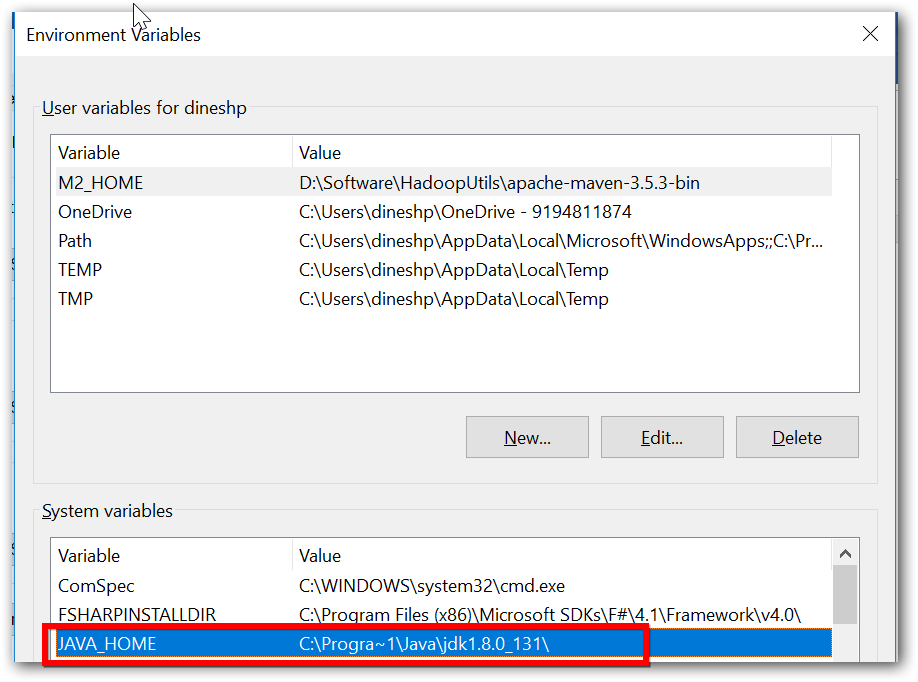
Hands on

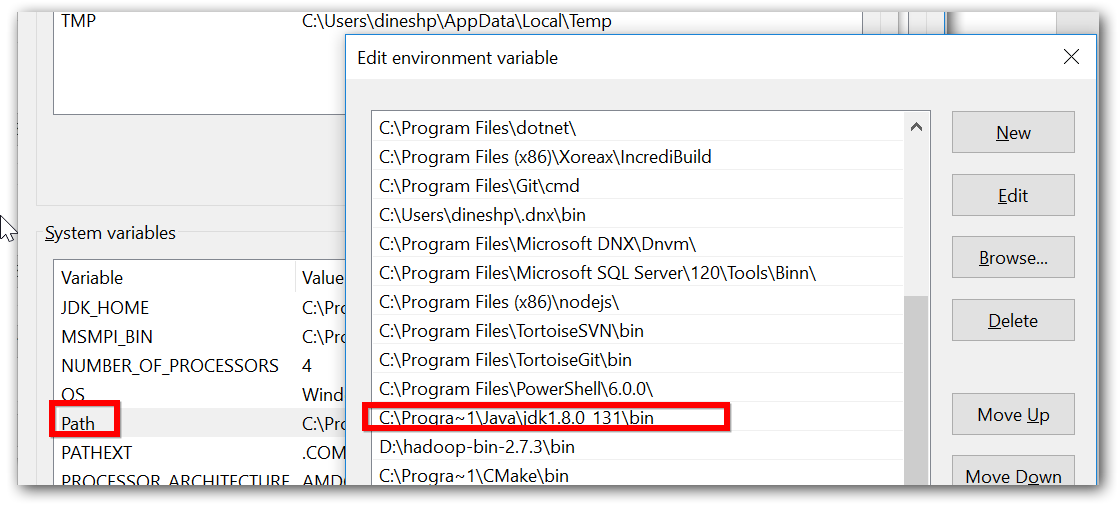
# Pre-requisites

1. Install **7z1801-x64** in your machine.
2. Untar **“hadoop-3.0.1.tar”** using 7z. Prefer any other drive apart from OS drive. For e.g. if ‘C:\’ is OS drive, have Hadoop package in ‘D:\’ or some other drive.
3. Install Java 8 in your machine.
4. Set JAVA\_HOME in environment variables – system variables

