Impala

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com

About the Tutorial

Impala is the open source, native analytic database for Apache Hadoop. It is shipped by vendors such as Cloudera, MapR, Oracle, and Amazon. The examples provided in this tutorial have been developing using Cloudera Impala.

Audience

This tutorial is intended for those who want to learn Impala. Impala is used to process huge volumes of data at lightning-fast speed using traditional SQL knowledge.

Prerequisites

To make the most of this tutorial, you should have a good understanding of the basics of Hadoop and HDFS commands. It is also recommended to have a basic knowledge of SQL before going through this tutorial.

Copyright & Disclaimer

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

TABLE SPECIFIC STATEMENTS 43

IMPALA – CLAUSES 81

Impala – Introduction

1. IMPALA – OVERVIEW

What is Impala?

Impala is a MPP (Massive Parallel Processing) SQL query engine for processing huge volumes of data that is stored in Hadoop cluster. It is an open source software which is written in C++ and Java. It provides high performance and low latency compared to other SQL engines for Hadoop.

In other words, Impala is the highest performing SQL engine (giving RDBMS-like experience) which provides the fastest way to access data that is stored in Hadoop Distributed File System.

Why Impala?

Impala combines the SQL support and multi-user performance of a traditional analytic database with the scalability and flexibility of Apache Hadoop, by utilizing standard components such as HDFS, HBase, Metastore, YARN, and Sentry.

- With Impala, users can communicate with HDFS or HBase using SQL queries in a faster way compared to other SQL engines like Hive.

- Impala can read almost all the file formats such as Parquet, Avro, RCFile used by Hadoop.

Impala uses the same metadata, SQL syntax (Hive SQL), ODBC driver, and user interface (Hue Beeswax) as Apache Hive, providing a familiar and unified platform for batch-oriented or real-time queries.

Unlike Apache Hive, **Impala is not based on MapReduce algorithms**. It implements a distributed architecture based on **daemon processes** that are responsible for all the aspects of query execution that run on the same machines.

Thus, it reduces the latency of utilizing MapReduce and this makes Impala faster than Apache Hive.

Advantages of Impala

Here is a list of some noted advantages of Cloudera Impala.

- Using impala, you can process data that is stored in HDFS at lightning-fast speed with traditional SQL knowledge.

- Since the data processing is carried where the data resides (on Hadoop cluster), data transformation and data movement is not required for data stored on Hadoop, while working with Impala.

- Using Impala, you can access the data that is stored in HDFS, HBase, and Amazon s3 without the knowledge of Java (MapReduce jobs). You can access them with a basic idea of SQL queries.

- To write queries in business tools, the data has to be gone through a complicated extract-transform-load (ETL) cycle. But, with Impala, this procedure is shortened. The time-consuming stages of loading & reorganizing is overcome with the new techniques such as ***exploratory data analysis* & *data discovery*** making the process faster.

- Impala is pioneering the use of the Parquet file format, a columnar storage layout that is optimized for large-scale queries typical in data warehouse scenarios.

Features of Impala

Given below are the features of cloudera Impala:

- Impala is available freely as open source under the Apache license.

- Impala supports in-memory data processing, i.e., it accesses/analyzes data that is stored on Hadoop data nodes without data movement.

- You can access data using Impala using SQL-like queries.

- Impala provides faster access for the data in HDFS when compared to other SQL engines.

- Using Impala, you can store data in storage systems like HDFS, Apache HBase, and Amazon s3.

- You can integrate Impala with business intelligence tools like Tableau, Pentaho, Micro strategy, and Zoom data.

- Impala supports various file formats such as, LZO, Sequence File, Avro, RCFile, and Parquet.

- Impala uses metadata, ODBC driver, and SQL syntax from Apache Hive.

Relational Databases and Impala

Impala uses a Query language that is similar to SQL and HiveQL. The following table describes some of the key dfferences between SQL and Impala Query language.

Impala	Relational databases
	Relational databases use SQL language.

Impala uses an SQL like query language that is similar to HiveQL.	
In Impala, you cannot update or delete individual records.	In relational databases, it is possible to update or delete individual records.
Impala does not support transactions.	Relational databases support transactions.
Impala does not support indexing.	Relational databases support indexing.
Impala stores and manages large amounts of data (petabytes).	Relational databases handle smaller amounts of data (terabytes) when compared to Impala.

Hive, Hbase, and Impala

Though Cloudera Impala uses the same query language, metastore, and the user interface as Hive, it differs with Hive and HBase in certain aspects. The following table presents a comparative analysis among HBase, Hive, and Impala.

HBase	Hive	Impala
HBase is wide-column store database based on Apache Hadoop. It uses the concepts of BigTable.	Hive is a data warehouse software. Using this, we can access and manage large distributed datasets, built on Hadoop.	Impala is a tool to manage, analyze data that is stored on Hadoop.
The data model of HBase is wide column store.	Hive follows Relational model.	Impala follows Relational model.
HBase is developed using Java language.	Hive is developed using Java language.	Impala is developed using C++.
The data model of HBase is schema-free.	The data model of Hive is Schema-based.	The data model of Impala is Schema-based.
HBase provides Java, RESTful and, Thrift API's.	Hive provides JDBC, ODBC, Thrift API's.	Impala provides JDBC and ODBC API's.

Supports programming languages like C, C#, C++, Groovy, Java PHP, Python, and Scala.	Supports programming languages like C++, Java, PHP, and Python.	Impala supports all languages supporting JDBC/ODBC.
HBase provides support for triggers.	Hive does not provide any support for triggers.	Impala does not provide any support for triggers.

All these three databases –

- Are NOSQL databases.
- Available as open source.
- Support server-side scripting.
- Follow ACID properties like Durability and Concurrency.
- Use **sharding** for **partitioning**.

Drawbacks of Impala

Some of the drawbacks of using Impala are as follows:

- Impala does not provide any support for Serialization and Deserialization.
- Impala can only read text files, not custom binary files.
- Whenever new records / files are added to the data directory in HDFS, the table needs to be refreshed.

2. IMPALA – ENVIRONMENT

This chapter explains the prerequisites for installing Impala, how to download, install and set up **Impala** in your system.

Similar to Hadoop and its ecosystem software, we need to install Impala on Linux operating system. Since cloudera shipped Impala, it is available with **Cloudera Quick Start VM.**

