9820/user/hive/warehouse/hive mysql db.db.
```

- Get the metadata of employees tmp and employees tables and its columns.

```
select * from tb1s;  
select * from columns_v2;
```

```

mysql> select * from tbls;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| TBL_ID | CREATE_TIME | DB_ID | LAST_ACCESS_TIME | OWNER | OWNER_TYPE | RETENTION | SD_ID | TBL_NAME | TBL_TYPE
| VIEW_EXPANDED_TEXT | VIEW_ORIGINAL_TEXT | IS_REWRITE_ENABLED |          |          |          |          |          |          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1717351754 | 31 | 0 | hp | USER | 0 | 1 | employees_tmp | EXTERNAL TA
BLE | NULL | NULL | 0x00 |          |          |          |          |          |
| 2 | 1717352413 | 31 | 0 | hp | USER | 0 | 2 | employees | EXTERNAL TA
BLE | NULL | NULL | 0x00 |          |          |          |          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from columns_v2;
+-----+-----+-----+-----+-----+
| CD_ID | COMMENT | COLUMN_NAME | TYPE_NAME | INTEGER_IDX |
+-----+-----+-----+-----+-----+
| 1 | NULL | commission_pct | int | 8 |
| 1 | NULL | department_id | int | 10 |
| 1 | NULL | email | string | 3 |
| 1 | NULL | employee_id | int | 0 |
| 1 | NULL | first_name | string | 1 |
| 1 | NULL | hire_date | string | 5 |
| 1 | NULL | job_id | string | 6 |
| 1 | NULL | last_name | string | 2 |
| 1 | NULL | manager_id | int | 9 |
| 1 | NULL | phone_number | string | 4 |
| 1 | NULL | salary | int | 7 |
+-----+-----+-----+-----+-----+
| 2 | NULL | commission_pct | int | 8 |
| 2 | NULL | department_id | int | 10 |
| 2 | NULL | email | string | 3 |
| 2 | NULL | employee_id | int | 0 |
| 2 | NULL | first_name | string | 1 |
| 2 | NULL | hire_date | date | 5 |
| 2 | NULL | job_id | string | 6 |
| 2 | NULL | last_name | string | 2 |
| 2 | NULL | manager_id | int | 9 |
| 2 | NULL | phone_number | string | 4 |
| 2 | NULL | salary | int | 7 |
+-----+-----+-----+-----+-----+
22 rows in set (0.00 sec)

mysql>
```

It shows employees tmp and employees tables and their respective columns.

## 10. Hive Web UI:

Hive provides the following web interface to monitor HiveServer2 service.

HiveServer2 UI: <http://localhost:10002/>

The screenshot shows the Hive Web UI at <http://localhost:10002>. The top navigation bar includes links for Home, Local logs, Metrics Dump, Hive Configuration (which is highlighted), Stack Trace, Log Daemons, and Configure logging. The main content area is titled "HiveServer2".

### Active Sessions

User Name	IP Address	Operation Count	Active Time (s)	Idle Time (s)
anonymous	127.0.0.1	0	2740	482

Total number of sessions: 1

### Open Queries

User Name	Query	Execution Engine	State	Opened Timestamp	Opened (s)	Latency (s)	Drilldown Link
Total number of queries: 0							

### Last Max 25 Closed Queries

User Name	Query	Execution Engine	State	Opened (s)	Closed Timestamp	Latency (s)	Drilldown Link
anonymous	select count(*) from employees	mr	FINISHED	41	Mon Jun 03 00:12:19 IST	41	<a href="#">Drilldown</a>

Since we last connected to Beeline as anonymous user, it shows the **User Name** as anonymous.

It also displays the last 25 queries executed on HiveServer2 after the latest restart.

We can also view Hive Configuration, Stack Trace, Configure logging and much more in this UI.

## 11. HCatalog and WebHCat:

**HCatalog** is a table and storage management layer for Hadoop that allows **MapReduce**, **Pig** and **Hive** users to easily read and write data on HDFS. HCatalog provides a relational view of data stored in HDFS. It is built on top of the Hive metastore and incorporates Hive's DDL. It provides read and write interfaces for Pig and MapReduce and uses Hive's command line interface for issuing data definition and metadata exploration commands.

**WebHCat** (previously called as **Templeton**) is the REST API provided to access HCatalog service. Unlike HCatalog, which executes the command directly, WebHCat keeps the Hive, PIG, and MapReduce jobs in queues. The jobs can then be monitored and stopped as needed.

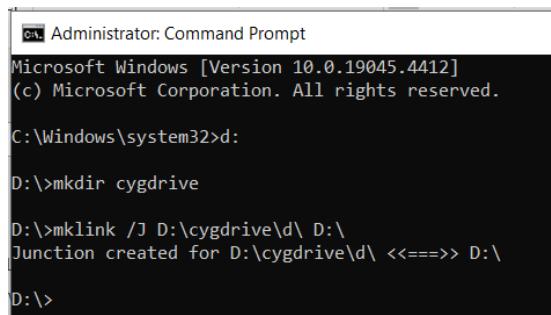
**Note:** We cannot start **HCatalog** and **WebHCat** server from Windows since they are not compatible for Windows and so, we will use Cygwin utility which allows executing Linux commands from Windows.

### 11.1. Create Symbolic Link for Cygwin:

Since Java cannot understand Cygwin paths properly, we will first create symbolic links for cygdrive to use Cygwin utility.

Open command prompt in Administrator mode and run the following commands:

