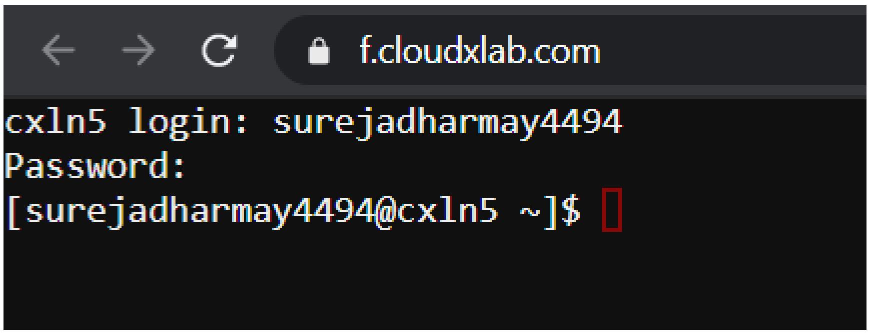
Practical-11

Perform ZOOKEEPER Command Line Interface (CLI) in CLoudXLab .

ZooKeeper Command Line Interface (CLI) is used to interact with the ZooKeeper ensemble which lets you perform simple, file-like operations. It is useful for debugging purposes. To perform ZooKeeper CLI operations, first start your ZooKeeper server and Client

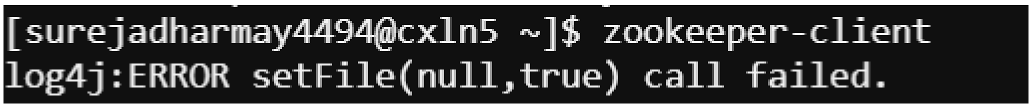
STEPS to run the Zookeeper on webconsole(in cloudxlab)

- First logged in webconsole using user name and password.



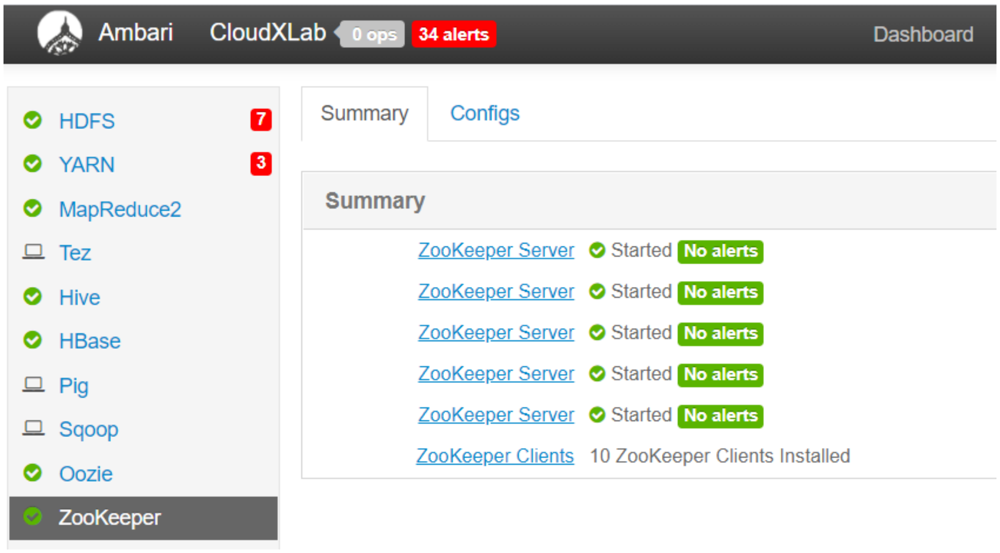
```
cxln5 login: surejadharmay4494
Password:
[surejadharmay4494@cxln5 ~]$
```

- Type the following command
  zookeeper-client



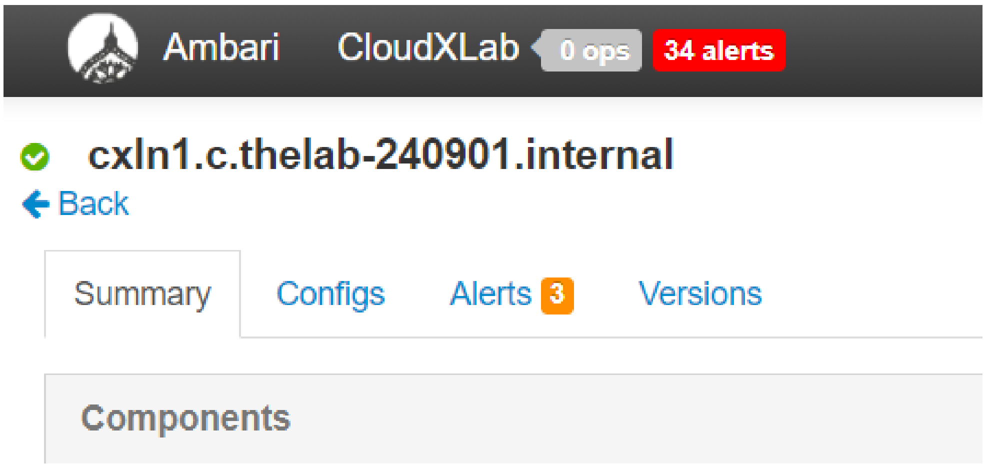
```
[surejadharmay4494@cxln5 ~]$ zookeeper-client
log4j:ERROR setFile(null,true) call failed.
```

- If you get any error then run the following command:
  1. First, you need to open the Ambari in cloudxlab then go to zookeeper and click on **"zookeeper Server under summery"** and copy **"cxln3.c.thelab-240901.internal"**



Summary

ZooKeeper Server Started No alerts
ZooKeeper Server Started No alerts
ZooKeeper Server Started No alerts
ZooKeeper Server Started No alerts
ZooKeeper Server Started No alerts
ZooKeeper Clients 10 ZooKeeper Clients Installed

