

Share on your Social Media      

# Big Data Hadoop Tutorial

Published On: July 4, 2024

## Big Data Hadoop Tutorial

A software system called Hadoop is made to handle big data. This Hadoop tutorial covers both fundamental and advanced concepts for both novices and experts.

[Download Big Data Hadoop Tutorial PDF](#)

## Introduction to Big Data Hadoop

**Big Data:** Big Data are extremely large-scale data sets. Typically, we work with data that is MB (Word Doc, Excel) or up to GB (movies, codes); big data is defined as data that is in petabytes.

**Big Data Hadoop:** Hadoop is an Apache open-source platform used for processing and analyzing massive volumes of data. Facebook, Yahoo, Google, Twitter, LinkedIn, and numerous other companies use it.

## Modules of Hadoop

**HDFS:** Hadoop Distributed File System. It says that blocks of files will be divided up and kept in nodes throughout the distributed architecture.

**Yarn:** Yet Another Resource Negotiator. It is used for managing the cluster and scheduling jobs.


**Map Reduce:** It is a framework that aids Java programs in leveraging key-value pairs to compute data in parallel. The 'Map' process converts data input into a collection of data that can be computed in key-value pairs.

**Hadoop Common:** Other Hadoop modules use these Java libraries, which are also used to launch Hadoop.

[Big Data Hadoop Syllabus PDF](#)






## Featured Articles

 **Want to know more about becoming an expert in IT?**

[Click Here to Get Started](#) >>

100% Placement Assurance

AUTHOR CERTIFIED PARTNER





Quick Enquiry

## Related Courses at SLA

- ➔ [Big Data Hadoop Training in OMR](#)
- ➔ [Big Data Hadoop Training in Chennai](#)

## Related Posts



### C and C++ Tutorial

Published On: August 1, 2024

C and C++ Tutorial C is a high-level, procedural, general-purpose programming language. Whereas C++, a...

## Hadoop Architecture

The MapReduce engine, the Hadoop Distributed File System (HDFS), and the file system comprise the Hadoop architecture.

A Hadoop cluster is made up of several slave nodes and one master node.

Master Nodes: Job Tracker, Task Tracker, NameNode, and DataNode.

Slave Nodes: TaskTracker and DataNode.

## Hadoop Distributed File System

HDFS has a master/slave architecture. This design consists of several DataNodes acting as slaves and a single NameNode acting as the master.

### NameNode

The HDFS cluster consists of a single master server. Being a single node, it could lead to a single point of failure.

It streamlines the system's architecture. It does this by opening, renaming, and shutting files to manage the file system namespace.

### DataNode

There are several DataNodes in the HDFS cluster. Multiple data blocks are present in every DataNode. The purpose of these data blocks is data storage.

DataNode is in charge of reading and writing requests from clients of the file system. On the NameNode's instruction, it creates, deletes, and replicates blocks.

### Job Tracker

Accepting MapReduce jobs from clients and using NameNode to process the data is Job Tracker's responsibility. As a result, NameNode gives Job Tracker metadata.

### Task Tracker

It functions as a Job Tracker slave node. It applies the code to the file after receiving the task and code from Job Tracker. Another name for this procedure is a mapper.

### Map Reduce Layer

The MapReduce is generated when the client application sends the MapReduce job to Job Tracker. In response, the job tracker forwards the request to the appropriate task trackers.



### ASP DOTNET Tutorial

Published On: July 31, 2024

ASP DOTNET Tutorial Microsoft created the web framework known as ASP.NET. It is employed in...



### Artificial Intelligence Tutorial

Published On: July 30, 2024

Artificial Intelligence Tutorial Artificial intelligence (AI) is significant since it enhances many facets of society...



### Appium Testing Tutorial

Published On: July 30, 2024

Appium Testing Tutorial Designed to make the UI automation of many app platforms easier, Appium...

The TaskTracker occasionally times out or fails. That portion of the work is rescheduled in such a scenario.

## Hadoop Installation

To install Hadoop from a 'tar ball' in a UNIX environment, you require the following:

- Java Installation
- SSH installation
- Hadoop Installation and File Configuration

## Java Installation

**Step 1:** Get Java at

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

if it's not already installed. On your computer, the tar file *jdk-7u71-linux-x64.tar.gz* will be downloaded.

**Step 2:** Use the command below to extract the file. *#tar xzf jdk-7u71-linux-x64.tar.gz*

**Step 3:** Move the file to */usr/local* and configure 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
```

Fill in 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. Instructions for the same 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 Installation**

Download links for Hadoop are available at

<http://developer.yahoo.com/hadoop/tutorial/module3.html>

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 *hadoop-env.sh* file add

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

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

```
<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 on *hdfs-site.xml* add,

```
<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:** Lastly, make updates to \$HOME/.bashrc.