```
d:  
mkdir cygdrive  
mklink /J D:\cygdrive\d\ D:\
```



```
Administrator: Command Prompt  
Microsoft Windows [Version 10.0.19045.4412]  
(c) Microsoft Corporation. All rights reserved.  
C:\Windows\system32>d:  
D:\>mkdir cygdrive  
D:\>mklink /J D:\cygdrive\d\ D:\  
Junction created for D:\cygdrive\d\ <=====> D:\  
D:\>
```

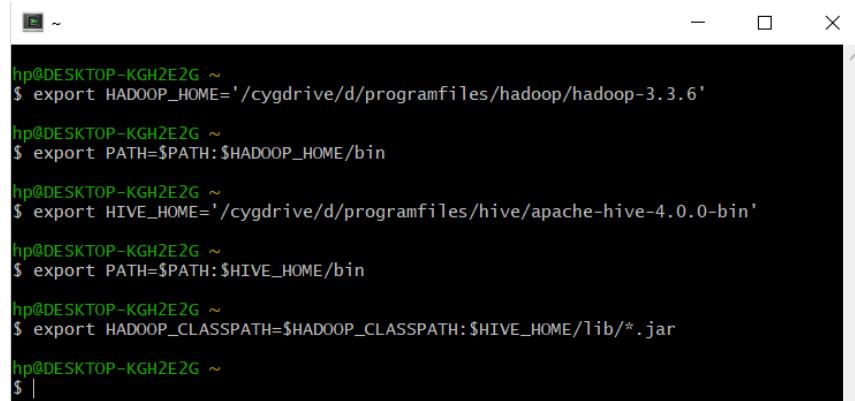
Note that we created `cygdrive` directory in `D` drive since we installed Hive in this drive. If you have installed Hive in different drive then create symbolic link to that drive.

### 11.2. Setup Env variables for Cygwin:

Now, open **Cygwin64 Terminal** and run the following commands to define environment variables. We can add the below lines to `~/.bashrc` file so we don't need to execute every time we open Cygwin.