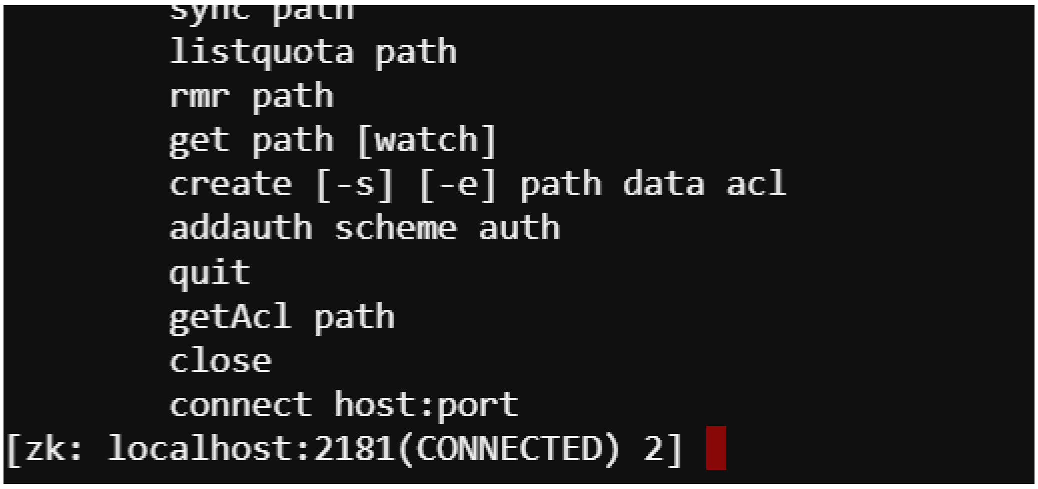


2. Run the following command on Webconsole.
   zookeeper-client-server cxln1.c.thelab-240901.internal

```
[surejadharmay4494@cxln5 ~]$ zookeeper-client-server cxln1.c.thelab-240901.internal
```

This would bring the zookeeper prompt . this is where we are going to type Zookeeper commands.

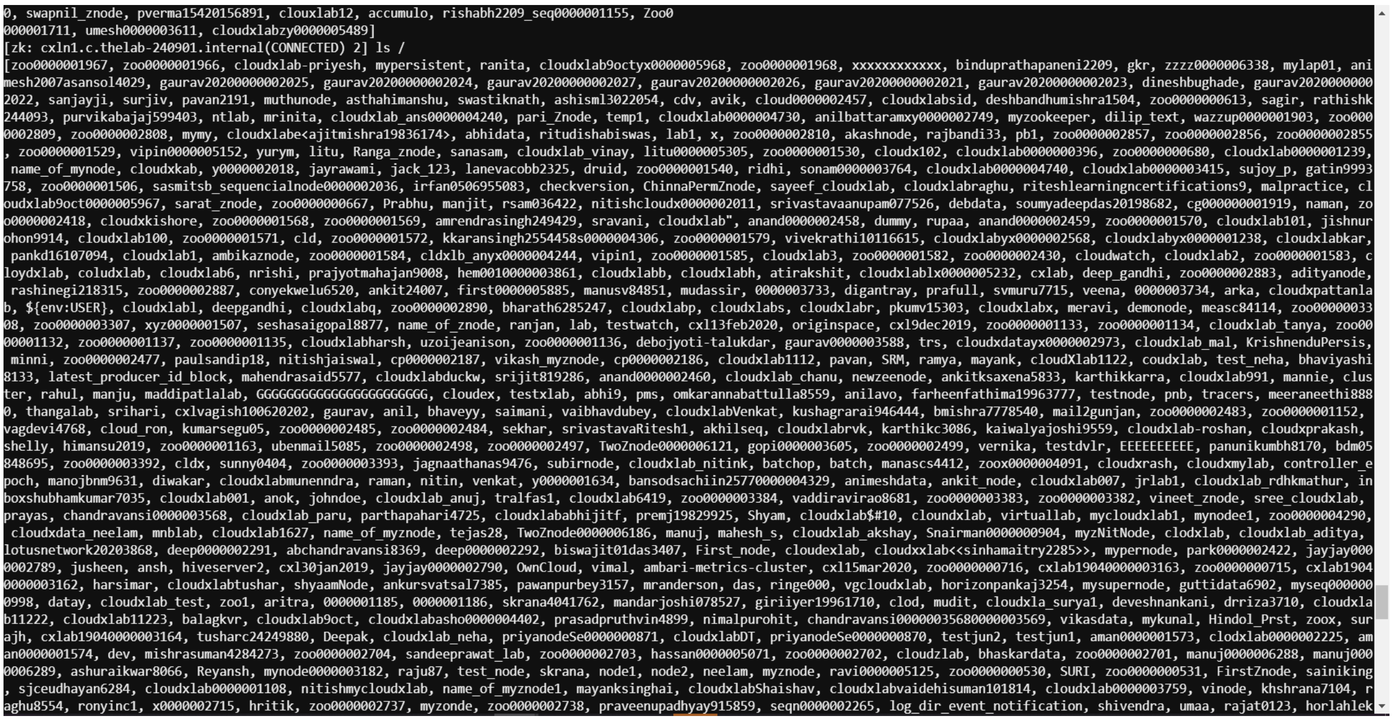
Output:



```
        sync path
        listquota path
        rmr path
        get path [watch]
        create [-s] [-e] path data acl
        addauth scheme auth
        quit
        getAcl path
        close
        connect host:port
[zk: localhost:2181(CONNECTED) 2]
```

