Hewlett Packard Enterprise

# SMOKE TEST

CDH 6.1.0

Date Prepared: Sept 2019

Hewlett Packard Enterprise

Document Information

| Project Name | **EPIC Accelerator Deployment & Integration Services** | | |
|---|---|---|---|
| Project Owner | | Document Version No | 0.1 |
| Quality Review Method | | | |
| Prepared By | Priyanka | Preparation Date | Sept 2019 |
| Reviewed By | | Review Date | |

## Table of Contents



## Table of Tables

**NO TABLE OF FIGURES ENTRIES FOUND.**

# 1 TESTING HDFS

In this section, we will test HDFS.

1. SSH to any HDFS DataNode/ worker (Use the instance IP from EPIC cluster page)

2. To list all the files and directory

```
hdfs dfs -ls /
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -ls /
Found 2 items
drwxrwxrwt   - hdfs supergroup          0 2019-09-26 22:25 /tmp
drwxrwxrwx   - hdfs supergroup          0 2019-09-26 22:24 /user
[bluedata@bluedata-85 ~]$
```

3. To create a directory, execute
   a. Create a directory in /user directory

```
hdfs dfs –mkdir /user/test
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -mkdir /user/test
[bluedata@bluedata-85 ~]$
```

   b. To verify, do ls

```
hdfs dfs –ls /user
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -ls /user
Found 6 items
drwx------   - bluedata supergroup          0 2019-09-30 09:34 /user/bluedata
drwxrwxrwx   - mapred   hadoop              0 2019-09-26 22:24 /user/history
drwxrwxr-t   - hive     hive                0 2019-09-26 22:24 /user/hive
drwxr-x--x   - spark    spark               0 2019-09-26 22:24 /user/spark
drwxr-xr-x   - bluedata supergroup          0 2019-10-08 23:14 /user/test
drwxr-xr-x   - hdfs     supergroup          0 2019-09-26 22:24 /user/yarn
[bluedata@bluedata-85 ~]$
```

4. To create a file with file size 0 bytes
   a. Create file using **touchz**

```
hdfs dfs –touchz /user/test/tesing
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -touchz /user/test/testing
[bluedata@bluedata-85 ~]$
```

b. Verify

```
hdfs dfs -ls /user/test
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -ls /user/test
Found 1 items
-rw-r--r--   3 bluedata supergroup          0 2019-10-08 23:22 /user/test/testing
[bluedata@bluedata-85 ~]$
```

**Or**

```
hdfs dfs -du -s /user/test/testing
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -du -s /user/test/testing
0  0  /user/test/testing
[bluedata@bluedata-85 ~]$
```

5. To copy an existing file, execute

```
hdfs dfs -put /home/bluedata/sample.txt /user/test
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -put /home/bluedata/sample.txt /user/test
[bluedata@bluedata-85 ~]$
```

**Or**

```
hdfs dfs -copyFromLocal /home/bluedata/sample.txt /user/test
```

**Note:** The file **sample.txt** is present in local path. Here we are coping to HDFS path.

6. To view the content of a file, execute

```
hdfs dfs -cat /user/test/sample.txt
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -cat /user/test/sample.txt
It takes a great deal of bravery to stand up to our enemies,
but just as much to stand up to our friends.
[bluedata@bluedata-85 ~]$
```

**Or**

```
hdfs dfs -text /user/test/sample.txt
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -text /user/test/sample.txt
It takes a great deal of bravery to stand up to our enemies,
but just as much to stand up to our friends.
```

7. To count the number of directories, files, and bytes of a directory, execute

```
hdfs dfs -count /user/test
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -count /user/test
           1            2                106 /user/test
[bluedata@bluedata-85 ~]$
```