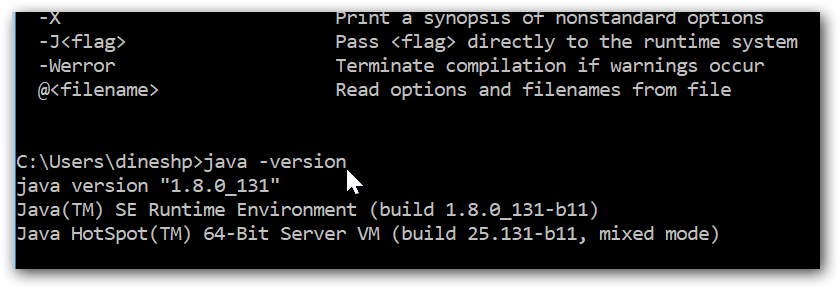
<https://confluence.atlassian.com/doc/setting-the-java_home-variable-in-windows-8895.html>



1. Add Java’s bin location PATH

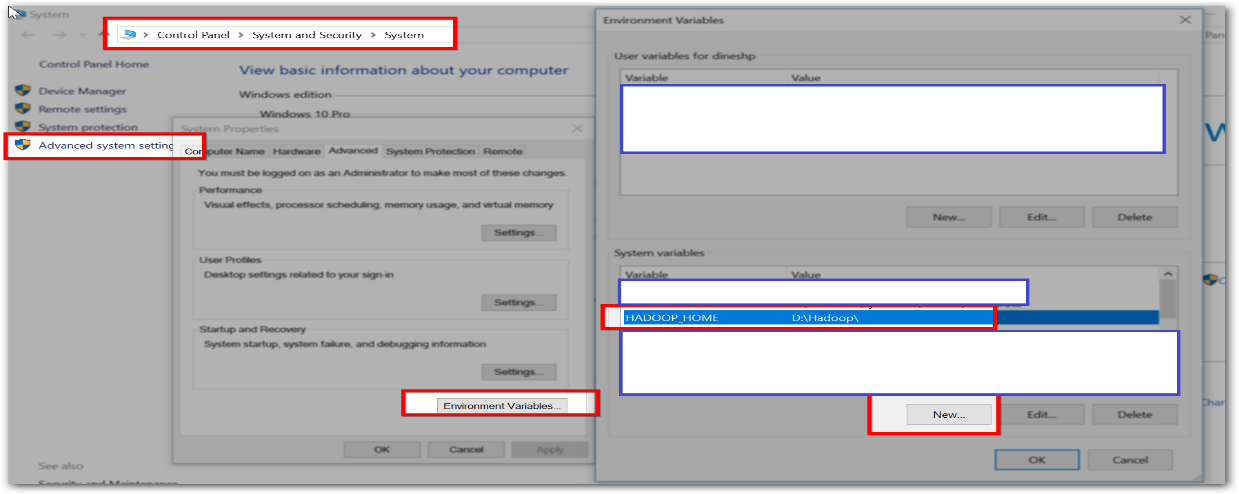


1. Ensure java home and path is set properly.
   1. Open command prompt
   2. Execute ‘javac’ or ‘java -version’ command.



1. Set HADOOP\_HOME as environment variables (without bin),

|  |  |
| --- | --- |
| HADOOP\_HOME | D:\hadoop\3.0.1 |



# Installation

Do the following changes in file specified in directory “..\..\hadoop-3.0.1\etc\hadoop”

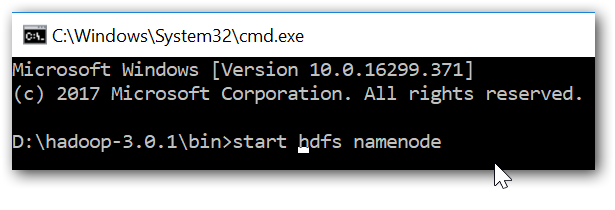
|  |  |  |
| --- | --- | --- |
| File name | Properties | Comments |
| hadoop-env.cmd | set HADOOP\_PREFIX=D:\hadoop-3.0.1  set HADOOP\_CONF\_DIR=%HADOOP\_PREFIX%\etc\hadoop  set YARN\_CONF\_DIR=%HADOOP\_CONF\_DIR%  set PATH=%PATH%;%HADOOP\_PREFIX%\bin | HADOOP\_PREFIX is the Hadoop package unzipped. |
| core-site.xml | <configuration>  <property>  <name>fs.defaultFS</name>  <value>hdfs://localhost:9000</value>  </property>  </configuration> |  |
| hdfs-site.xml | <configuration>  <property>  <name>dfs.replication</name>  <value>1</value>  </property>  <property>  <name>dfs.namenode.name.dir</name>  <value>file:///d:/Data/NameNode</value>  </property>  <property>  <name>dfs.datanode.data.dir</name>  <value> file:///d:/Data/DataNode</value>  </property>  </configuration> | Specify the directory where there won’t be any permission issues. |