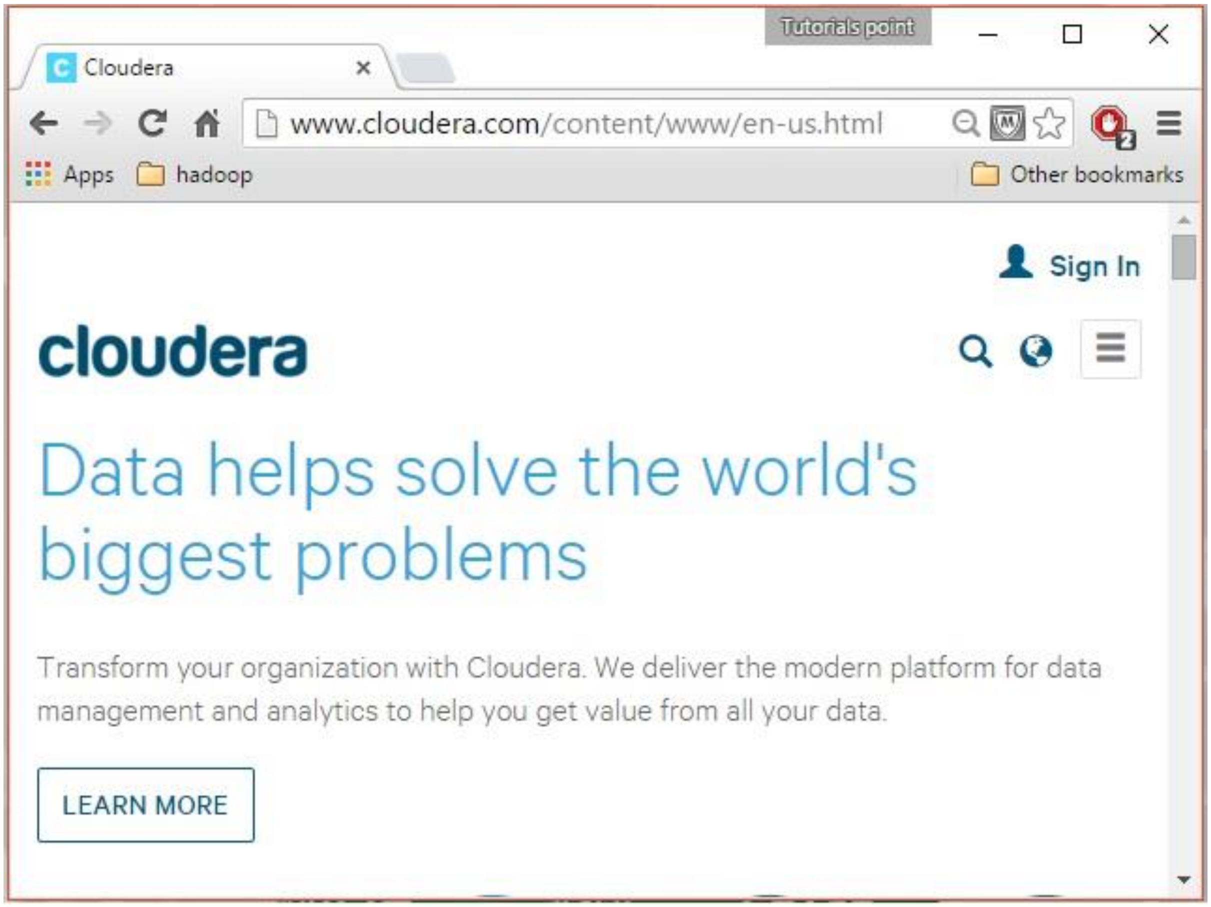
This chapter describes how to download **Cloudera Quick Start VM** and start Impala.

Downloading Cloudera Quick Start VM

Follow the steps given below to download the latest version of **Cloudera QuickStartVM**.

Step 1

Open the homepage of cloudera website **http://www.cloudera.com/.** You will get the page as shown below.

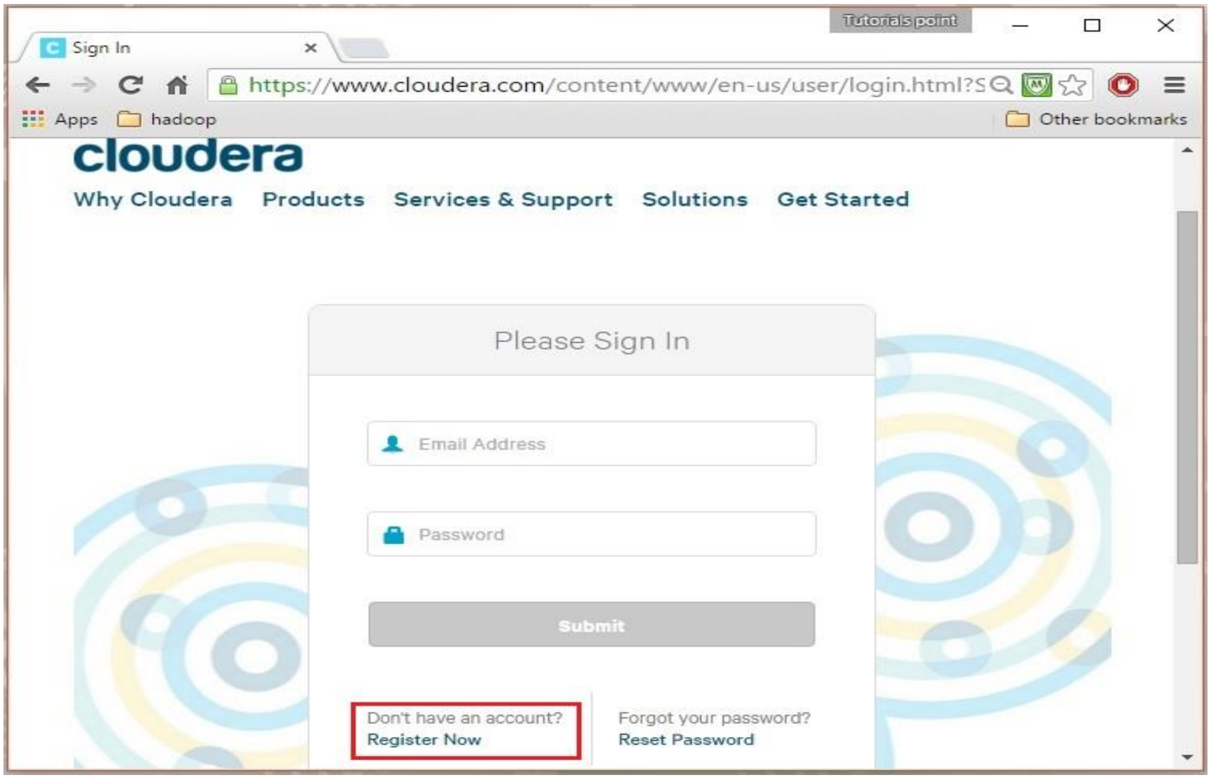


Data helps solve the world's biggest problems

Transform your organization with Cloudera. We deliver the modern platform for data management and analytics to help you get value from all your data.

Step 2

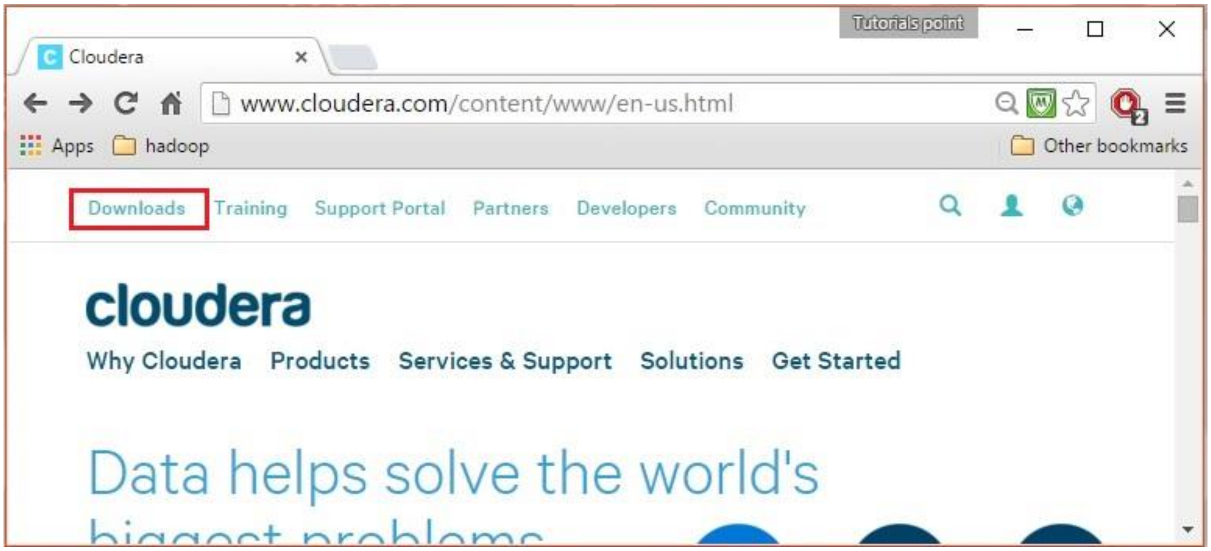
Click the **Sign in** link on the cloudera homepage, which will redirect you to the Sign in page as shown below.



If you haven't registered yet, click the **Register Now** link which will give you **Account Registration** form. Register there and sign in to cloudera account.