```
export HADOOP_HOME='/cygdrive/d/programfiles/hadoop/hadoop-3.3.6'  
export PATH=$PATH:$HADOOP_HOME/bin  
export HIVE_HOME='/cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin'  
export PATH=$PATH:$HIVE_HOME/bin  
export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:$HIVE_HOME/lib/*.jar
```

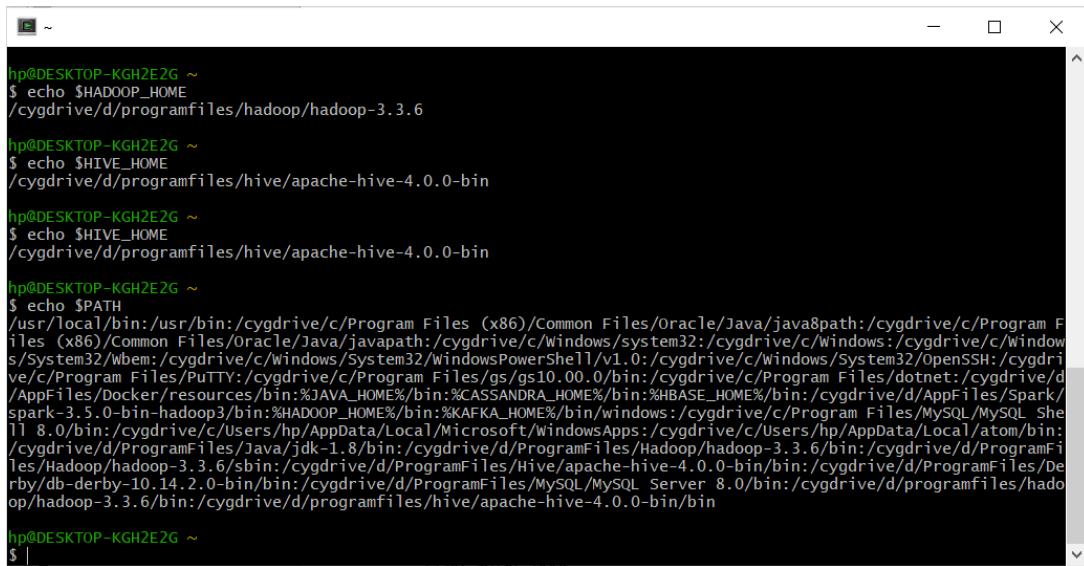
Make sure the above `HADOOP_HOME` and `HIVE_HOME` variables refer to the exact location where Hadoop and Hive were installed.



```
hp@DESKTOP-KGH2E2G ~
$ export HADOOP_HOME='/cygdrive/d/programfiles/hadoop/hadoop-3.3.6'
hp@DESKTOP-KGH2E2G ~
$ export PATH=$PATH:$HADOOP_HOME/bin
hp@DESKTOP-KGH2E2G ~
$ export HIVE_HOME='/cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin'
hp@DESKTOP-KGH2E2G ~
$ export PATH=$PATH:$HIVE_HOME/bin
hp@DESKTOP-KGH2E2G ~
$ export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:$HIVE_HOME/lib/*.jar
hp@DESKTOP-KGH2E2G ~
$ |
```

Verify if the above variables are properly set with these commands

