

Hadoop

Installation:

1. Download and install JDK

<https://www.oracle.com/in/java/technologies/downloads/#jdk19-windows>

2. Extracting the hadoop 3.3.0 tar file-

<https://archive.apache.org/dist/hadoop/common/hadoop-3.2.1/hadoop-3.2.1.tar.gz>

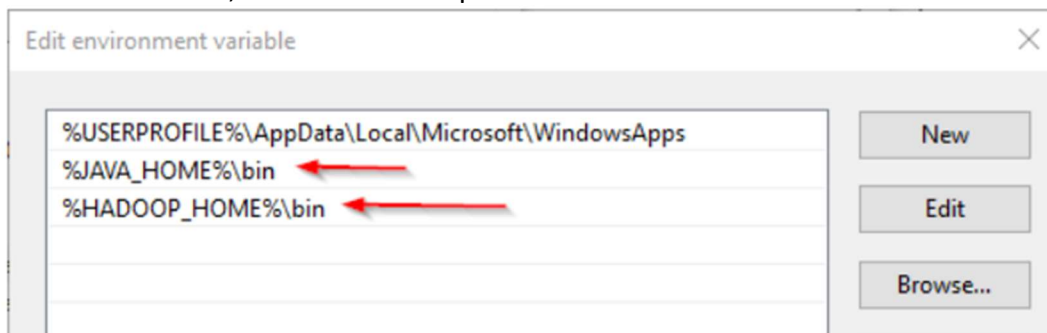
3. Adding hadoop winutils files to bin. (check same version)

<https://github.com/kontext-tech/winutils/tree/master/hadoop-3.2.1/bin>

4. Adding the hadoop files and jdk files to a right directory.(copy paste jdk files from Program files)

5. Add **Hadoop and jdk path** to Environment VARIABLES.

6. Also, in Path- add the paths as below:



7. Open CMD and check Hadoop version.

```
Microsoft Windows [Version 10.0.22621.819]
(c) Microsoft Corporation. All rights reserved.

C:\Users\gayat>hadoop --version
java 19.0.1 2022-10-18
Java(TM) SE Runtime Environment (build 19.0.1+10-21)
Java HotSpot(TM) 64-Bit Server VM (build 19.0.1+10-21, mixed mode, sharing)

C:\Users\gayat>
```

8. Startup commands for running the nodes from Hadoop-3.2.1/sbin> are:

.\start-dfs.cmd

.\start-yarn.cmd

If you face any error, follow the below steps further.

9. Configuring Hadoop cluster

There are four files we should alter to configure Hadoop cluster:

1. %HADOOP_HOME%\etc\hadoop\hdfs-site.xml
2. %HADOOP_HOME%\etc\hadoop\core-site.xml
3. %HADOOP_HOME%\etc\hadoop\mapred-site.xml
4. %HADOOP_HOME%\etc\hadoop\yarn-site.xml

9.1. HDFS site configuration

Hadoop is built using a master-slave paradigm. Before altering the HDFS configuration file, we should create a directory to store all master node (name node) data and another one to store data (data node). In this example, we created the following directories:

- E:\hadoop-env\hadoop-3.2.1\data\dfs\namenode
- E:\hadoop-env\hadoop-3.2.1\data\dfs\datanode

Now, open “hdfs-site.xml” file located in “%HADOOP_HOME%\etc\hadoop” directory, and add the following properties within the <configuration></configuration> element:

```
<property><name>dfs.replication</name><value>1</value></property><property><name>dfs.namenode.name.dir</name><value>file:///E:/hadoop-env/hadoop-3.2.1/data/dfs/namenode</value></property><property><name>dfs.datanode.data.dir</name><value>file:///E:/hadoop-env/hadoop-3.2.1/data/dfs/datanode</value></property>
```

9.2. Core site configuration

Now, configure the name node URL by adding the following XML code into the

<configuration></configuration> element within “core-site.xml”:

```
<property><name>fs.default.name</name><value>hdfs://localhost:9820</value></property>
```

9.3. Map Reduce site configuration

Now, add the following XML code into the <configuration></configuration> element within