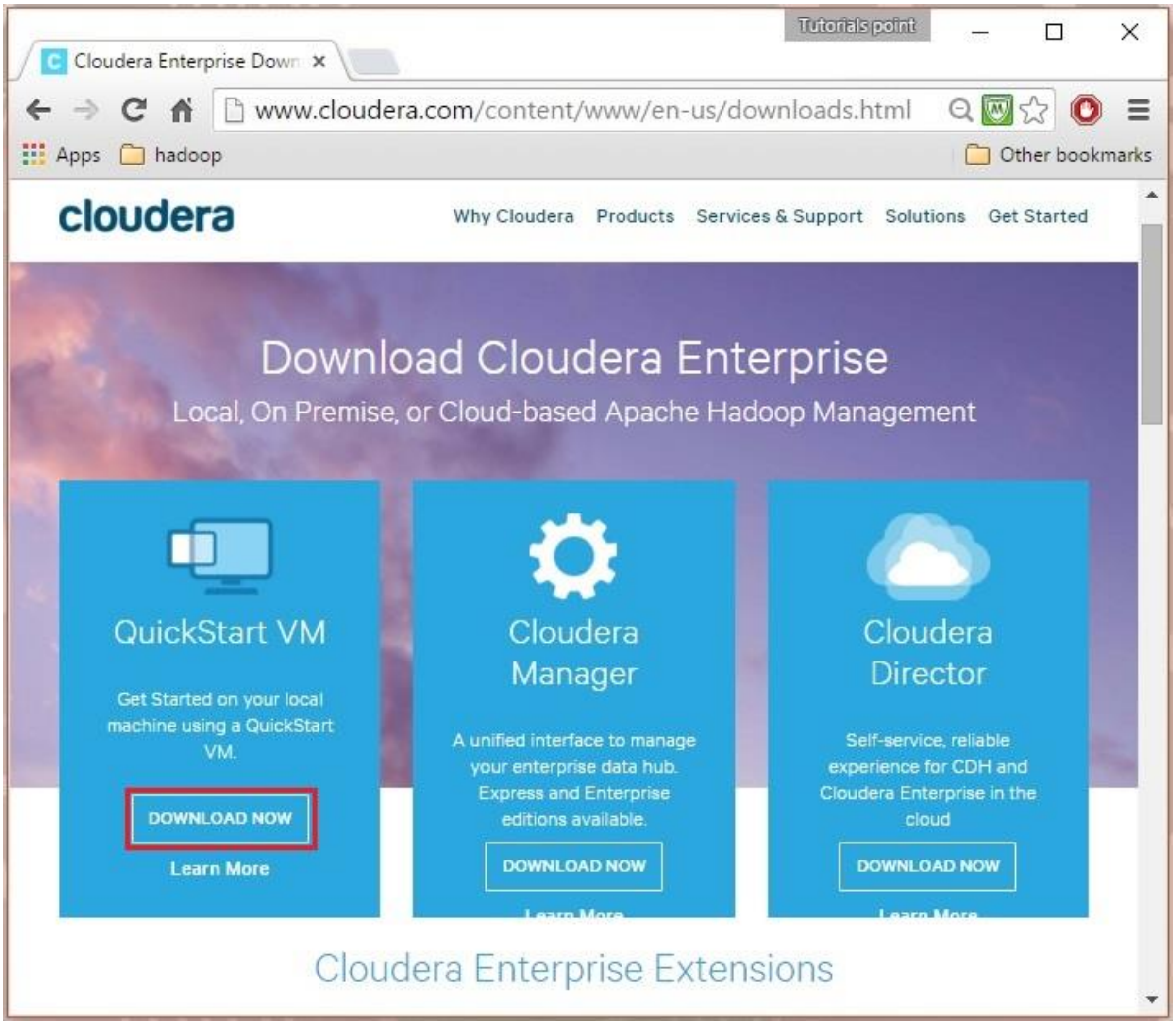
Step 3

After signing in, open the download page of cloudera website by clicking on the **Downloads** link highlighted in the following snapshot.



Step 4: Download QuickStartVM

Download the cloudera **QuickStartVM** by clicking on the **Download Now** button, as highlighted in the following snapshot.



Download Cloudera Enterprise

Local, On Premise, or Cloud-based Apache Hadoop Management

QuickStart VM

Get Started on your local machine using a QuickStart VM.

Cloudera Manager

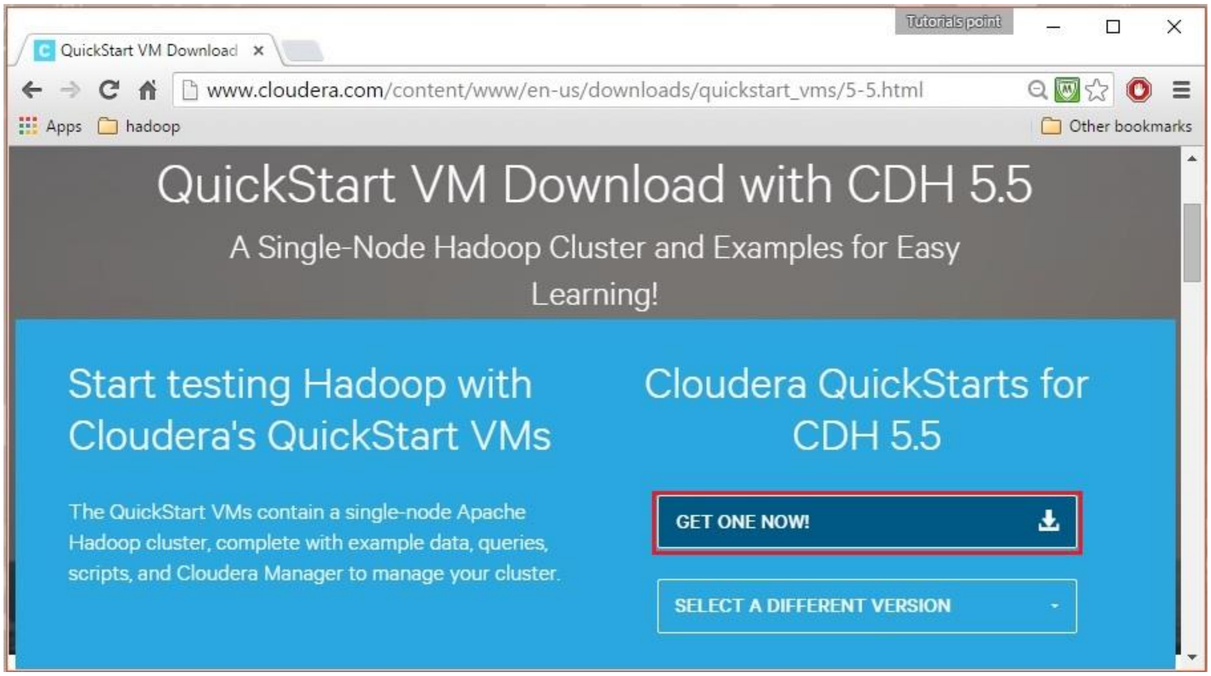
A unified interface to manage your enterprise data hub. Express and Enterprise editions available.

Cloudera Director

Self-service, reliable experience for CDH and Cloudera Enterprise in the cloud

Cloudera Enterprise Extensions

This will redirect you to the download page of **QuickStart VM**.


