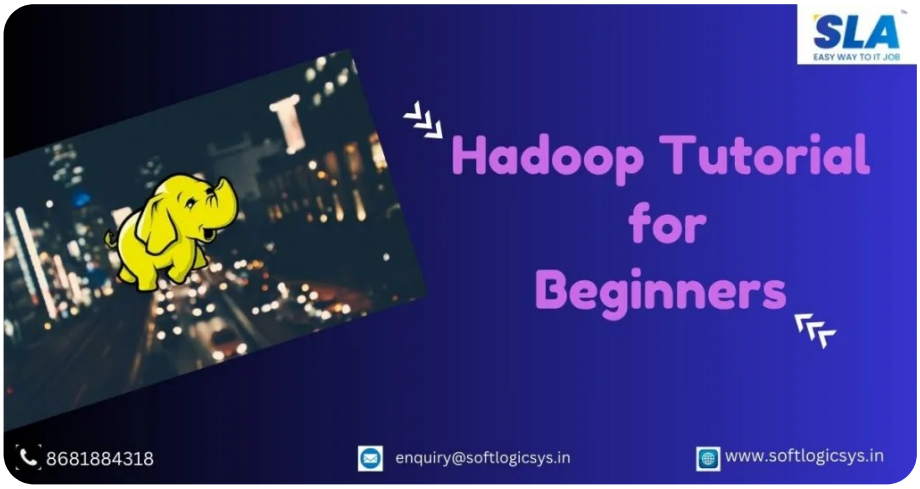

Share on your Social Media

Hadoop Tutorial for Beginners

Published On: September 16, 2024

Hadoop Tutorial for Beginners

The Hadoop framework provides customers with a processing solution for various database formats. Learn this Hadoop tutorial to understand the fundamental concepts to work efficiently with large data sets.

Download Hadoop Tutorial PDF

Introduction to Hadoop

Large-scale data processing and archiving for applications is handled by Hadoop, an open-source Java platform. Big data and analytics tasks are handled by Hadoop using distributed storage and parallel processing. They help in dividing workloads into smaller tasks that may be completed concurrently.

In this Hadoop tutorial, you will learn the following:

- Overview of Hadoop
- Hadoop Architecture
- Popular Hadoop Frameworks
- Map Reduce in Hadoop
- Hadoop Installation
- Advantages of Hadoop

Overview of Hadoop

In a distributed computing context, Hadoop is an open-source

software framework used for processing and storing massive volumes of data. It is built on the MapReduce programming approach, which enables the concurrent processing of huge datasets and is intended to manage big data.

Hadoop Interview Questions and Answers

Primary Modules of Hadoop

Here are the four main parts of Hadoop:

HDFS: Hadoop File System Distributed Access. HDFS was created based on Google's GFS article, which was released. It specifies that the files will be divided into blocks and kept on different distributed architecture nodes.

Yarn: The cluster is managed and jobs are scheduled by "Yet Another Resource Negotiator."

Map Reduce: Java programs can use key-value pairs to compute data in parallel thanks to this architecture.

- The Map task converts the input data into a set of data from which key-value pairs can be calculated.
- The reduce job uses the output from the map task, and the reducer's output produces the expected result.

Hadoop Common: Hadoop is started with these **Java libraries**, which are also utilized by other Hadoop modules.

Hadoop Architecture

- The MapReduce engine, the Hadoop Distributed File System (HDFS), and the file system comprise the Hadoop architecture.
- Either YARN/MR2 or MapReduce/MR1 can be the MapReduce engine.
- A Hadoop cluster is made up of several slave nodes and one master node.
    - **Master Node:** Job Tracker, Task Tracker, NameNode, and DataNode.
    - **Slave Node:** TaskTracker and DataNode.

Features of Hadoop

- **Distributed Storage:** Hadoop distributes the storage of massive data sets among numerous computers, enabling the processing and storing of extraordinarily huge volumes of data.
- **Scalability:** Hadoop is scalable, meaning it is simple to add more capacity as needed, ranging from a single server to thousands of machines.
- **Fault-Tolerance:** Hadoop is built with a high degree of fault tolerance, which enables it to function even when hardware fails.
- **Data Locality:** One of Hadoop's features, data locality, allows

Hadoop Project Ideas
Published On: October 12, 2024

Introduction A Hadoop professional focuses on utilizing the Hadoop framework for big data tasks. Their...