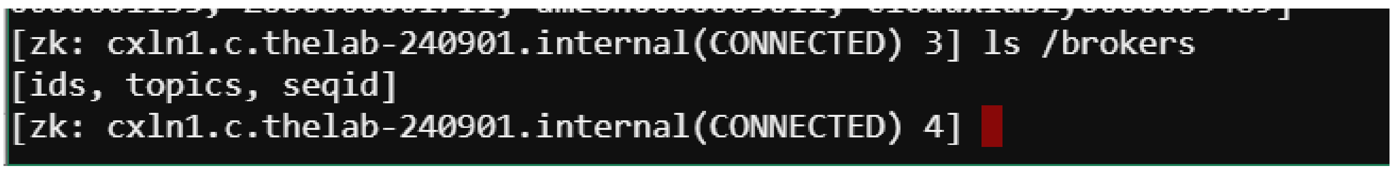
Perform following Commands

1. See the list of ZNodes at the top level
   ls /



```
0, swapnil_znode, pverma15420156891, clouxlab12, accumulo, rishabh2209_seq0000001155, Zoo0
000001711, umesh0000003611, cloudxlabzy0000005489]
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 2] ls /
[zoo0000001967, zoo0000001966, cloudxlab-priyesh, mypersistent, ranita, cloudxlab9octyx0000005968, zoo0000001968, xxxxxxxxxxx, binduprathapaneni2209, gkr, zzzz0000006338, mylap01, ani
mesh2007asansol4029, gaurav20200000002025, gaurav20200000002024, gaurav20200000002027, gaurav20200000002026, gaurav20200000002021, gaurav20200000002023, dineshbughade, gaurav2020000000
2022, sanjayji, surjiv, pavan2191, muthunode, asthahimanshu, swastiknath, ashisml3022054, cdv, avik, cloud0000002457, cloudxlabsid, deshbandhumishra1504, zoo0000000613, sagir, rathishk
244093, purvikabajaj599403, ntlab, mrinita, cloudxlab_ans0000004240, pari_Znode, temp1, cloudxlab0000004730, anilbattaramxy0000002749, myzookeeper, dilip_text, wazzup0000001903, zoo000
0002809, zoo0000002808, mymy, cloudxlabe<ajitmishra19836174>, abhidata, ritudishabiswas, lab1, x, zoo0000002810, akashnode, rajbandi33, pb1, zoo0000002857, zoo0000002856, zoo0000002855
, zoo0000001529, vipin0000005152, yurym, litu, Ranga_znode, sanasam, cloudxlab_vinay, litu0000005305, zoo0000001530, cloudx102, cloudxlab0000000396, zoo0000000680, cloudxlab0000001239,
 name_of_mynode, cloudxkab, y0000002018, jayrawami, jack_123, lanevacobb2325, druid, zoo0000001540, ridhi, sonam0000003764, cloudxlab0000004740, cloudxlab0000003415, sujoy_p, gatin9993
758, zoo0000001506, sasmitsb_sequencialnode0000002036, irfan0506955083, checkversion, ChinnaPermZnode, sayeef_cloudxlab, cloudxlabraghu, riteshlearningncertifications9, malpractice, cl
oudxlab9oct0000005967, sarat_znode, zoo0000000667, Prabhu, manjit, rsam036422, nitishcloudx0000002011, srivastavaanupam077526, debdata, soumyadeepdas20198682, cg000000001919, naman, zo
o0000002418, cloudxkishore, zoo0000001568, zoo0000001569, amrendrasingh249429, sravani, cloudxlab", anand0000002458, dummy, rupaa, anand0000002459, zoo0000001570, cloudxlab101, jishnur
ohon9914, cloudxlab100, zoo0000001571, cld, zoo0000001572, kkaransingh2554458s0000004306, zoo0000001579, vivekrathi10116615, cloudxlabyx0000002568, cloudxlabyx0000001238, cloudxlabkar,
 pankd16107094, cloudxlab1, ambikaznode, zoo0000001584, cldxlb_anyx0000004244, vipin1, zoo0000001585, cloudxlab3, zoo0000001582, zoo0000002430, cloudwatch, cloudxlab2, zoo0000001583, c
loydxlab, coludxlab, cloudxlab6, nrishi, prajyotmahajan9008, hem0010000003861, cloudxlabb, cloudxlabh, atirakshit, cloudxlablx0000005232, cxlab, deep_gandhi, zoo0000002883, adityanode,
 rashinegi218315, zoo0000002887, conyekwelu6520, ankit24007, first0000005885, manusv84851, mudassir, 0000003733, digantray, prafull, svmuru7715, veena, 0000003734, arka, cloudxpattanla
b, ${env:USER}, cloudxlabl, deepgandhi, cloudxlabq, zoo0000002890, bharath6285247, cloudxlabp, cloudxlabs, cloudxlabr, pkumv15303, cloudxlabx, meravi, demonode, measc84114, zoo00000033
08, zoo0000003307, xyz0000001507, seshasaigopal8877, name_of_znode, ranjan, lab, testwatch, cxl13feb2020, originspace, cxl9dec2019, zoo0000001133, zoo0000001134, cloudxlab_tanya, zoo00
00001132, zoo0000001137, zoo0000001135, cloudxlabharsh, uzoijeanison, zoo0000001136, debojyoti-talukdar, gaurav0000003588, trs, cloudxdatayx0000002973, cloudxlab_mal, KrishnenduPersis,
 minni, zoo0000002477, paulsandip18, nitishjaiswal, cp0000002187, vikash_myznode, cp0000002186, cloudxlab1112, pavan, SRM, ramya, mayank, cloudXlab1122, coudxlab, test_neha, bhaviyashi
8133, latest_producer_id_block, mahendrasaid5577, cloudxlabduckw, srijit819286, anand0000002460, cloudxlab_chanu, newzeenode, ankitksaxena5833, karthikkarra, cloudxlab991, mannie, clus
ter, rahul, manju, maddipatlalab, GGGGGGGGGGGGGGGGGGGGGGG, cloudex, testxlab, abhi9, pms, omkarannabattulla8559, anilavo, farheenfathima19963777, testnode, pnb, tracers, meeraneethi888
0, thangalab, srihari, cxlvagish100620202, gaurav, anil, bhaveyy, saimani, vaibhavdubey, cloudxlabVenkat, kushagrarai946444, bmishra7778540, mail2gunjan, zoo0000002483, zoo0000001152,
vagdevi4768, cloud_ron, kumarsegu05, zoo0000002485, zoo0000002484, sekhar, srivastavaRitesh1, akhilseq, cloudxlabrvk, karthikc3086, kaiwalyajoshi9559, cloudxlab-roshan, cloudxprakash,
shelly, himansu2019, zoo0000001163, ubenmail5085, zoo0000002498, zoo0000002497, TwoZnode0000006121, gopi0000003605, zoo0000002499, vernika, testdvlr, EEEEEEEEEE, panunikumbh8170, bdm05
848695, zoo0000003392, cldx, sunny0404, zoo0000003393, jagnaathanas9476, subirnode, cloudxlab_nitink, batchop, batch, manascs4412, zoox0000004091, cloudxrash, cloudxmylab, controller_e
poch, manojbnm9631, diwakar, cloudxlabmunenndra, raman, nitin, venkat, y0000001634, bansodsachiin25770000004329, animeshdata, ankit_node, cloudxlab007, jrlab1, cloudxlab_rdhkmathur, in
boxshubhamkumar7035, cloudxlab001, anok, johndoe, cloudxlab_anuj, tralfas1, cloudxlab6419, zoo0000003384, vaddiravirao8681, zoo0000003383, zoo0000003382, vineet_znode, sree_cloudxlab,
prayas, chandravansi0000003568, cloudxlab_paru, parthapahari4725, cloudxlababhijitf, premj19829925, Shyam, cloudxlab$#10, cloundxlab, virtuallab, mycloudxlab1, mynodee1, zoo0000004290,
 cloudxdata_neelam, mnblab, cloudxlab1627, name_of_myznode, tejas28, TwoZnode0000006186, manuj, mahesh_s, cloudxlab_akshay, Snairman0000000904, myzNitNode, clodxlab, cloudxlab_aditya,
lotusnetwork20203868, deep0000002291, abchandravansi8369, deep0000002292, biswajit01das3407, First_node, cloudexlab, cloudxxlab<<sinhamaitry2285>>, mypernode, park0000002422, jayjay000
0002789, jusheen, ansh, hiveserver2, cxl30jan2019, jayjay0000002790, OwnCloud, vimal, ambari-metrics-cluster, cxl15mar2020, zoo0000000716, cxlab19040000003163, zoo0000000715, cxlab1904
0000003162, harsimar, cloudxlabtushar, shyaamNode, ankursvatsal7385, pawanpurbey3157, mranderson, das, ringe000, vgcloudxlab, horizonpankaj3254, mysupernode, guttidata6902, myseq000000
0998, datay, cloudxlab_test, zoo1, aritra, 0000001185, 0000001186, skrana4041762, mandarjoshi078527, giriiyer19961710, clod, mudit, cloudxla_surya1, deveshnankani, drriza3710, cloudxla
b11222, cloudxlab11223, balagkvr, cloudxlab9oct, cloudxlabasho0000004402, prasadpruthvin4899, nimalpurohit, chandravansi00000035680000003569, vikasdata, mykunal, Hindol_Prst, zoox, sur
ajh, cxlab19040000003164, tusharc24249880, Deepak, cloudxlab_neha, priyanodeSe0000000871, cloudxlabDT, priyanodeSe0000000870, testjun2, testjun1, aman0000001573, clodxlab0000002225, am
an0000001574, dev, mishrasuman4284273, zoo0000002704, sandeeprawat_lab, zoo0000002703, hassan0000005071, zoo0000002702, cloudzlab, bhaskardata, zoo0000002701, manuj0000006288, manuj000
0006289, ashuraikwar8066, Reyansh, mynode0000003182, raju87, test_node, skrana, node1, node2, neelam, myznode, ravi0000005125, zoo0000000530, SURI, zoo0000000531, FirstZnode, sainiking
, sjceudhayan6284, cloudxlab0000001108, nitishmycloudxlab, name_of_myznode1, mayanksinghai, cloudxlabShaishav, cloudxlabvaidehisuman101814, cloudxlab0000003759, vinode, khshrana7104, r
aghu8554, ronyinc1, x0000002715, hritik, zoo0000002737, myzonde, zoo0000002738, praveenupadhyay915859, seqn0000002265, log_dir_event_notification, shivendra, umaa, rajat0123, horlahlek
```