```
echo $HADOOP_HOME
echo $HIVE_HOME
echo $HADOOP_CLASSPATH
echo $PATH
```



```
hp@DESKTOP-KGH2E2G ~
$ echo $HADOOP_HOME
/cygdrive/d/programfiles/hadoop/hadoop-3.3.6
hp@DESKTOP-KGH2E2G ~
$ echo $HIVE_HOME
/cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin
hp@DESKTOP-KGH2E2G ~
$ echo $HADOOP_CLASSPATH
$PATH
/usr/local/bin:/usr/bin:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/java8path:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/javapath:/cygdrive/c/windows/system32:/cygdrive/c/windows:/cygdrive/c/windows/System32/wbem:/cygdrive/c/windows/System32/windowsPowerShell/v1.0:/cygdrive/c/windows/System32/OpenSSH:/cygdrive/c/Program Files/PuTTY:/cygdrive/c/Program Files/gs/gs10.00.0/bin:/cygdrive/c/Program Files/dotnet:/cygdrive/d/AppFiles/docker/resources/bin:%JAVA_HOME%/bin:%CASSANDRA_HOME%/bin:%HBASE_HOME%/bin:/cygdrive/d/AppFiles/Spark/spark-3.5.0-bin-hadoop3/bin:%HADOOP_HOME%/bin:%KAFKA_HOME%/bin/windows:/cygdrive/c/Program Files/MySQL/MySQL Shell 8.0/bin:/cygdrive/c/Users/hp/AppData/Local/Microsoft/WindowsApps:/cygdrive/c/Users/hp/AppData/Local/atom/bin:/cygdrive/d/ProgramFiles/Java/jdk-1.8/bin:/cygdrive/d/ProgramFiles/Hadoop/hadoop-3.3.6/bin:/cygdrive/d/ProgramFiles/Hadoop/hadoop-3.3.6/sbin:/cygdrive/d/ProgramFiles/Hive/apache-hive-4.0.0-bin/bin:/cygdrive/d/ProgramFiles/derby/db-derby-10.14.2.0-bin/bin:/cygdrive/d/ProgramFiles/MySQL/MySQL Server 8.0/bin:/cygdrive/d/programfiles/hadoop/hadoop-3.3.6/bin:/cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/bin
hp@DESKTOP-KGH2E2G ~
$ |
```

### 11.3. Start HCatalog CLI:

We can start the HCatalog command line interface by using `hcat` file available in `HIVE_HOME\hcatalog\bin`

In Cygwin, run the following commands to start HCatalog

```
cd $HIVE_HOME/hcatalog/bin
./hcat
```

```

/cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/hcatalog/bin
hp@DESKTOP-KGH2E2G ~
$ cd $HIVE_HOME/hcatalog/bin
hp@DESKTOP-KGH2E2G /cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/hcatalog/bin
$ ./hcat
which: no hbase in (/usr/local/bin:/usr/bin:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/javapath:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/javapath:/cygdrive/c/Windows/system32:/cygdrive/c/Windows:/cygdrive/c/windows/System32/wbem:/cygdrive/c/Windows/System32/WindowsPowerShell/v1.0:/cygdrive/c/windows/System32/openSSH:/cygdrive/c/Program Files/PuTTY:/cygdrive/c/Program Files/gs/gs10.00.0/bin:/cygdrive/c/Program Files/dotnet:/cygdrive/d/AppFiles/Do
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2
.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j
-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.SLF4J: Actual bindin
g is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = 681efad-3132-41df-858b-07e01c2aded6
usage: hcat { -e "<query>" | -f "<filepath>" } [ -g "<group>" ] [ -p "<perms>" ] [ -D"<name>=<value>
" ]
-D <property=value>      use hadoop value for given property
-e <execs>                hcat command given from command line
-f <file>                 hcat commands in file
-g <group>                group for the db/table specified in CREATE statement
-h,--help                  Print help information
-p <perms>                permissions for the db/table specified in CREATE statement
hp@DESKTOP-KGH2E2G /cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/hcatalog/bin
$ |

```

Note that **hcat** commands can be issued as **hive** commands and HCatalog CLI supports the following command line options:

**-g** : Tells HCatalog that the table which needs to be created must have group `mygroup`.

For example:

```
hcat -g mygroup ...
```

**-p** : Tells HCatalog that the table which needs to be created must have permissions

`"rwxr-xr-x"`.

For example:

```
hcat -p rwxr-xr-x ...
```

**-f** : Tells HCatalog that `myscript.hcatalog` is a file containing DDL commands to execute.

For example:

```
hcat -f myscript.hcatalog ...
```

**-e** : Tells HCatalog to treat the following string as a DDL command and execute it

For example:

```
hcat -e 'create table mytable(a int);' ...
```

**-D** : Passes the key-value pair to HCatalog as a Java System Property.

For example:

```
hcat -Dkey=value ...
```

**hcat**: Prints a usage message.

## For example:

Run the following hcat command to create employee\_sample table in hive\_mysql\_db database:

```
./hcat -e 'create table hive_mysql_db.employee_sample(emp_id int, emp_name string)'
```