“mapred-site.xml”:

```
<property><name>mapreduce.framework.name</name><value>yarn</value><description>MapReduce framework name</description></property>
```

9.4. Yarn site configuration

Now, add the following XML code into the <configuration></configuration> element within

“yarn-site.xml”:

```
<property><name>yarn.nodemanager.aux-services</name><value>mapreduce_shuffle</value><description>Yarn Node Manager Aux Service</description></property>
```

10. Formatting Name node

After the configuration, we'll format the name node using the following command:

```
hdfs namenode -format
```

If you face any error, this issue will be solved within the next release. For now, you can fix it temporarily using the following steps:

1. Download hadoop-hdfs-3.2.1.jar file from the [following link](#).

2. Rename the file name `hadoop-hdfs-3.2.1.jar` to `hadoop-hdfs-3.2.1.bak` in folder
`%HADOOP_HOME%\share\hadoop\hdfs`
 3. Copy the downloaded `hadoop-hdfs-3.2.1.jar` to folder
`%HADOOP_HOME%\share\hadoop\hdfs`
11. Start the Hadoop services now from Hadoop for running the nodes from -
`Hadoop-3.2.1/sbin>`

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

Running a prebuilt example of a Map Reduce program

1. Navigate to mapreduce directory on hadoop.
`\hadoop-3.3.0\share\hadoop\mapreduce> hadoop jar hadoop-mapreduce-examples-3.3.0.jar`

```
E:\Learning\HadoopHive\hadoop-env\hadoop-3.3.0\share\hadoop\mapreduce>hadoop jar hadoop-mapreduce-examples-3.3.0.jar
An example program must be given as the first argument.
Valid program names are:
aggregatewordcount: An Aggregate based map/reduce program that counts the words in the input files.
aggregatewordhist: An Aggregate based map/reduce program that computes the histogram of the words in the input files.
bbp: A map/reduce program that uses Bailey-Borwein-Plouffe to compute exact digits of Pi.
dbcount: An example job that count the pageview counts from a database.
distbbp: A map/reduce program that uses a BBP-type formula to compute exact bits of Pi.
grep: A map/reduce program that counts the matches of a regex in the input.
join: A job that effects a join over sorted, equally partitioned datasets
multifilewc: A job that counts words from several files.
pentomino: A map/reduce tile laying program to find solutions to pentomino problems.
pi: A map/reduce program that estimates Pi using a quasi-Monte Carlo method.
randomtextwriter: A map/reduce program that writes 10GB of random textual data per node.
randomwriter: A map/reduce program that writes 10GB of random data per node.
secondarysort: An example defining a secondary sort to the reduce.
sort: A map/reduce program that sorts the data written by the random writer.
sudoku: A sudoku solver.
teragen: Generate data for the terasort
terasort: Run the terasort
teravalidate: Checking results of terasort
wordcount: A map/reduce program that counts the words in the input files.
wordmean: A map/reduce program that counts the average length of the words in the input files.
wordmedian: A map/reduce program that counts the median length of the words in the input files.
wordstandarddeviation: A map/reduce program that counts the standard deviation of the length of the words in the input files.
```

2. Executing word count program of map reduce.
3. Have a text file containing the words in multiple lines.

```
Deer Rabbit Fox Dog
Dog Rabbit Deer
Deer Deer Fox Dog
Rabbit Fox Rabbit
Deer Rabbit Fox
```

4. Start the hadoop clusters- nodes and yarn managers from hadoop/sbin directory.

```
.\start-dfs.cmd
```

```
.\start-yarn.cmd
```

5. To move the input file to hdfs root, Run the command from mapreduce directory-

```
\hadoop-3.3.0\share\hadoop\mapreduce> hadoop dfs -put
```

```
E:/RapidData/Learning/HadoopHive/hadoop-env/hadoop-3.3.0/examples/words.txt /
```