Tutorials point
QuickStart VM Download
www.cloudera.com/content/www/en-us/downloads/quickstart_vms/5-5.html
Apps
hadoop
Other bookmarks
GET ONE NOW!
SELECT A DIFFERENT VERSION

QuickStart VM Download with CDH 5.5

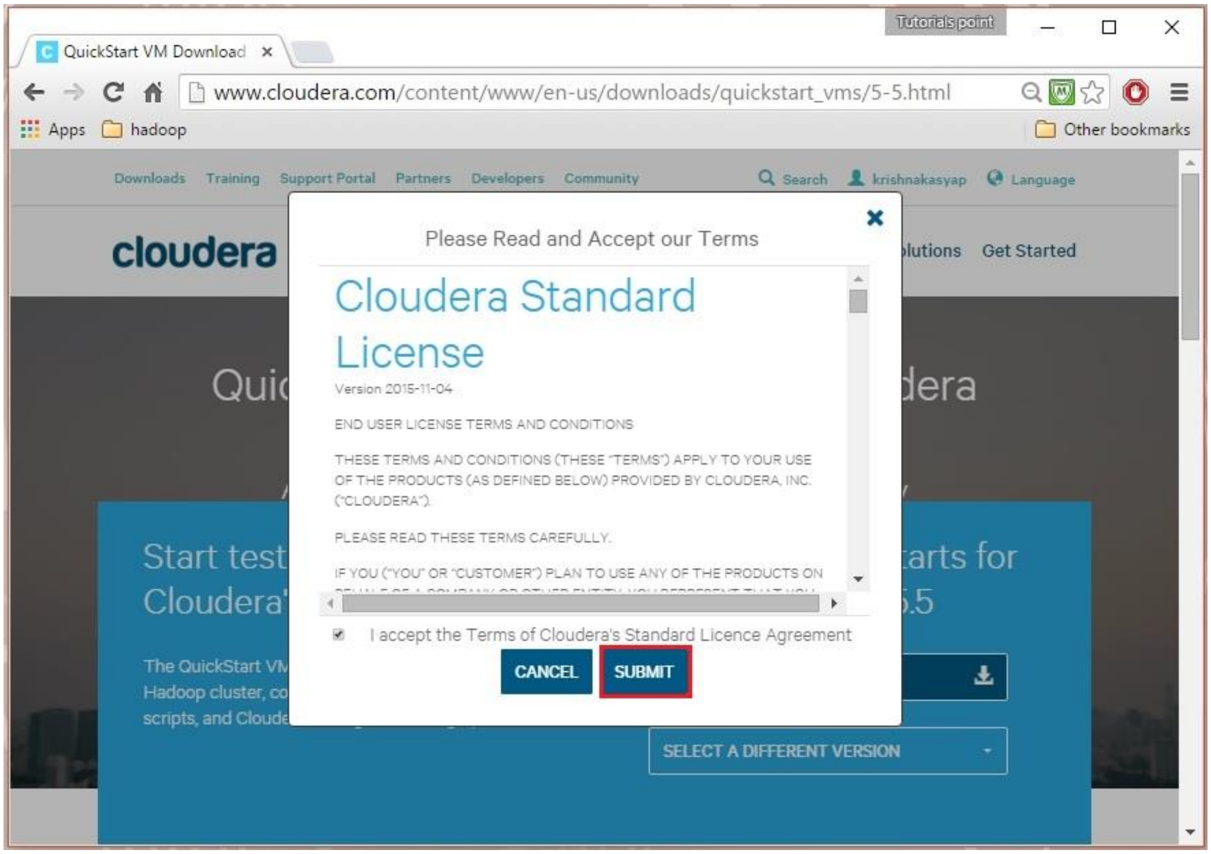
A Single-Node Hadoop Cluster and Examples for Easy Learning!

Start testing Hadoop with Cloudera's QuickStart VMs

The QuickStart VMs contain a single-node Apache Hadoop cluster, complete with example data, queries, scripts, and Cloudera Manager to manage your cluster.

Cloudera QuickStarts for CDH 5.5

Click the **Get ONE NOW** button, accept the license agreement, and click the submit button as shown below.



Please Read and Accept our Terms

Cloudera Standard License

Version 2015-11-04

END USER LICENSE TERMS AND CONDITIONS

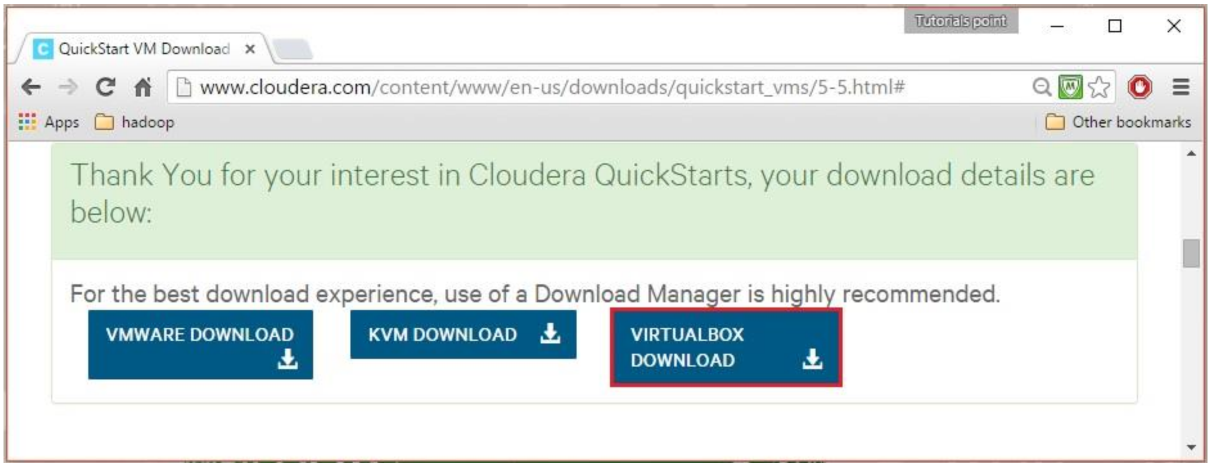
THESE TERMS AND CONDITIONS (THESE "TERMS") APPLY TO YOUR USE OF THE PRODUCTS (AS DEFINED BELOW) PROVIDED BY CLOUDERA, INC. ("CLOUDERA").

PLEASE READ THESE TERMS CAREFULLY.

IF YOU ("YOU" OR "CUSTOMER") PLAN TO USE ANY OF THE PRODUCTS ON

☑ I accept the Terms of Cloudera's Standard Licence Agreement

Cloudera provides its VM compatible VMware, KVM and VIRTUALBOX. Select the required version. Here in our tutorial, we are demonstrating the **Cloudera QuickStartVM** setup using virtual box, therefore click the **VIRTUALBOX DOWNLOAD** button, as shown in the snapshot given below.



Thank You for your interest in Cloudera QuickStarts, your download details are below:

For the best download experience, use of a Download Manager is highly recommended.

This will start downloading a file named **cloudera-quickstart-vm-5.5.0-0-virtualbox.ovf** which is a virtual box image file.

Importing the Cloudera QuickStartVM

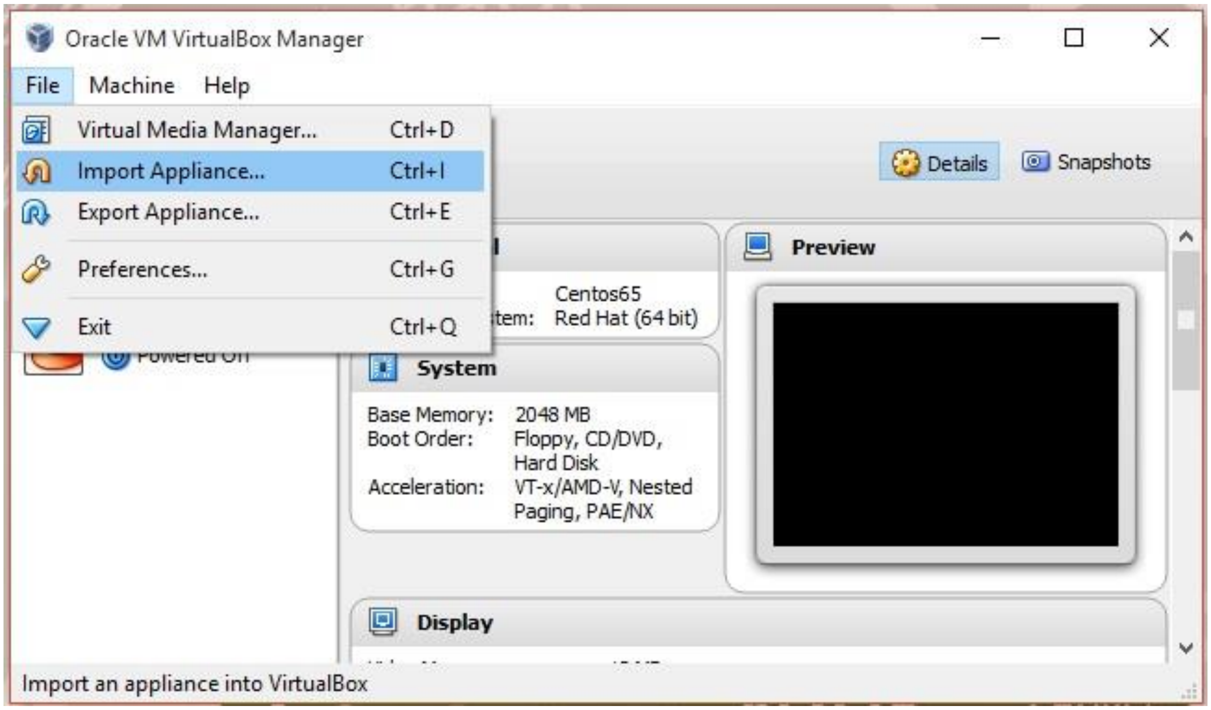
After downloading the **cloudera-quickstart-vm-5.5.0-0-virtualbox.ovf** file, we need to import it using virtual box. For that, first of all, you need to install virtual box in your system. Follow the steps given below to import the downloaded image file.