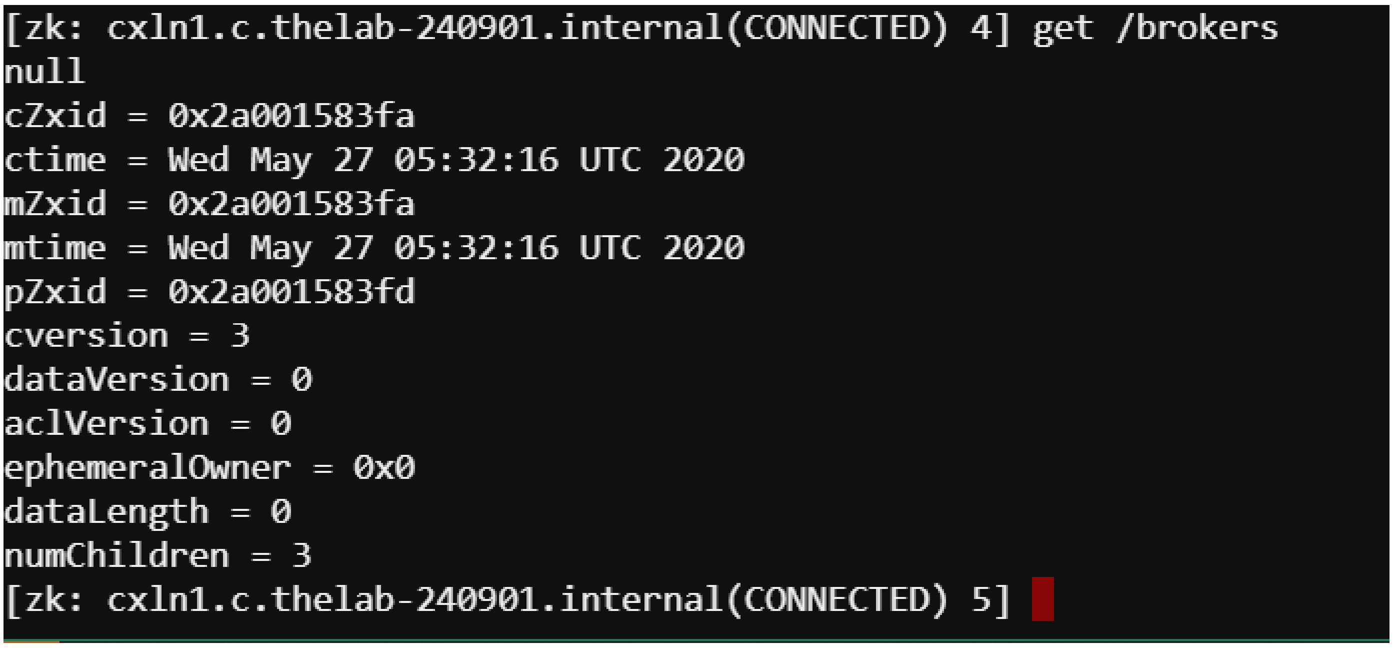
2. See the children of ZNodes under a node called “**brokers**” using the command .
   (You can find the name of znode from the above command)
   ls /brokers



```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 3] ls /brokers
[ids, topics, seqid]
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 4]
```