```
hp@DESKTOP-KGH2E2G /cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/hcatalog/bin
$ ./hcat -e 'create table hive_mysql_db.employee_sample(emp_id int, emp_name string)'
which: no hbase in (/usr/local/bin:/usr/bin:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/java8path:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/javapath:/cygdrive/c/Windows/system32:/cygdrive/c/Windows/System32/OpenSSH:/cygdrive/c/Program Files/PuTTY:/cygdrive/c/Program Files/gs10.00.0/bin:/cygdrive/c/Program Files/dotnet:/cygdrive/d/AppFiles/Docker/resources/bin:%JAVA_HOME%/bin:%CASSANDRA_HOME%/bin:%HBASE_HOME%/bin:/cygdrive/d/AppFiles/Spark/spark-3.5.0-bin-hadoop3/bin:%HADOOP_HOME%/bin:%KAFKA_HOME%/bin/windows:/cygdrive/c/Program Files/MySQL/MySQL Shell 8.0/bin:/cygdrive/c/Users/hp/AppData/Local/Microsoft/WindowsApps:/cygdrive/c/Users/hp/AppData/Local/atom/bin:/cygdrive/d/ProgramFiles/Java/jdk-1.8/bin:/cygdrive/d/ProgramFiles/Hadoop/hadoop-3.3.6/bin:/cygdrive/d/ProgramFiles/Hadoop/hadoop-3.3.6/sbin:/cygdrive/d/Programfiles/Hive/apache-hive-4.0.0-bin/bin:/cygdrive/d/ProgramFiles/Derby/db-derby-10.14.2.0-bin/bin:/cygdrive/d/ProgramFiles/MySQL/MySQL Server 8.0/bin:/cygdrive/d/Programfiles/hadoop/hadoop-3.3.6/bin:/cygdrive/d/Programfiles/hive/apache-hive-4.0.0-bin/bin)
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hive/apache-hive-4.0.0-bin/lib/log4j-slf4j-impl-2.18.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/D:/ProgramFiles/Hadoop/hadoop-3.3.6/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = 212622d3-cc65-40fe-a489-e08d20be5c51
Time taken: 5.393 seconds
hp@DESKTOP-KGH2E2G /cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/hcatalog/bin
```

We can see the newly created table in Beeline CLI by executing these queries.