Step 1

Download virtual box from the following link and install it https://www.virtualbox.org/

Step 2

Open the virtual box software. Click **File** and choose **Import Appliance**, as shown below.


Oracle VM VirtualBox Manager
File
Machine
Help
Virtual Media Manager... Ctrl+D
Import Appliance... Ctrl+I
Export Appliance... Ctrl+E
Preferences... Ctrl+G
Exit Ctrl+Q
Details
Snapshots
Preview
System
Display
Powered Off

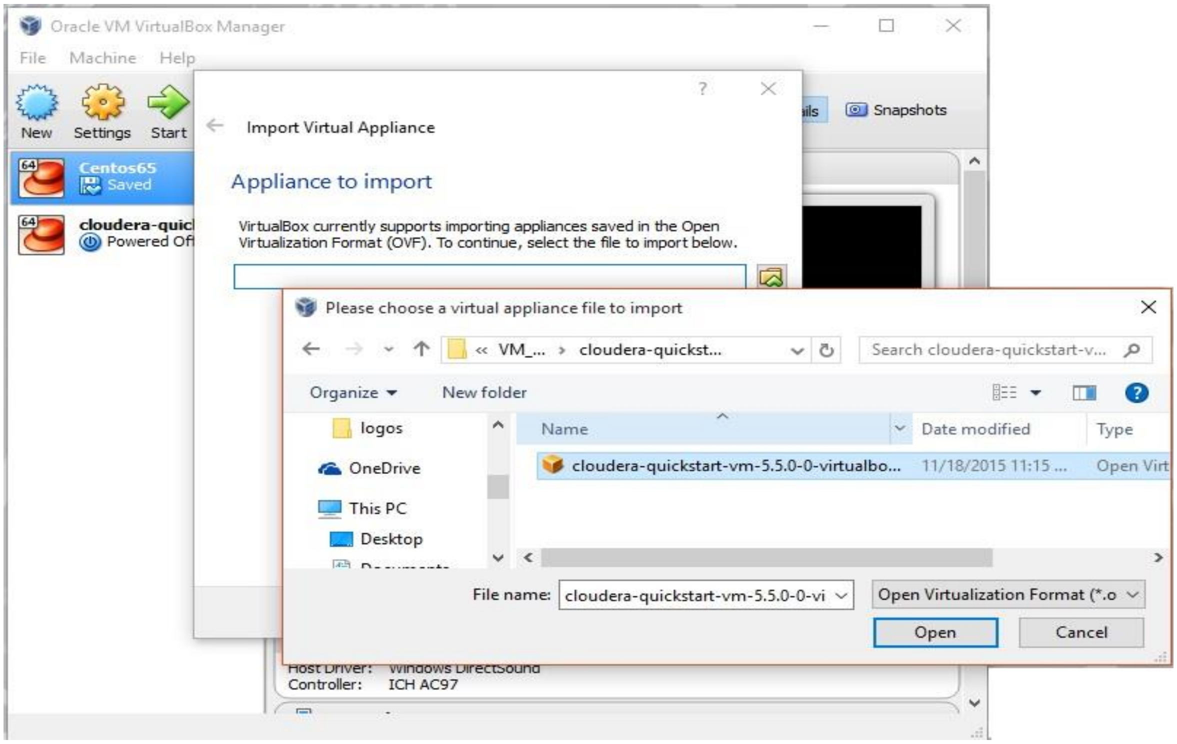
Centos65
Red Hat (64 bit)

Base Memory: 2048 MB
Boot Order: Floppy, CD/DVD, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, PAE/NX

Import an appliance into VirtualBox

Step 3

On clicking **Import Appliance**, you will get the Import Virtual Appliance window. Select the location of the downloaded image file as shown below.



Appliance to import

VirtualBox currently supports importing appliances saved in the Open Virtualization Format (OVF). To continue, select the file to import below.

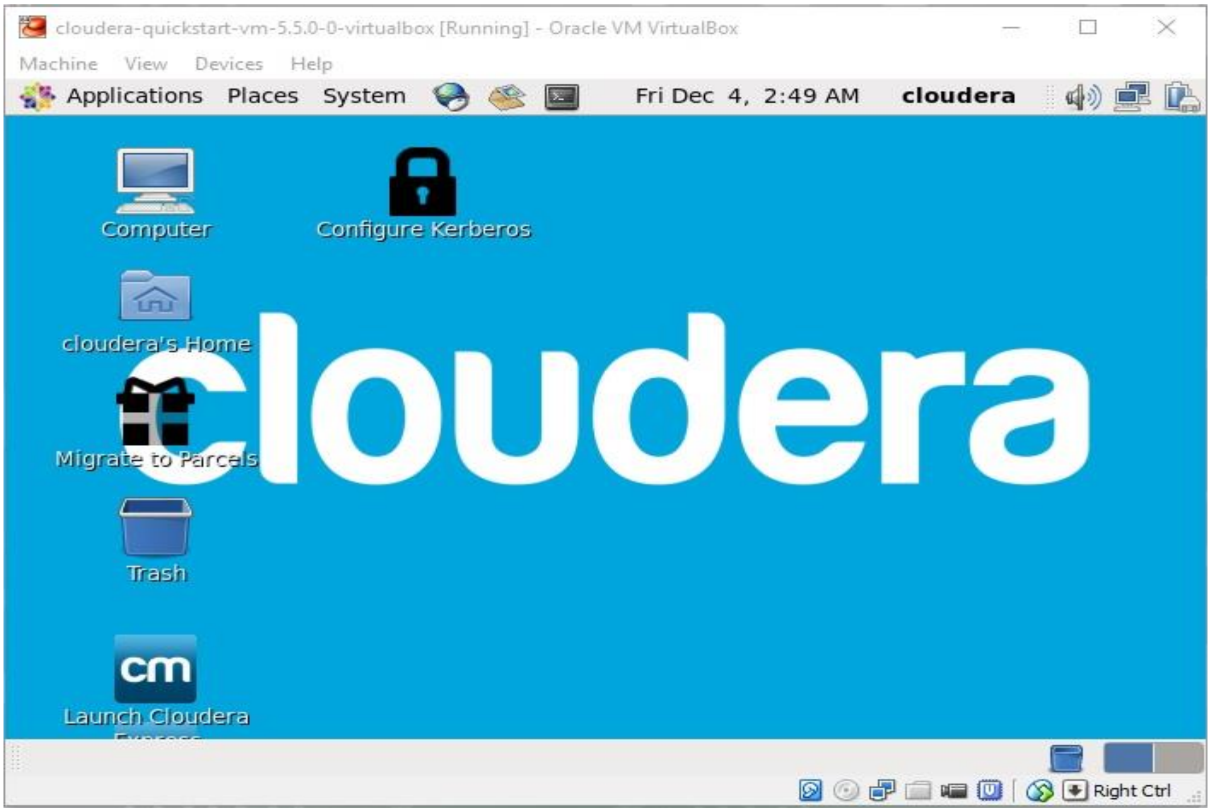
cloudera-quickstart-vm-5.5.0-0-virtualbo... 11/18/2015 11:15 ... Open Virt

File name: cloudera-quickstart-vm-5.5.0-0-vi

Open Virtualization Format (*.o

Host Driver: Windows DirectSound
Controller: ICH AC97

After importing **Cloudera QuickStartVM** image, start the virtual machine. This virtual machine has Hadoop, cloudera Impala, and all the required software installed. The snapshot of the VM is shown below.