VMware Tutorial for Cloud Computing Aspirants
Published On: October 12, 2024

VMware Tutorial for Cloud Computing Aspirants VMware software allows you to run a virtual machine...

VBA Macros Tutorial for Beginners
Published On: October 10, 2024

VBA Macros Tutorial for Beginners VBA macros are programs that automate repetitive operations in Microsoft...

data to be stored on the same node as its processing, reducing network traffic and enhancing performance.
- **High Availability:** Hadoop has a feature called High Availability that helps to ensure that data is never lost and is always available.
- **Flexible Data Processing:** A wide range of data processing jobs may be easily implemented thanks to Hadoop's MapReduce programming architecture, which enables distributed data processing.
- **Data Integrity:** A built-in checksum mechanism in Hadoop helps to guarantee that the data is accurate and consistent when saved.
- **Data Replication:** Hadoop has a feature called data replication that aids in fault tolerance by replicating data throughout the cluster.
- **Data Compression:** One of Hadoop's built-in features, data compression helps to save storage space and boost efficiency.
- **YARN:** A platform for resource management that enables the execution and processing of several data processing engines, including interactive SQL, batch processing, and real-time streaming, on HDFS data.

For all your **data science training** needs, get in touch with us.

Popular Hadoop frameworks

Here are some popular frameworks used in Hadoop:

- **Hive:** It writes complex MapReduce programs on HDFS and structures data using HiveQl.
- **Drill:** This data exploration tool is made up of user-defined functions.
- **Storm:** It enables data streaming and processing in real-time.
- **Spark** is a popular data processing tool that includes a Machine Learning Library (MLlib) for improved machine learning. It is also compatible with Scala, **Python**, and Java.
- **Pig** has a **SQL**-like language called Pig Latin that can transform unstructured data.
- **Tez:** It makes Hive and Pig less complex and speeds up the execution of their programs.

Map Reduce in Hadoop

The programming model used in the Hadoop framework is called MapReduce. The map and reduce technique is used in MapReduce to analyze data.

- The massive volumes of data stored in the Hadoop Distributed File System (HDFS) can be accessed using the MapReduce parallel, distributed programming style of the Hadoop framework.
- Hadoop can run the MapReduce application built in many languages, such as Python, Ruby, and Java.

One of the benefits of MapReduce algorithms is that they are naturally parallel, which facilitates large-scale data analysis.

When the MapReduce programs run in parallel, they perform faster. The steps to execute MapReduce scripts are as follows:

- **Dividing the input into fixed-size segments:** First, it splits the job into segments of the same size. Dividing the task into equal-sized portions isn't the best course of action when the file size changes because some processes will finish considerably sooner than others, and some may take a very long time to finish.

Therefore, dividing the input into fixed-size chunks and assigning each chunk to a process is one of the better options; however, it does involve more work.

- **Combining the results:** In MapReduce programming, combining the results from several processes is an important operation since it frequently requires additional processing, such as aggregating and completing the results.

Major Components of Map Reduce

The MapReduce consists of two main parts. The map phase and the reduction phase are the two main stages of the MapReduce process. Each phase includes a map function and a reducer function in addition to the key-value pairs that serve as input and output.

- **Mapper:** The MapReduce process begins with the Mapper phase. Each input record is processed by the Mapper, while the RecordReader and InputSplit create the key-value pairs. where the input pair and these key-value pairs may differ entirely. All of these key-value pairs are collected in the MapReduce output.
- **Reducer:** The second stage of the MapReduce process is the reducer phase. It is in charge of handling the mapper's output. The reducer now creates a new set of output that can be saved in HDFS as the final output data after it has finished processing the mapper's output.

How Does Map Reduce Work?

MapReduce uses the many components of the data to process it in stages.

- **Input Files:** The input files include the data needed for MapReduce jobs. HDFS is where these input files are kept.
- **InputFormat:** The files or items utilized as input are chosen by it. The input split is created using the "*InputFormat.*" Until the file reading process is finished, the Record Reader and the Input Split exchange messages.
- **Record Reader:** In the Hadoop MapReduce, the RecordReader

can interact with the Input Split. For the mapper to read the data, it can also turn it into key-value pairs.