3. See the data inside ZNode “**brokers**”

   get /brokers

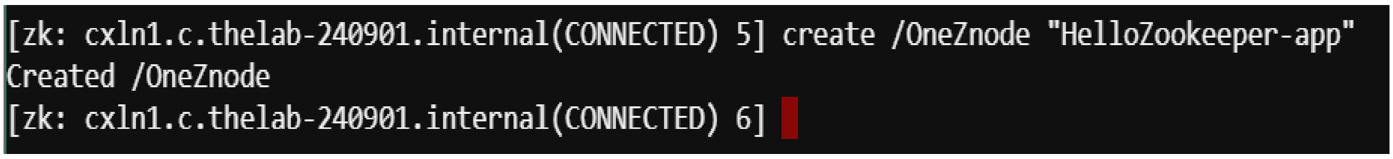


```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 4] get /brokers
null
cZxid = 0x2a001583fa
ctime = Wed May 27 05:32:16 UTC 2020
mZxid = 0x2a001583fa
mtime = Wed May 27 05:32:16 UTC 2020
pZxid = 0x2a001583fd
cversion = 3
dataVersion = 0
aclVersion = 0
ephemeralOwner = 0x0
dataLength = 0
numChildren = 3
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 5]
```

4. Create ZNodes

   <u>Syntax :</u> create /path /data

   Create /OneZnode “HelloZookeeper-app”



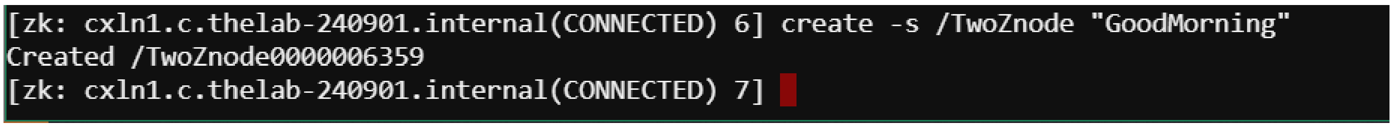
```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 5] create /OneZnode "HelloZookeeper-app"
Created /OneZnode
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 6]
```

5. create a Sequential znode, add -s flag

   <u>Syntax:</u> create -s /path /data

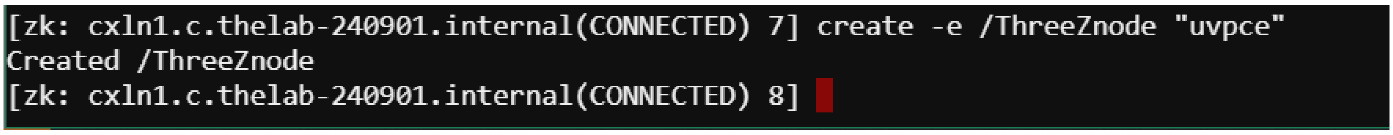
   create -s /TwoZnode “GoodMorning”

NOTE: Sequential znodes guaranty that the znode path will be unique. ZooKeeper ensemble will add sequence number along with 10 digit padding to the znode path. For example, the znode path /myapp will be converted to /myapp0000000001 and the next sequence number will be /myapp0000000002. If no flags are specified, then the znode is considered as persistent.



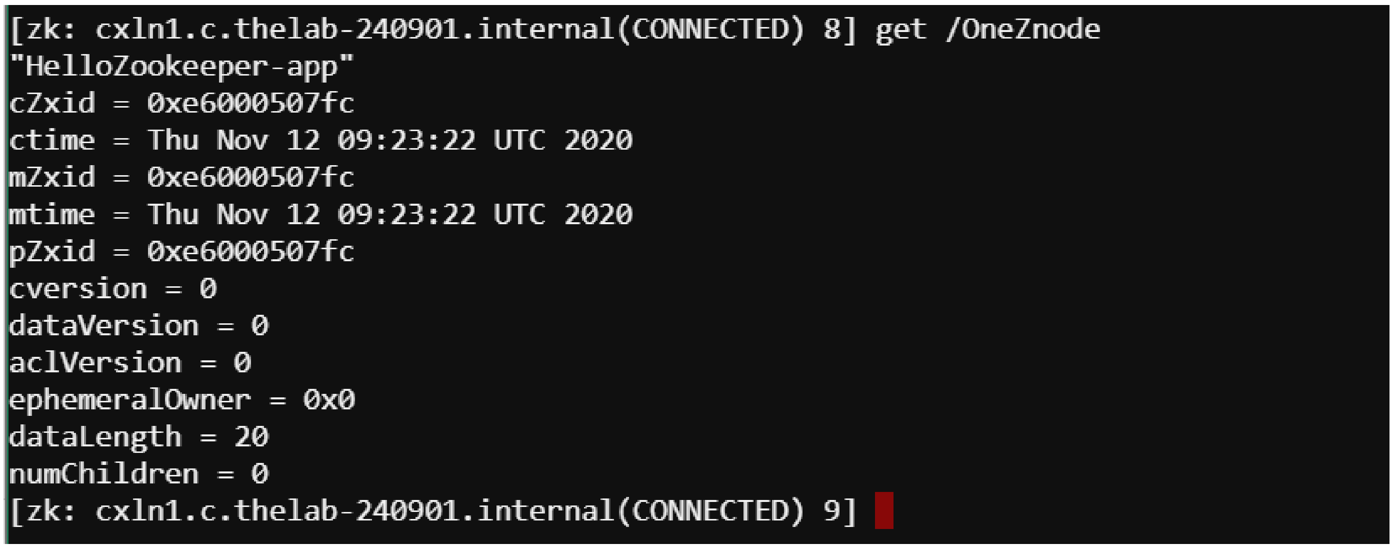
```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 6] create -s /TwoZnode "GoodMorning"
Created /TwoZnode0000006359
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 7]
```