Starting Impala Shell

To start Impala, open the terminal and execute the following command.

```
[cloudera@quickstart ~] $ impala-shell
```

This will start the Impala Shell, displaying the following message.

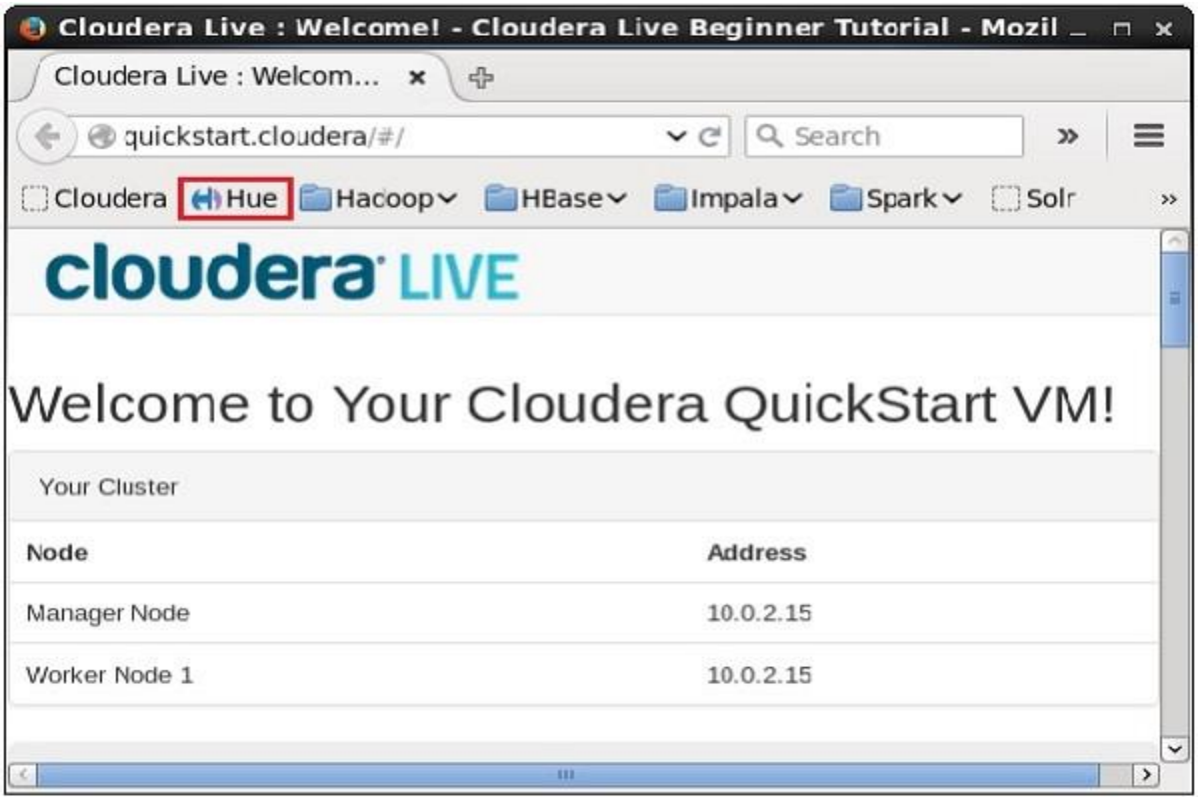
```
Starting Impala Shell without Kerberos authentication
Connected to quickstart.cloudera:21000
Server version: impalad version 2.3.0-cdh5.5.0 RELEASE (build
0c891d79aa38f297d244855a32f1e17280e2129b)
******************************************************************************
Welcome to the Impala shell. Copyright (c) 2015 Cloudera, Inc. All rights
reserved.
(Impala Shell v2.3.0-cdh5.5.0 (0c891d7) built on Mon Nov  9 12:18:12 PST 2015)

Press TAB twice to see a list of available commands.
******************************************************************************
[quickstart.cloudera:21000] >
```

Note: We will discuss all the impala-shell commands in later chapters.

Impala Query editor

In addition to **Impala shell**, you can communicate with Impala using the Hue browser. After installing CDH5 and starting Impala, if you open your browser, you will get the cloudera homepage as shown below.

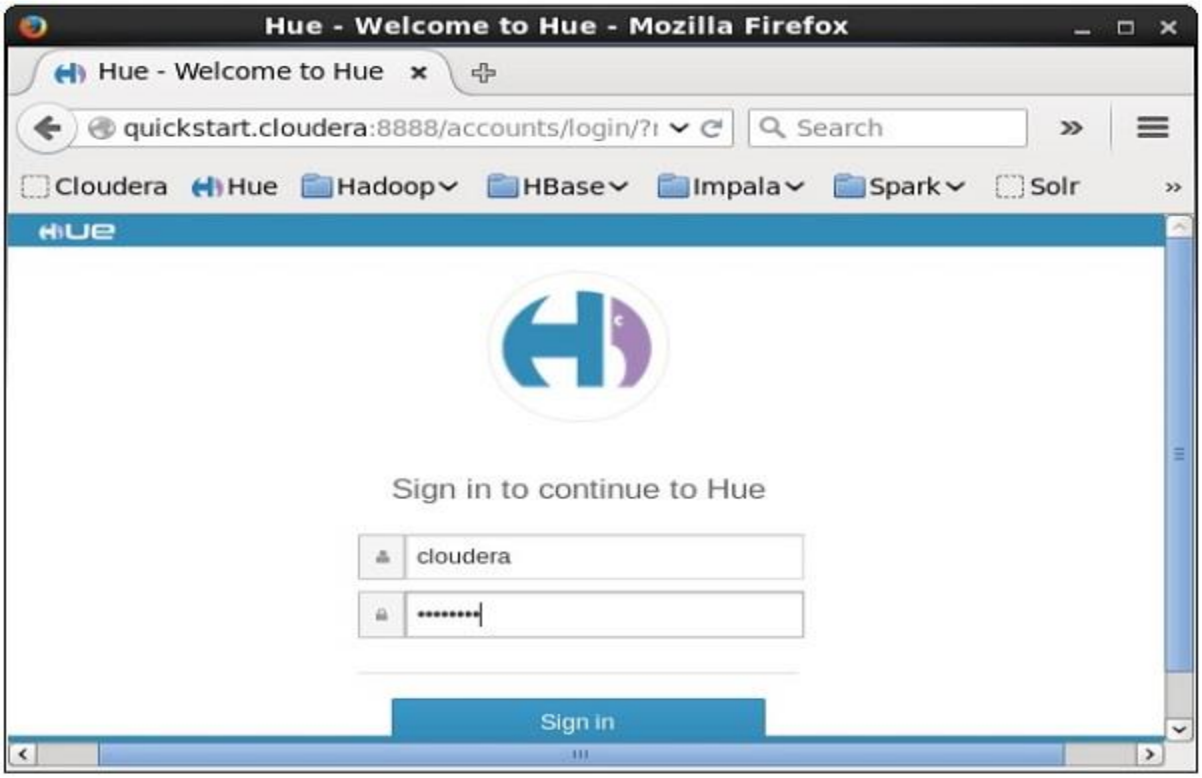

Cloudera Live : Welcome! - Cloudera Live Beginner Tutorial - Mozil
Cloudera Live : Welcom...
quickstart.cloudera/#/
Search
Cloudera
Hue
Hadoop
HBase
Impala
Spark
Solr
cloudera LIVE

Welcome to Your Cloudera QuickStart VM!

Your Cluster

Node	Address
Manager Node	10.0.2.15
Worker Node 1	10.0.2.15

Now, click the bookmark **Hue** to open the Hue browser. On clicking, you can see the login page of the Hue Browser, logging with the credentials cloudera and cloudera.