- **Mapper:** The input records are sent to the mapper via the RecordReader. The RecordReader's input records must be processed by the Mapper for it to produce the new key-value pair.
- **Combiner:** Another name for the Combiner in MapReduce is Mini-reducer. To limit the amount of data transferred between the mapper and reducer, the Hadoop MapReduce combiner locally aggregates the mapper's output. The output of the combiner is sent to the partitioner for additional processing when it has finished its job.
- **Partitioner:** When using several reducers in Hadoop MapReduce, the partitioner is utilized.
    - After removing the output from the combiner, the partitioner divides the data.
    - After the output is divided according to the key, it is sorted.
    - The partition is derived from the key using a hash function.

Every combiner output is divided, a record with the same key value moves into the same partition, and each partition is then delivered to the reducer since MapReduce execution relies on key-value mapping. The even distribution of the map output over the reducer is made possible by partitioning the combiner's output.

- **Sorting and shuffling:** Before the mapper's output is forwarded to the reduction phase, it undergoes a sorting step. This intermediate output is mixed and sorted after every mapper has finished their work and their output has been said to be shuffled on the reducer nodes. The reduction phase receives this sorted output as input.
- **Reducer:** It uses the mapper's set of intermediate key-value pairs as the input and applies the reducer function to each pair of keys to produce the desired result. The final output of the reduction phase is this one, which is kept in the HDFS.
- **Record Writer:** The ability to write these output key-value pairs from the reduction phase to the output files is possessed by the Record Writer.
- **Output format:** This format controls how the record reader writes these output values in the output files. Writing files to the local disk or HDFS, is the usual usage for the Hadoop output format instances.

Hadoop Installation

Although Hadoop is designed for usage in a UNIX production environment, Cygwin enables Windows users to access it. Map Reduce programs must be run on Java 1.6 or later. To install Hadoop from a tarball in a **Unix environment**, you must

- Java Installation
- SSH installation

- Hadoop Installation and File Configuration

Java Installation

Step 1: To check if Java is installed or not, type “java -version” into the prompt. If not, go here to download Java.

http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html

Step 2: Use the command below to extract the file.

#tar zxf jdk-7u71-linux-x64.tar.gz

Step 3: Move the file to /usr/local and set the path to enable Java for all UNIX users. To relocate the JDK to /usr/lib, switch to the root user at the prompt and enter the following command.

 # mv jdk1.7.0_71 /usr/lib/

To configure the path, add the following instructions to the ~/.bashrc file.

 # export JAVA_HOME=/usr/lib/jdk1.7.0_71

 # export PATH=PATH:$JAVA_HOME/bin

Now that you have typed “java -version” into the prompt, you may verify the installation.

SSH Installation

Passwords are not requested while interacting with the master and slave computers over SSH. Make a Hadoop user on the master and slave systems first.

 # useradd hadoop

 # passwd Hadoop

To map the nodes, open the host file located in each machine’s /etc/ folder and provide the hostname and IP address.

 # vi /etc/hosts

Enter the lines below:

190.12.1.114 hadoop-master

190.12.1.121 hadoop-salve-one

190.12.1.143 hadoop-slave-two

Configure each node with an SSH key so that it may communicate with the others without a password. The commands are:

 # su hadoop

$ ssh-keygen -t rsa

$ ssh-copy-id -i ~/.ssh/id_rsa.pub tutorialspoint@hadoop-master

$ ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop_tp1@hadoop-slave-1

$ ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop_tp2@hadoop-slave-2

$ chmod 0600 ~/.ssh/authorized_keys

$ exit

Hadoop Developer Salary

Hadoop Installation

After installing Hadoop, extract the Hadoop now and move it to a different location.

$ mkdir /usr/hadoop

$ sudo tar vxzf hadoop-2.2.0.tar.gz ?c /usr/hadoop

Modify who owns the Hadoop folder.