6. Create an Ephemeral Znode, add -e flag as shown below.
   __Syntax:__ **create -e /path /data**
   create -e /ThreeZnode "uvpce"
   NOTE: Ephemeral znodes (flag: e) will be automatically deleted when a session expires or when the client disconnects. Remember when a client connection is lost, the ephemeral znode will be deleted. You can try it by quitting the ZooKeeper Client and then re-opening the Client.



```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 7] create -e /ThreeZnode "uvpce"
Created /ThreeZnode
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 8]
```

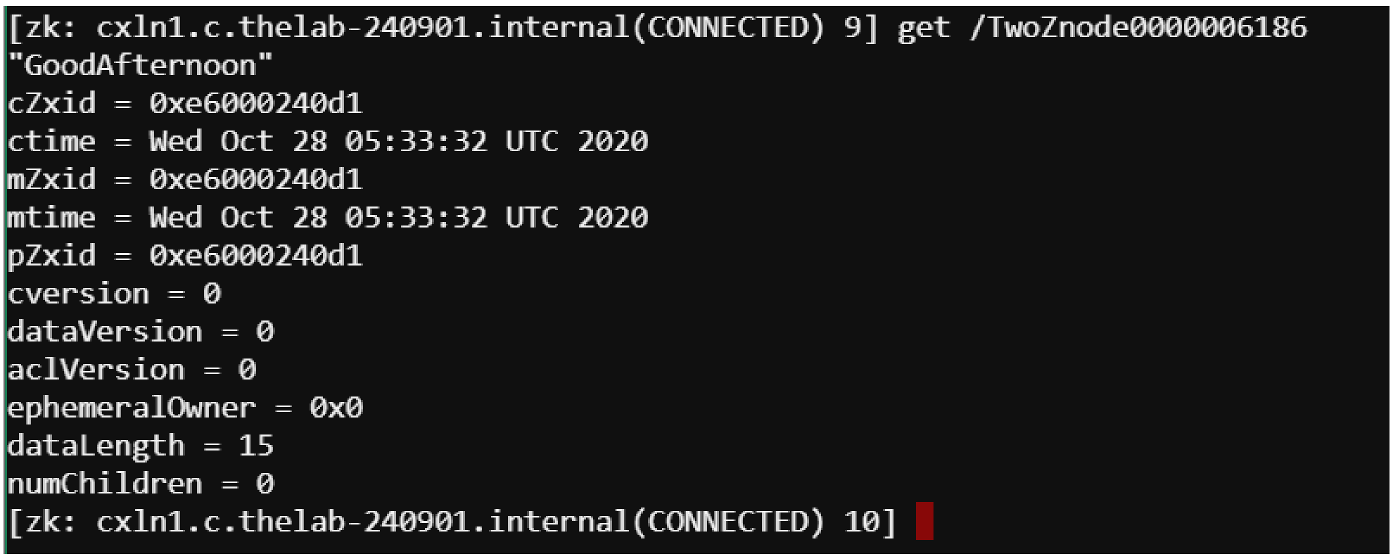
7. Get Data
   __Syntax:__ **get /path**
   get /OneZnode



```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 8] get /OneZnode
"HelloZookeeper-app"
cZxid = 0xe6000507fc
ctime = Thu Nov 12 09:23:22 UTC 2020
mZxid = 0xe6000507fc
mtime = Thu Nov 12 09:23:22 UTC 2020
pZxid = 0xe6000507fc
cversion = 0
dataVersion = 0
aclVersion = 0
ephemeralOwner = 0x0
dataLength = 20
numChildren = 0
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 9]
```

get /TwoZnode0000000023 (To access a sequential znode, you must enter the full path of the znode.)

Output:



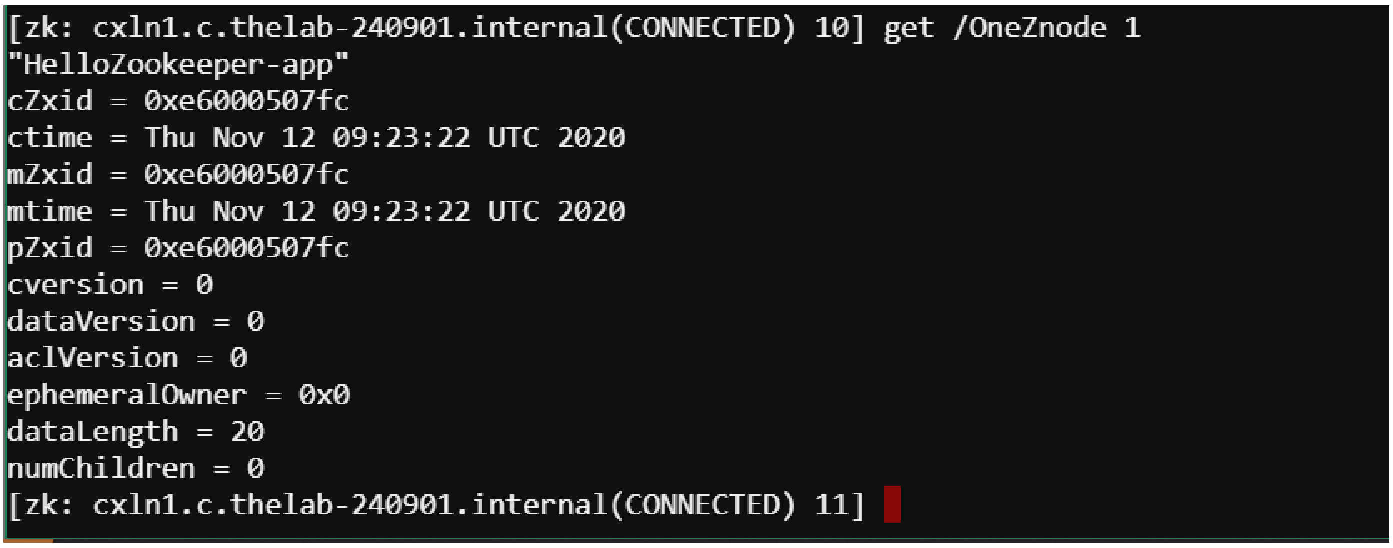
```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 9] get /TwoZnode0000006186
"GoodAfternoon"
cZxid = 0xe6000240d1
ctime = Wed Oct 28 05:33:32 UTC 2020
mZxid = 0xe6000240d1
mtime = Wed Oct 28 05:33:32 UTC 2020
pZxid = 0xe6000240d1
cversion = 0
dataVersion = 0
aclVersion = 0
ephemeralOwner = 0x0
dataLength = 15
numChildren = 0
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 10]
```