```
use hive_mysql_db;
show tables;
```

```
0: jdbc:hive2://localhost:10000> use hive_mysql_db;
INFO : Compiling command(queryId=hp_20240603003050_ab267c66-6698-44a7-9793-7175871f6c85): use hive_mysql_db
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:null, properties:null)
INFO : Completed compiling command(queryId=hp_20240603003050_ab267c66-6698-44a7-9793-7175871f6c85); Time taken: 0.018 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240603003050_ab267c66-6698-44a7-9793-7175871f6c85): use hive_mysql_db
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240603003050_ab267c66-6698-44a7-9793-7175871f6c85); Time taken: 0.029 seconds
No rows affected (0.098 seconds)
0: jdbc:hive2://localhost:10000> show tables;
INFO : Compiling command(queryId=hp_20240603003052_9beb5a1c-a923-4329-9c03-8be4e9be4f14): show tables
INFO : Semantic Analysis Completed (retrial = false)
INFO : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:tab_name, type:string, comment:from deserializer)], properties:nul
)
INFO : Completed compiling command(queryId=hp_20240603003052_9beb5a1c-a923-4329-9c03-8be4e9be4f14); Time taken: 0.033 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=hp_20240603003052_9beb5a1c-a923-4329-9c03-8be4e9be4f14): show tables
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hp_20240603003052_9beb5a1c-a923-4329-9c03-8be4e9be4f14); Time taken: 0.054 seconds
+-----+
| tab_name |
+-----+
| employee_sample |
| employees |
| employees_tmp |
+-----+
3 rows selected (0.245 seconds)
0: jdbc:hive2://localhost:10000>
```

#### 11.4. Start WebHCat server:

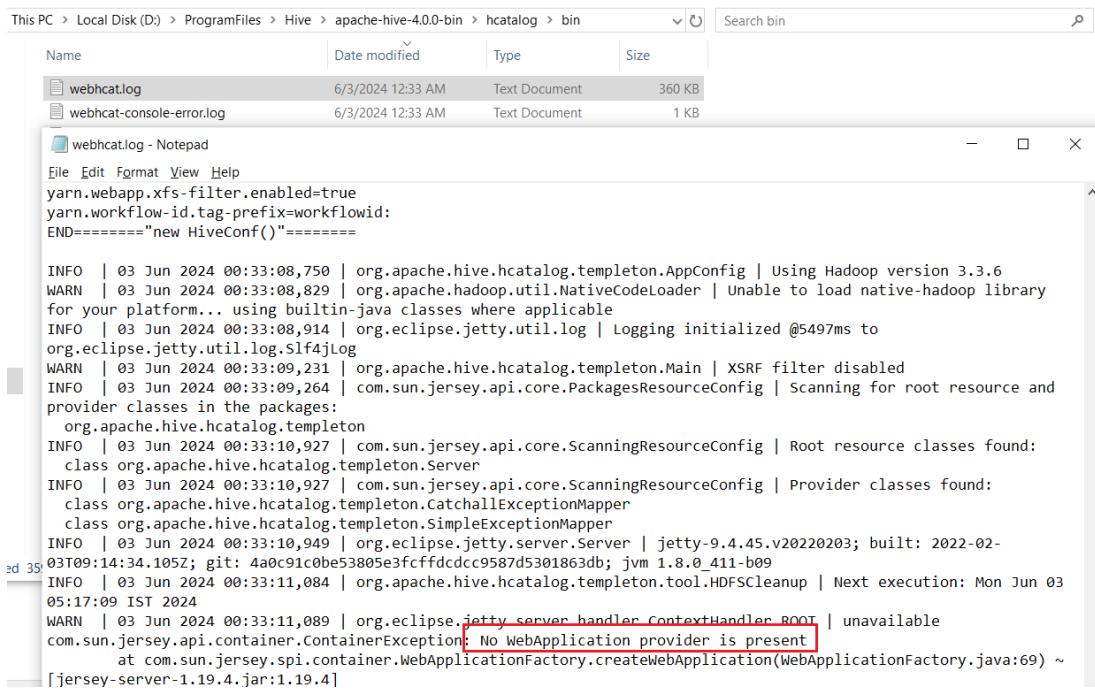
Now, start the **WebHCat** server using the following command in Cygwin:

```
$HIVE_HOME/hcatalog/sbin/webhcatt_server.sh start
```

```
$ $HIVE_HOME/hcatalog/sbin/webhcatt_server.sh start
Length of string is non zero
which: no hbase in (/usr/local/bin:/usr/bin:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/java8path:/cygdrive/c/Program Files (x86)/Common Files/Oracle/Java/javapath:/cygdrive/c/Windows/system32:/cygdrive/c/Windows/System32/OpenSSH:/cygdrive/c/Program Files/PUTTY:/cygdrive/c/Program Files/gs/gs10.00.0/bin:/cygdrive/c/Program Files/dotnet:/cygdrive/d/AppFiles/docker/resources/bin:%JAVA_HOME%/bin:%CASSANDRA_HOME%/bin:%HBASE_HOME%/bin:/cygdrive/d/AppFiles/Spark/spark-3.5.0-bin-hadoop3/bin:%KAFKA_HOME%/bin/windows:/cygdrive/c/Program Files/MySQL/MySQL Shell 8.0/bin:/cygdrive/c/Users/hp/AppData/Local/Microsoft/WindowsApps:/cygdrive/c/Users/hp/AppData/Local/atom/bin:/cygdrive/d/ProgramFiles/Java/jdk-1.8/bin:/cygdrive/d/ProgramFiles/Hadoop/hadoop-3.3.6/bin:/cygdrive/d/ProgramFiles/Hadoop/hadoop-3.3.6/sbin:/cygdrive/d/ProgramFiles/Hive/apache-hive-4.0.0-bin/bin:/cygdrive/d/ProgramFiles/Derby/db-derby-10.14.2.0-bin/bin:/cygdrive/d/ProgramFiles/MySQL/MySQL Server 8.0/bin:/cygdrive/d/programfiles/hadoop/hadoop-3.3.6/bin:/cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/bin)
webhcatt: starting ...
webhcatt: /cygdrive/d/programfiles/hadoop/hadoop-3.3.6/bin/hadoop jar /cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/hcatalog/sbin/../share/webhcatt/srv/lib/hive-webhcatt-4.0.0.jar org.apache.hive.hcatalog.templeton.Main
webhcatt: starting ... started.
webhcatt: done