6. Check and display from hdfs-

```
\hadoop-3.3.0\share\hadoop\mapreduce> hadoop dfs -cat /words.txt
```

```
E:\a\Learning\HadoopHive\hadoop-env\hadoop-3.3.0\share\hadoop\mapreduce>hadoop dfs -cat /words.txt
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
Deer Rabbit Fox Dog
Dog Rabbit Deer
Deer Deer Fox Dog
Rabbit Fox Rabbit
Deer Rabbit Fox
```

7. Run the mapreduce program.

```
\hadoop-3.3.0\share\hadoop\mapreduce> hadoop jar hadoop-mapreduce-examples-3.3.0.jar  
wordcount /words.txt /FirstExampleOut
```

```

... \Learning\HadoopHive\hadoop-env\hadoop-3.3.0\share\hadoop\mapreduce>hadoop jar hadoop-mapreduce-examples-3.3.0.jar wordcount /words.txt /FirstExampleOut
2022-12-09 15:58:18,405 INFO client.DefaultHadoopFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2022-12-09 15:58:18,928 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/gayat/.staging/job_1670581651650_0002
2022-12-09 15:58:19,674 INFO input.FileInputFormat: Total input files to process : 1
2022-12-09 15:58:19,755 INFO mapreduce.JobSubmitter: number of splits:1
2022-12-09 15:58:20,328 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1670581651650_0002
2022-12-09 15:58:20,328 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-12-09 15:58:20,469 INFO conf.Configuration: resource-types.xml not found
2022-12-09 15:58:20,469 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-12-09 15:58:20,893 INFO impl.YarnClientImpl: Submitted application application_1670581651650_0002
2022-12-09 15:58:20,928 INFO mapreduce.Job: The url to track the job: http://RDLAP0014:8088/proxy/application_1670581651650_0002/
2022-12-09 15:58:20,929 INFO mapreduce.Job: Running job: job_1670581651650_0002
2022-12-09 15:58:29,073 INFO mapreduce.Job: Job job_1670581651650_0002 running in uber mode : false
2022-12-09 15:58:29,077 INFO mapreduce.Job: map 0% reduce 0%
2022-12-09 15:58:34,169 INFO mapreduce.Job: map 100% reduce 0%
2022-12-09 15:58:38,093 INFO mapreduce.Job: map 100% reduce 100%
2022-12-09 15:58:39,113 INFO mapreduce.Job: Job job_1670581651650_0002 completed successfully
2022-12-09 15:58:39,180 INFO mapreduce.Job: Counters: 54
  File System Counters
    FILE: Number of bytes read=62
    FILE: Number of bytes written=530625
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
    HDFS: Number of bytes read=186
    HDFS: Number of bytes written=36
    HDFS: Number of read operations=8
    HDFS: Number of large read operations=0
    HDFS: Number of write operations=2
    HDFS: Number of bytes read erasure-coded=0
  Job Counters
    Launched map tasks=1
    Launched reduce tasks=1
    Data-local map tasks=1
    Total time spent by all maps in occupied slots (ms)=2787
    Total time spent by all reduces in occupied slots (ms)=2468
    Total time spent by all map tasks (ms)=2787
    Total time spent by all reduce tasks (ms)=2468
    Total vcore-milliseconds taken by all map tasks=2787
    Total vcore-milliseconds taken by all reduce tasks=2468
    Total megabyte-milliseconds taken by all map tasks=2853888
    Total megabyte-milliseconds taken by all reduce tasks=2527232
  Map-Reduce Framework
    Map input records=5
    Map output records=17
    Map output bytes=155
    Map output materialized bytes=62
    Input split bytes=96

```

8. Check if the output file is available.

\hadoop-3.3.0\share\hadoop\mapreduce> **hadoop dfs -ls /FirstExampleOut**



9. Display the output from the file.

\hadoop-3.3.0\share\hadoop\mapreduce> **hadoop dfs -cat /FirstExampleOut/part-r-00000**

```

E:\      \Learning\HadoopHive\hadoop-env\hadoop-3.3.0\share\hadoop\mapreduce>hadoop dfs -cat /FirstExampleOut/part-r-00000
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
Deer    5
Dog     3
Fox     4
Rabbit  4
Rabit   1

```

Thus, the count of each occurrences of the words in the file are done.