NOTE: It returns the associated data of the znode and metadata of the specified znode. You will get information such as when the data was last modified, where it was modified, and information about the data. This CLI is also used to assign watches to show notification about the data.

NOTE: Below show the output of "get" command and description.

my_data : This line of text is the data that we stored in the znode.
cZxid = 0x8 : The zxid (ZooKeeper Transaction Id) of the change that caused this znode to be created.
ctime = Mon Nov 30 18:41:06 IST 2015 : The time when this znode was created.
mZxid = 0x8 : The zxid of the change that last modified this znode.
mtime = Mon Nov 30 18:41:06 IST 2015 : The time when this znode was last modified.
pZxid = 0x8 : The zxid of the change that last modified children of this znode.
cversion = 0 : The number of changes to the children of this znode.
dataVersion = 0 : The number of changes to the data of this znode.
aclVersion = 0 : The number of changes to the ACL of this znode.
ephemeralOwner = 0x0: The session id of the owner of this znode if the znode is an ephemeral node. If it is not an ephemeral node, it will be zero.
dataLength = 7 : The length of the data field of this znode.
numChildren = 0 : The number of children of this znode

8. Watch
   <u>**Syntax :**</u> **get /path [watch] 1**
   Get /OneZnode1



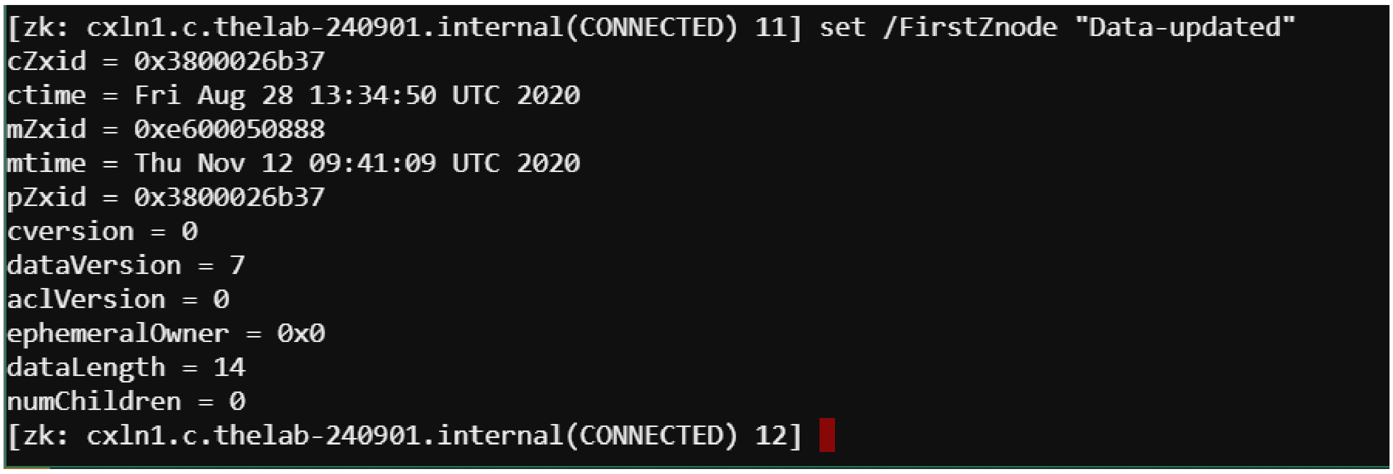
```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 10] get /OneZnode 1
"HelloZookeeper-app"
cZxid = 0xe6000507fc
ctime = Thu Nov 12 09:23:22 UTC 2020
mZxid = 0xe6000507fc
mtime = Thu Nov 12 09:23:22 UTC 2020
pZxid = 0xe6000507fc
cversion = 0
dataVersion = 0
aclVersion = 0
ephemeralOwner = 0x0
dataLength = 20
numChildren = 0
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 11]
```

Note: Watches show a notification when the specified znode or znode's children data changes. You

can set a watch only in get command. The output is similar to normal get command, but it will

wait for znode changes in the background.

9. Set Data
   <u>**Syntax:**</u> **set /path /data**
   set /FirstZnode "Data-updated"



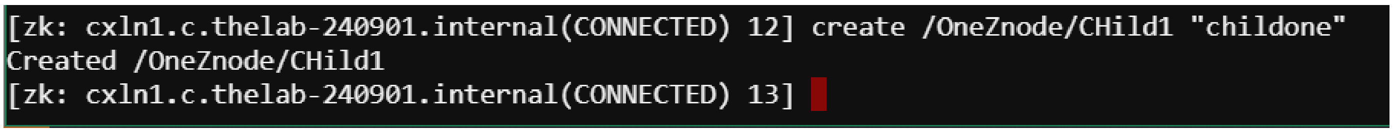
```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 11] set /FirstZnode "Data-updated"
cZxid = 0x3800026b37
ctime = Fri Aug 28 13:34:50 UTC 2020
mZxid = 0xe600050888
mtime = Thu Nov 12 09:41:09 UTC 2020
pZxid = 0x3800026b37
cversion = 0
dataVersion = 7
aclVersion = 0
ephemeralOwner = 0x0
dataLength = 14
numChildren = 0
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 12]
```

NOTE: Set the data of the specified znode. Once you finish this set operation, you can check the data using the get CLI command.