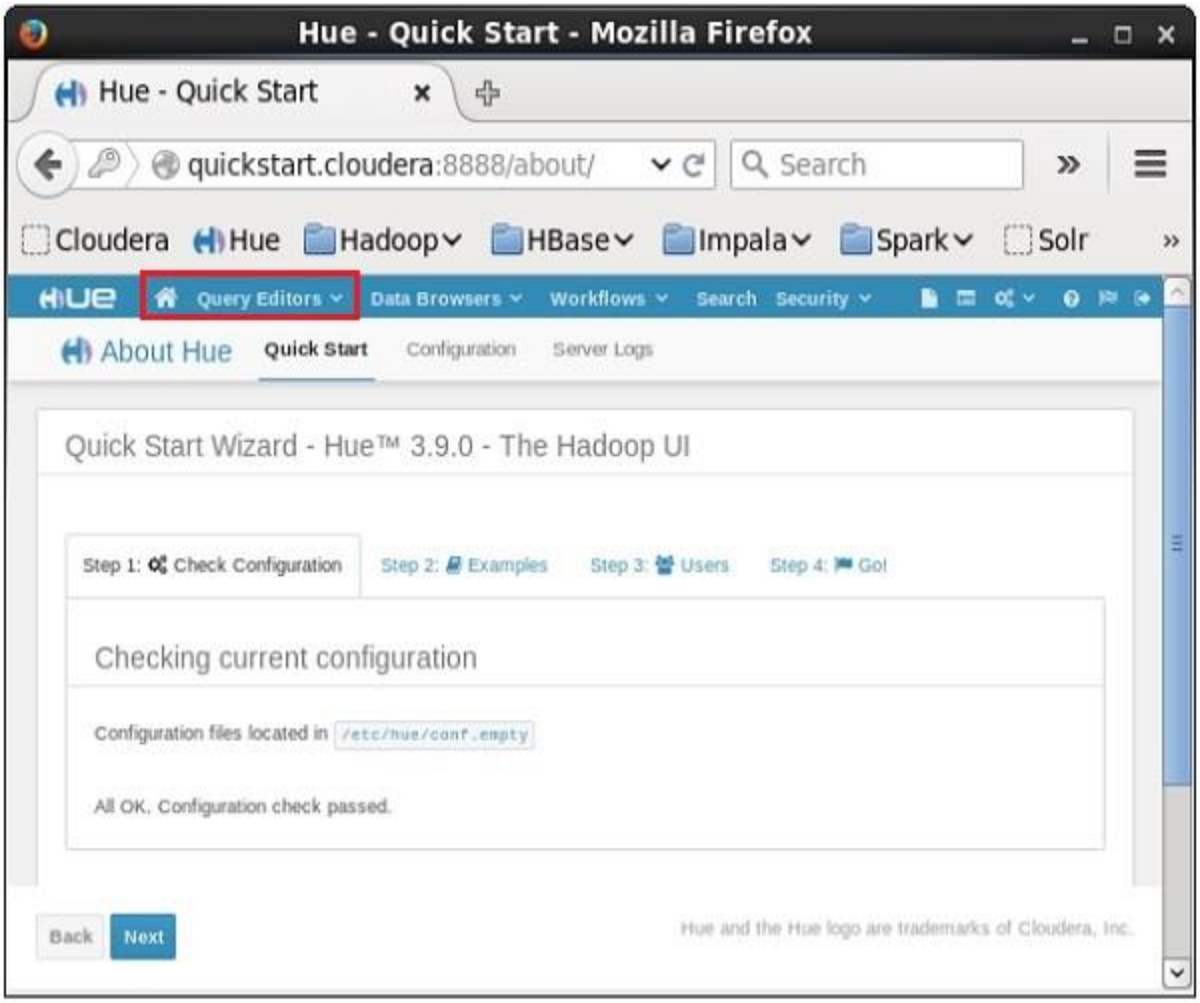
Hue - Welcome to Hue - Mozilla Firefox
Hue - Welcome to Hue
quickstart.cloudera:8888/accounts/login/?r
Search
Cloudera
Hue
Hadoop
HBase
Impala
Spark
Solr
Hue
Sign in

Sign in to continue to Hue

cloudera

•••••••••

As soon as you log on to the Hue browser, you can see the Quick Start Wizard of Hue browser as shown below.



Quick Start Wizard - Hue™ 3.9.0 - The Hadoop UI

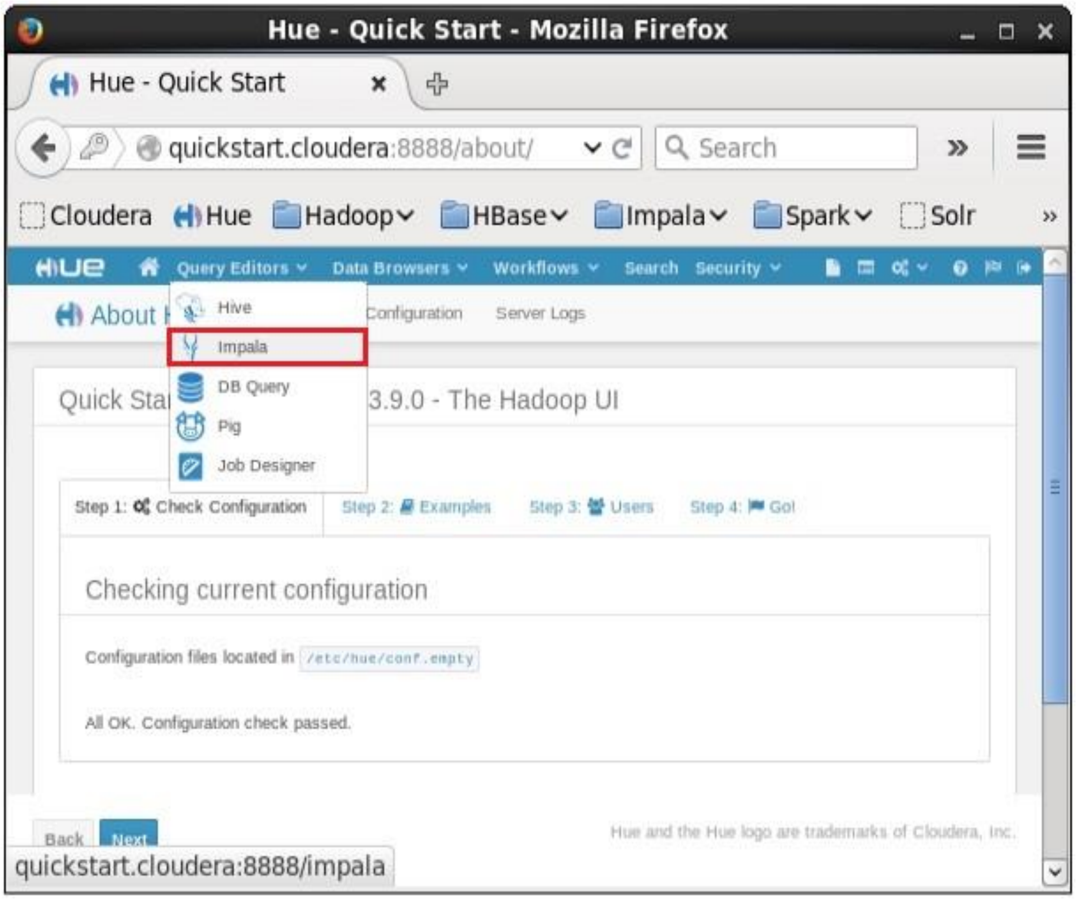
Checking current configuration

Configuration files located in `/etc/hue/conf.empty`

All OK. Configuration check passed.

Hue and the Hue logo are trademarks of Cloudera, Inc.

On clicking the **Query Editors** drop-down menu, you will get the list of editors Impala supports as shown in the following screenshot.