## Start Hadoop cluster services

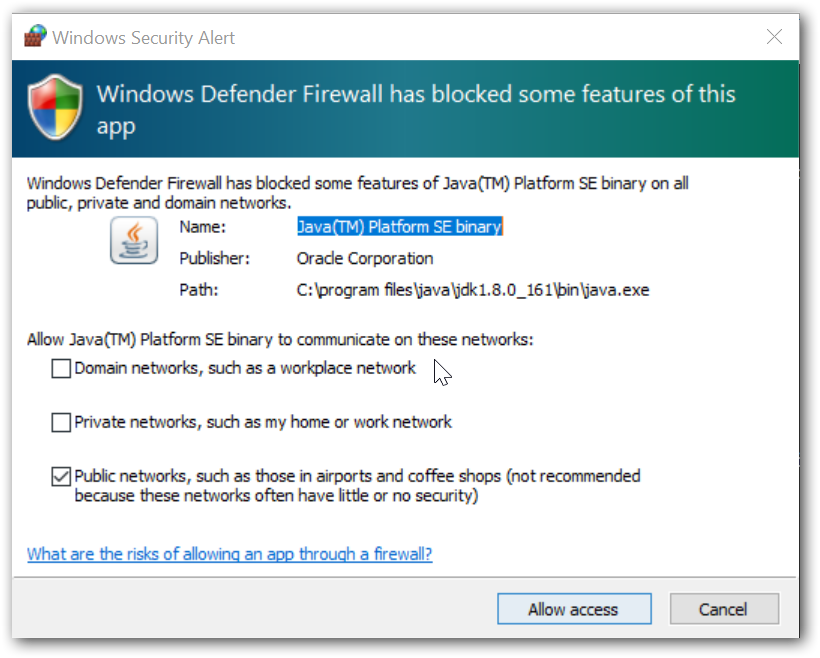
Open command prompt as ‘Administrator’ and navigate to Hadoop bin path,

1. Start NameNode

\bin> start hdfs namenode

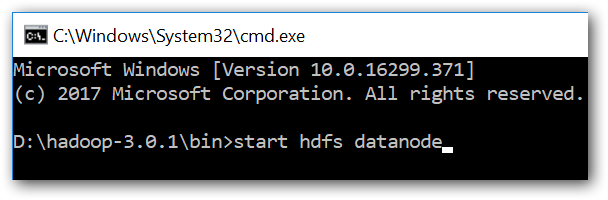


If such a popup is showed while starting service, click ‘Allow access’



1. Start DataNode

\bin> start hdfs datanode



1. Web UI - HDFS - <http://localhost:9870/dfshealth.html#tab-overview>

# Work-out

1. Create a file with the following content and do SaveAs with filename ‘Trip.csv’

|  |
| --- |
| vendorid,passenger\_count,trip\_distance,payment\_type,fare\_amount,tip\_amount,total\_amount  2,1,1.55,2,7.5,0,8.8  1,3,1.2,2,7,0,8.3  1,1,2.4,2,10.5,0,11.8  1,1,7.3,2,21.5,0,22.8  1,1,0.4,2,3.5,0,4.8  2,1,0.71,2,5,0,6.3  1,1,1.6,1,7.5,1.2,10  2,1,0.52,1,4,0.9,6.2  2,3,6.23,1,18.5,3.8,23.6 |

1. Create a file with following content and do SaveAs with filename ‘Vendor.csv’

|  |
| --- |
| vendorid,name  1,nyc  2,noc |

1. Create a file with following content and do SaveAs with filename ‘Payment.csv’

|  |
| --- |
| payment\_type,name  1,cash  2,creditcard  3,debitcard |

1. Open HDFS bin location in command prompt,
2. Create a folder called ‘Data’,
   1. Command: hdfs dfs -mkdir /Data
   2. Result:
3. Upload data into HDFS from local file system,
   1. Command: hdfs dfs -put "D:\Dataset\CSV\Customers.csv" /Data
   2. Result:
4. Word count,
   1. Command: yarn jar "..\..\hadoop-3.0.1\share\hadoop\mapreduce\hadoop-mapreduce-examples-3.0.1.jar" wordcount /Data /output
   2. Result:
5. Copy data between clusters,
   1. Command: hadoop distcp /Data /Data-Copy
   2. Result:
6. Web UI,
   1. HDFS - <http://localhost:9870/dfshealth.html#tab-overview>
   2. Resource Manager -