10. Create Children / Sub-znode

    <u>Syntax:</u> create /parent/path/subnode/path /data

    create /OneZnode/Child1 “childone”



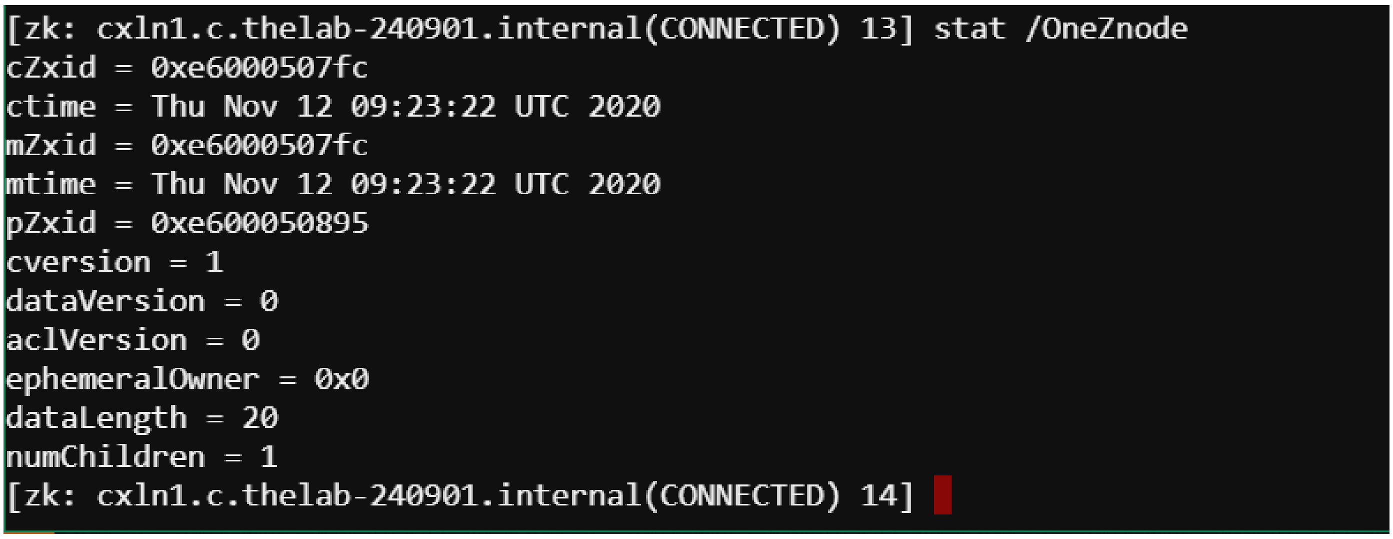
```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 12] create /OneZnode/CHild1 "childone"
Created /OneZnode/CHild1
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 13]
```

Note: Creating children is similar to creating new nodes. The only difference is that the path of the child znode will have the parent path as well.

11. Check Status

    <u>Syntax:</u> stat /path

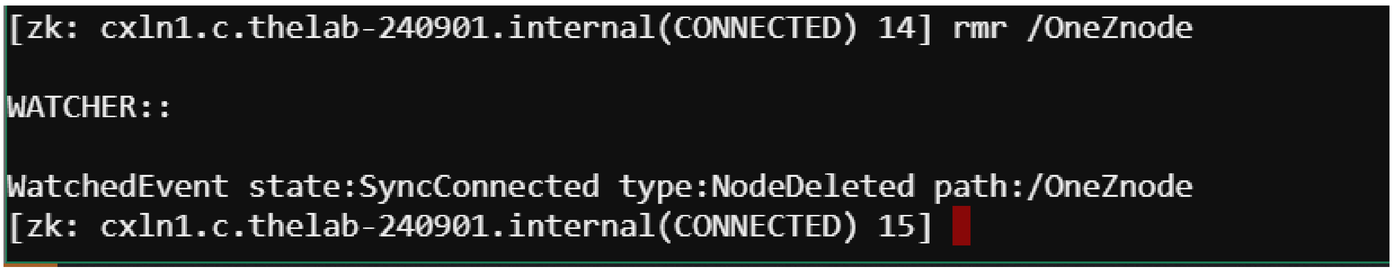
    stat /OneZnode



```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 13] stat /OneZnode
cZxid = 0xe6000507fc
ctime = Thu Nov 12 09:23:22 UTC 2020
mZxid = 0xe6000507fc
mtime = Thu Nov 12 09:23:22 UTC 2020
pZxid = 0xe600050895
cversion = 1
dataVersion = 0
aclVersion = 0
ephemeralOwner = 0x0
dataLength = 20
numChildren = 1
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 14]
```

Note: Status describes the metadata of a specified znode. It contains details such as Timestamp, Version number, ACL, Data length, and Children znode.

12. Remove a Znode
    Syntax: rmr /path
    rmr /OneZnode



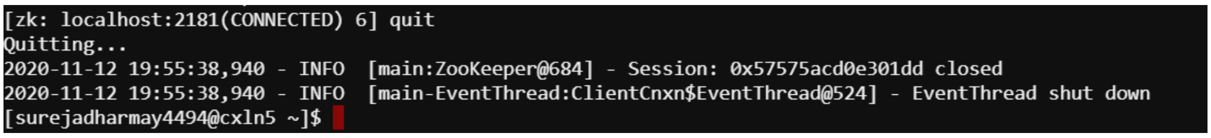
```
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 14] rmr /OneZnode

WATCHER::

WatchedEvent state:SyncConnected type:NodeDeleted path:/OneZnode
[zk: cxln1.c.thelab-240901.internal(CONNECTED) 15]
```

Note: Delete (delete /path) command is similar to remove command, except the fact that it works only on znodes with no children.

13. Quit
    quit



```
[zk: localhost:2181(CONNECTED) 6] quit
Quitting...
2020-11-12 19:55:38,940 - INFO  [main:ZooKeeper@684] - Session: 0x57575acd0e301dd closed
2020-11-12 19:55:38,940 - INFO  [main-EventThread:ClientCnxn$EventThread@524] - EventThread shut down
[surejadharmay4494@cxln5 ~]$
```