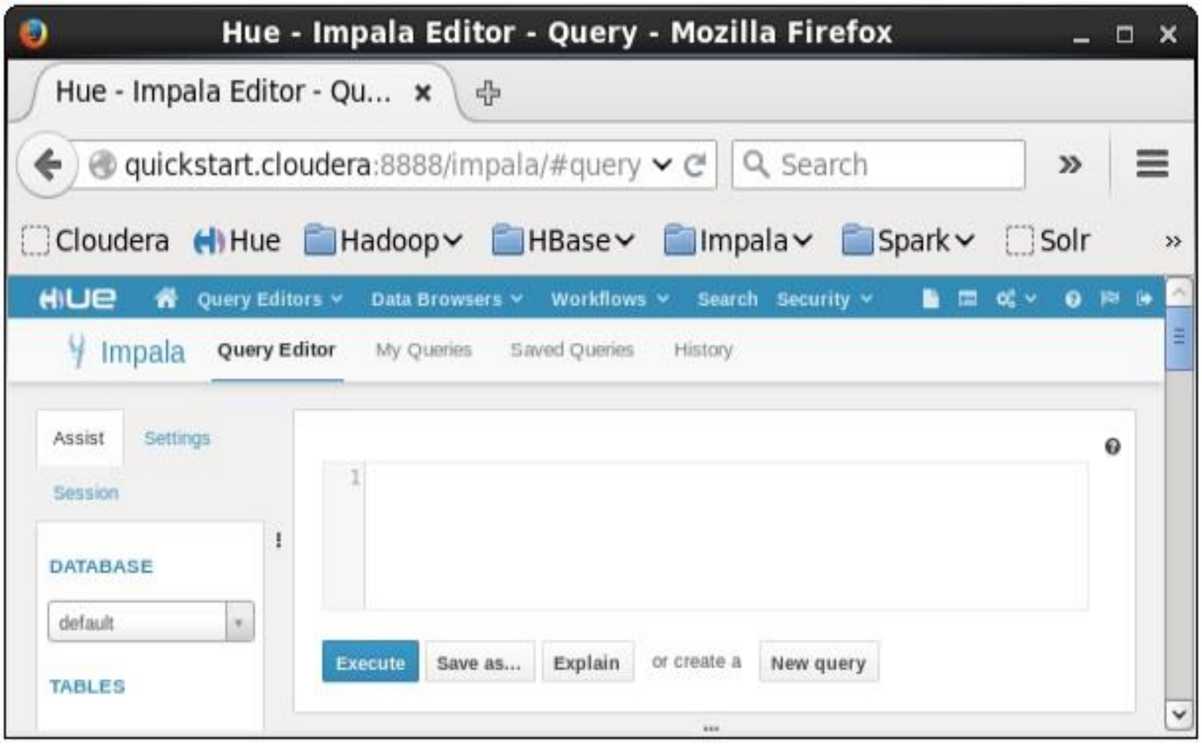


Checking current configuration

Configuration files located in `/etc/hue/conf.empty`

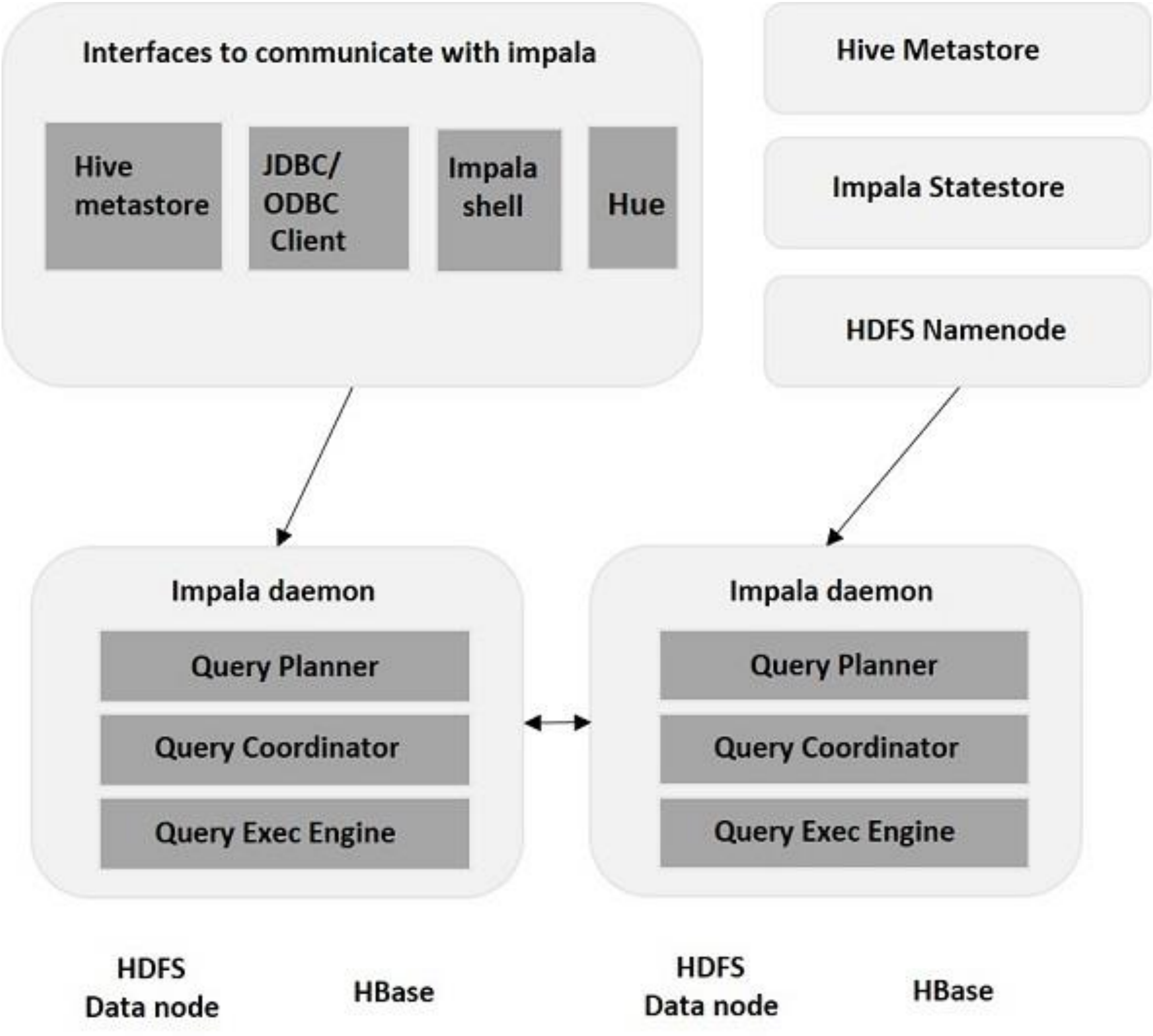
All OK. Configuration check passed.

On clicking **Impala** in the drop-down menu, you will get the Impala query editor as shown below.


Hue - Impala Editor - Query - Mozilla Firefox
Hue - Impala Editor - Qu...
quickstart.cloudera:8888/impala/#query
Search
Cloudera
Hue
Hadoop
HBase
Impala
Spark
Solr
HUE
Query Editors
Data Browsers
Workflows
Search
Security
Impala
Query Editor
My Queries
Saved Queries
History
Assist
Settings
Session
DATABASE
default
TABLES
1
Execute
Save as...
Explain
or create a
New query

3. IMPALA – ARCHITECTURE

Impala is an MPP (Massive Parallel Processing) query execution engine that runs on a number of systems in the Hadoop cluster. Unlike traditional storage systems, impala is decoupled from its storage engine. It has three main components namely, Impala daemon (*Impalad*), Impala Statestore, and Impala metadata or metastore.



Impala daemon (*Impalad*)

Impala daemon (also known as **impalad**) runs on each node where Impala is installed. It accepts the queries from various interfaces like impala shell, hue browser, etc.… and processes them.

Whenever a query is submitted to an impalad on a particular node, that node serves as a "**coordinator node**" for that query. Multiple queries are served by *Impalad* running on other nodes as well. After accepting the query, *Impalad* reads and writes to data files and parallelizes the queries by distributing the work to the other Impala nodes in the Impala.