```
cd $HOME
```

```
vi .bashrc
```

Append following lines in the end and save and exit

```
#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
```

## Big Data Hadoop Interview Questions

### HDFS Basic File Operations

#### Step 1: Transferring data from the local file system to HDFS

Create an HDFS folder first so that data from the local file system can be stored there.

```
$ hadoop fs -mkdir /user/test
```

Copy the file "data.txt" from a file stored in the local folder /usr/home/Desktop to the HDFS folder /user/test

```
$ hadoop fs -copyFromLocal /usr/home/Desktop/data.txt /user/test
```

Show the contents of the HDFS folder with the command

```
$ Hadoop fs -ls /user/test
```

#### Step 2: Transfer data from HDFS to the local file system with the command

```
$ hadoop fs -copyToLocal /user/test/data.txt /usr/bin/data_copy.txt
```

#### Step 3: Verify if the files are identical by comparing them.

```
$ md5 /usr/bin/data_copy.txt /usr/home/Desktop/data.txt
```

## Recursive Deleting

`hadoop fs -rmr <arg>`

**Example:** `hadoop fs -rmr /user/sonoo/`

## HDFC Other Commands

The commands make use of the following.

- “<path>” denotes the name of any file or directory.
- “<path>...” denotes a file or directory name or names.
- “<file>” can refer to any filename.
- In a directed operation, the path designations are “<dest>” and “<src>”.
- “<localSrc>” and “<localDest>” are paths on the local file system, similar to those above.

**put <localSrc><dest>**: It copies the file or directory from the local file system, denoted with localSrc, to dest in the DFS.

**copyFromLocal <localSrc><dest>**: Similar to -put

**copyFromLocal <localSrc><dest>**: Similar to -put

**moveFromLocal <localSrc><dest>**: It copies the file or directory to dest in HDFS from the local file system that localSrc has identified, then, upon success, deletes the local copy.

**get [-crc] <src><localDest>**: The file or directory is moved locally from HDFS, denoted with src, to the local file system path, denoted as localDest.

**cat <file-name>**: It shows the contents of the filename on the standard output.

**moveToLocal <src><localDest>**: Similar to -get, except it removes the HDFS copy upon success.

**setrep [-R] [-w] rep <path>**: It sets the file names indicated by the path to the rep’s target replication factor. (Over time, the replication factor itself will approach the target.)

**touchz <path>**: It creates a file at the path with a timestamp of the present moment. fails if there is already a file in the path unless the file has a zero size.

**test -[ezd] <path>**: Returns 0 otherwise, 1 if the path is a directory, has zero length, or both.

**stat [format] <path>**: It prints the path information. File size in blocks (%b), filename (%n), block size (%o), replication (%r), and modification date (%y, %Y) are all accepted in the format, which is



a string.

## Big Data Hadoop Training

### Conclusion

We cover the fundamentals of Hadoop technology in this Big Data Hadoop Tutorial. We hope this is useful for you to get started with big data analytics. Learn comprehensively with our [Big Data Hadoop training in Chennai](#).

Share on your Social Media



### 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](tel:+918681884318)

**Email:** [enquiry@softlogicsys.in](mailto:enquiry@softlogicsys.in)

**Map:** [Google Maps Link](#)

### OMR

No. EI-A10, RTS Food Street  
92, Rajiv Gandhi Salai (OMR),  
Navalur, Chennai – 600 130.

**Landmark:** Adj. to AGS Cinemas

**Phone:** [+91 89256 88858](tel:+918925688858)

**Email:** [info@softlogicsys.in](mailto:info@softlogicsys.in)

**Map:** [Google Maps Link](#)

### Courses

Python

Software Testing

### Navigation

[About Us](#)

[Blog Posts](#)

[Careers](#)

[Contact](#)

[Placement Training](#)

[Corporate Training](#)

[Hire With Us](#)

[Job Seekers](#)

[SLA's Recently Placed Students](#)

[Reviews](#)

[Sitemap](#)

### Important Links

[Disclaimer](#)

[Privacy Policy](#)

[Terms and Conditions](#)

### Social Media Links



Full Stack Developer

Java

Power BI

Clinical SAS

Data Science

Embedded

Cloud Computing

Hardware and Networking

VBA Macros

Mobile App Development

DevOps

## Review Sources

---

Google

Trustpilot

Glassdoor

Mouthshut

Sulekha

Justdial

Ambitionbox

Indeed

Software Suggest

Sitejabber