1. Install the journal node software on the journal nodes.

```
sudo yum install hadoop-hdfs-journalnode
```

1. Add the following to the core-site.xml file on the namenode:

```
<property>
   <name>dfs.journalnode.edits.dir</name>
   <value>/home/edureka/HA/data/jn</value>
  </property>
```

2. Add the following to the hdfs-site.xml file on all nodes.

```
<property>
  <name>dfs.namenode.name.dir</name>

  <value>some path here</value>

  </property>

  <property>

  <name>dfs.replication</name>

  <value>1</value>

  </property>

  <property>

  <name>dfs.permissions</name>

  <value>false</value>

  </property>

  <property>

  <name>dfs.nameservices</name>

  <value>ha-cluster</value>

  </property>

  <property>

  <name>dfs.ha.namenodes.your-cluster-name-here</name>

  <value>nn1,nn2</value>

  </property>

  <property>

  <name>dfs.namenode.rpc-address.your-cluster-name-here.nn1</name>

  <value>nn1.cluster.com:9000</value>
```

```
 </property>
 <property>
 <name>dfs.namenode.rpc-address.your-cluster-name-here.nn2</name>
 <value>your-cluster-name-here:9000</value>
 </property>
 <property>
 <name>dfs.namenode.http-address.ha-cluster.nn1</name>
 <value>nn1.cluster.com:50070</value>
 </property>
 <property>
 <name>dfs.namenode.http-address.your-cluster-name-here.nn2</name>
 <value>nn2.your domain name here.com:50070</value>
 </property>
 <property>
 <name>dfs.namenode.shared.edits.dir</name>
 <value>qjournal://nn1.your-domain-name-here.com:8485;nn2.your-domain-
name-here.com:8485;dn1.your-domain-name-here.com:8485/your-cluster-
name</value>
 </property>
 <property>
 <name>dfs.client.failover.proxy.provider.your-cluster-name</name>

<value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyPr
ovider</value>
 </property>
 <property>
 <name>dfs.ha.automatic-failover.enabled</name>
 <value>true</value>
 </property>
 <property>
 <name>ha.zookeeper.quorum</name>
```

```
 <value> nn1.cluster.com:2181,nn2.cluster.com:2181,dn1.cluster.com:2181
</value>
 </property>
 <property>
 <name>dfs.ha.fencing.methods</name>
 <value>sshfence</value>
 </property>
 <property>
 <name>dfs.ha.fencing.ssh.private-key-files</name>
 <value>/home/edureka/.ssh/id_rsa</value>
 </property>
```

3. Copy /usr/lib/zookeeper/conf/zoo_sample.cfg to /usr/lib/zookeeper/conf/zoo.cfg

In your zookeeper configuration file (/usr/lib/zookeeper/zoo.cfg) add the following lines:

Server.1=nn1.cluster.com:2888:3888

Server.2=nn2.cluster.com:2888:3888

Server.3=dn1.cluster.com:2888:3888

4. Create a secondary namenode (from Lab 2).

bootstrap the secondary namenode by running:

hdfs namenode -bootstrapStandby

5. Create a myid file for each journal node. Each file will have an integer

starting with 1, the next one will have 2, and so on. The file should be stored

in dataDir property path and have the filename myid

Now start the Zookeeper service on all the journal nodes.

zkServer.sh start

Start the journal node on all the journal servers.

Add the following environment variables:

export HADOOP_HOME=< Path to your Hadoop-2.6.0 directory>

export HADOOP_MAPRED_HOME=$HADOOP_HOME

export HADOOP_COMMON_HOME=$HADOOP_HOME

```
export HADOOP_HDFS_HOME=$HADOOP_HOME

export YARN_HOME=$HADOOP_HOME

export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop

export YARN_CONF_DIR=$HADOOP_HOME/etc/hadoop

export JAVA_HOME=<Path to your Java Directory>

export ZOOKEEPER_HOME =<Path to your Zookeeper Directory>

export PATH=$PATH: $JAVA_HOME/bin: $HADOOP_HOME/bin: $HADOOP_HOME/sbin:
$ZOOKEEPER_HOME/bin
```

Next, we format the ZooKeeper failover controller for both the namenode and the secondary.

hdfs zkfc-formatZK

Start the zkfc.

hadoop-daemon.sh start zkfc

Run haadmin to get the service state for the name node.

hdfs haadmin -getServiceState <your namenode>

Do this for all your name nodes.