hp@DESKTOP-KGH2E2G /cygdrive/d/programfiles/hive/apache-hive-4.0.0-bin/hcatalog/bin
$ |
```

Though it says “webchat started” on console, but in `webchat.log` (available in the location from where ever `webhcatt_server.sh` script was executed), there is an error “**Server failed to start: No WebApplication provider is present**”.



Name	Date modified	Type	Size
webhcatt.log	6/3/2024 12:33 AM	Text Document	360 KB
webhcatt-console-error.log	6/3/2024 12:33 AM	Text Document	1 KB

File Edit Format View Help  
yarn.webapp.xfs-filter.enabled=true  
yarn.workflow-id.tag-prefix=workflowid:  
END===== "new HiveConf()"=====

INFO | 03 Jun 2024 00:33:08,750 | org.apache.hive.hcatalog.templeton.AppConfig | Using Hadoop version 3.3.6  
WARN | 03 Jun 2024 00:33:08,829 | org.apache.hadoop.util.NativeCodeLoader | Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
INFO | 03 Jun 2024 00:33:08,914 | org.eclipse.jetty.util.log | Logging initialized @5497ms to org.eclipse.jetty.util.log.Slf4jLog  
WARN | 03 Jun 2024 00:33:09,231 | org.apache.hive.hcatalog.templeton.Main | XSRF filter disabled  
INFO | 03 Jun 2024 00:33:09,264 | com.sun.jersey.api.core.PackagesResourceConfig | Scanning for root resource and provider classes in the packages:  
org.apache.hive.hcatalog.templeton  
INFO | 03 Jun 2024 00:33:10,927 | com.sun.jersey.api.core.ScanningResourceConfig | Root resource classes found:  
class org.apache.hive.hcatalog.templeton.Server  
INFO | 03 Jun 2024 00:33:10,927 | com.sun.jersey.api.core.ScanningResourceConfig | Provider classes found:  
class org.apache.hive.hcatalog.templeton.CatchallExceptionMapper  
class org.apache.hive.hcatalog.templeton.SimpleExceptionMapper  
INFO | 03 Jun 2024 00:33:10,949 | org.eclipse.jetty.server.Server | jetty-9.4.45.v20220203; built: 2022-02-03T09:14:34.105Z; git: 4a0c91c0be53805e3fcffddcc9587d5301863db; jvm 1.8.0\_411-b09  
INFO | 03 Jun 2024 00:33:11,084 | org.apache.hive.hcatalog.templeton.tool.HDFScleanup | Next execution: Mon Jun 03 05:17:09 IST 2024  
WARN | 03 Jun 2024 00:33:11,089 | org.eclipse.jetty.server.handler.ContextHandler ROOT | unavailable com.sun.jersey.api.container.ContainerException: No WebApplication provider is present  
at com.sun.jersey.spi.container.WebApplicationFactory.createWebApplication(WebApplicationFactory.java:69) ~[jersey-server-1.19.4.jar:1.19.4]

This is because some configuration is needed before starting WebHCat server. Follow the [Apache Hive documentation](#) on how to setup WebHCat.

Once WebHCat service is successfully running, HCatlog resources can be accessed by REST APIs using the URI format: <http://localhost:50111/templeton/v1/<resource>> (For example: <http://localhost:50111/templeton/v1/status>)

**Congratulations!! You have now successfully installed and configured Hive 4.x with 3 metastore modes in Windows operating system. You also got a glimpse of HCatlog and WebHCat components.**