$sudo chown -R hadoop usr/hadoop

Modify the configuration files for Hadoop:

There are all the files in */usr/local/Hadoop/etc/hadoop*

Step 1: In the file hadoop-env.sh, add

export JAVA_HOME=/usr/lib/jvm/jdk/jdk1.7.0_71

Step 2: Add the following between the configuration tabs in core-site.xml

<configuration>

<property>

<name>fs.default.name</name>

<value>hdfs://hadoop-master:9000</value>

</property>

<property>

<name>dfs.permissions</name>

<value>false</value>

</property>

</configuration>

Step 3: After switching between the configuration tabs in hdfs-site.xmladd,

<configuration>

<property>

<name>dfs.data.dir</name>

<value>usr/hadoop/dfs/name/data</value>

<final>true</final>

</property>

<property>

<name>dfs.name.dir</name>

<value>usr/hadoop/dfs/name</value>

<final>true</final>

</property>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

</configuration>

Step 4: Make the necessary changes to Mapred-site.xml as indicated below.

<configuration>

<property>

<name>mapred.job.tracker</name>

<value>hadoop-master:9001</value>

</property>

</configuration>

Step 5: Now, make updates to $HOME/.bahsrc.

cd $HOME

vi .bashrc

Add the final few lines, save, then close the document.

#Hadoop variables

export JAVA_HOME=/usr/lib/jvm/jdk/jdk1.7.0_71

export HADOOP_INSTALL=/usr/hadoop

export PATH=$PATH:$HADOOP_INSTALL/bin

export PATH=$PATH:$HADOOP_INSTALL/sbin

export HADOOP_MAPRED_HOME=$HADOOP_INSTALL

export HADOOP_COMMON_HOME=$HADOOP_INSTALL

export HADOOP_HDFS_HOME=$HADOOP_INSTALL

export YARN_HOME=$HADOOP_INSTALL

Use the following command to install Hadoop on the slave system.

 # su hadoop

$ cd /opt/hadoop

$ scp -r hadoop hadoop-slave-one:/usr/hadoop

$ scp -r hadoop hadoop-slave-two:/usr/Hadoop

Set up the slave and master nodes.

$ vi etc/hadoop/masters

hadoop-master

$ vi etc/hadoop/slaves

hadoop-slave-one

hadoop-slave-two

Following this pattern, launch every daemon and name the node.

 # su hadoop

$ cd /usr/hadoop

$ bin/hadoop namenode -format

$ cd $HADOOP_HOME/sbin

$ start-all.sh

Advantages of Hadoop

- **Quick:** The data in HDFS is mapped and dispersed throughout the cluster, facilitating quicker retrieval. Processing times are shortened because even the data processing tools are frequently located on the same servers.
- **Scalable:** Adding nodes to an existing Hadoop cluster allows it to grow.
- **Cost-effective:** Hadoop is significantly less expensive than a standard relational database management system because it is open source and stores data on commodity hardware.
- **Resilient to failure:** Hadoop can use the other copy of the data in the event of a network failure or node outage since HDFS possesses the ability to duplicate data across networks. Replication factors are customizable, although data are typically copied three times.

Get placed in your dream job by enrolling in our IT training and **placement training institute**.

Conclusion

In this Hadoop tutorial, we have explained Hadoop fundamentals that cover architectures, frameworks, MapReduce, and installation. Gain expertise with our **Hadoop training in Chennai**.

Softlogic Academy

Softlogic Systems

KK Nagar [Corporate Office]

No.10, PT Rajan Salai, K.K. Nagar, Chennai – 600 078.
Landmark: Karnataka Bank Building
Phone: +91 86818 84318
Email: enquiry@softlogicsys.in
Map: Google Maps Link

OMR

No. E1-A10, RTS Food Street
92, Rajiv Gandhi Salai (OMR),
Navalur, Chennai - 600 130.
Landmark: Adj. to AGS Cinemas
Phone: +91 89256 88858
Email: info@softlogicsys.in
Map: Google Maps Link

Terms and Conditions

Courses

Python
Software Testing
Full Stack Developer
Java
Power BI
Clinical SAS
Data Science
Embedded
Cloud Computing
Hardware and Networking
VBA Macros
Mobile App Development
DevOps

Social Media Links

Review Sources

Google
Trustpilot
Glassdoor
Mouthshut
Sulekha
Justdial
Ambitionbox
Indeed
Software Suggest
Sitejabber