8. To remove a file

```
hdfs dfs -rm /user/test/testing
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -rm /user/test/testing
19/10/09 00:29:13 INFO fs.TrashPolicyDefault: Moved: 'hdfs://bluedata-81.dev.team.bdlocal:8020/user/test/testing'
to trash at: hdfs://bluedata-81.dev.team.bdlocal:8020/user/bluedata/.Trash/Current/user/test/testing1570606153113
[bluedata@bluedata-85 ~]$
[bluedata@bluedata-85 ~]$
```

9. To remove entire directory and all its content

```
hdfs dfs -rm -r /user/test
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -rm -r /user/test
19/10/09 00:34:41 INFO fs.TrashPolicyDefault: Moved: 'hdfs://bluedata-81.dev.team.bdlocal:8020/user/test' to trash
at: hdfs://bluedata-81.dev.team.bdlocal:8020/user/bluedata/.Trash/Current/user/test
[bluedata@bluedata-85 ~]$
[bluedata@bluedata-85 ~]$
```

10. To get help from HDFS

```
hdfs dfs -help
```

```
[bluedata@bluedata-85 ~]$ hdfs dfs -help
Usage: hadoop fs [generic options]
        [-appendToFile <localsrc> ... <dst>]
        [-cat [-ignoreCrc] <src> ...]
        [-checksum <src> ...]
        [-chgrp [-R] GROUP PATH...]
        [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
        [-chown [-R] [OWNER][:[GROUP]] PATH...]
        [-copyFromLocal [-f] [-p] [-l] [-d] [-t <thread count>] <localsrc> ... <dst>]
        [-copyToLocal [-f] [-p] [-ignoreCrc] [-crc] <src> ... <localdst>]
        [-count [-q] [-h] [-v] [-t [<storage type>]] [-u] [-x] [-e] <path> ...]
        [-cp [-f] [-p | -p[topax]] [-d] <src> ... <dst>]
        [-createSnapshot <snapshotDir> [<snapshotName>]]
        [-deleteSnapshot <snapshotDir> <snapshotName>]
        [-df [-h] [<path> ...]]
        [-du [-s] [-h] [-v] [-x] <path> ...]
        [-expunge]
        [-find <path> ... <expression> ...]
        [-get [-f] [-p] [-ignoreCrc] [-crc] <src> ... <localdst>]
        [-getfacl [-R] <path>]
        [-getfattr [-R] {-n name | -d} [-e en] <path>]
        [-getmerge [-nl] [-skip-empty-file] <src> <localdst>]
        [-help [cmd ...]]
        [-ls [-C] [-d] [-h] [-q] [-R] [-t] [-S] [-r] [-u] [-e] [<path> ...]]
        [-mkdir [-p] <path> ...]
        [-moveFromLocal <localsrc> ... <dst>]
        [-moveToLocal <src> <localdst>]
        [-mv <src> ... <dst>]
        [-put [-f] [-p] [-l] [-d] <localsrc> ... <dst>]
        [-renameSnapshot <snapshotDir> <oldName> <newName>]
        [-rm [-f] [-r|-R] [-skipTrash] [-safely] <src> ...]
        [-rmdir [--ignore-fail-on-non-empty] <dir> ...]
        [-setfacl [-R] [{-b|-k} {-m|-x <acl_spec>} <path>]|[--set <acl_spec> <path>]]
        [-setfattr {-n name [-v value] | -x name} <path>]
        [-setrep [-R] [-w] <rep> <path> ...]
        [-stat [format] <path> ...]
        [-tail [-f] <file>]
        [-test -[defsz] <path>]
        [-text [-ignoreCrc] <src> ...]
        [-touch [-a] [-m] [-t TIMESTAMP ] [-c] <path> ...]
        [-touchz <path> ...]
        [-truncate [-w] <length> <path> ...]
        [-usage [cmd ...]]

-appendToFile <localsrc> ... <dst> :
  Appends the contents of all the given local files to the given dst file. The dst
  file will be created if it does not exist. If <localSrc> is -, then the input is
```

11. To get help for individual command

```
hdfs dfs -usage appendToFile
```

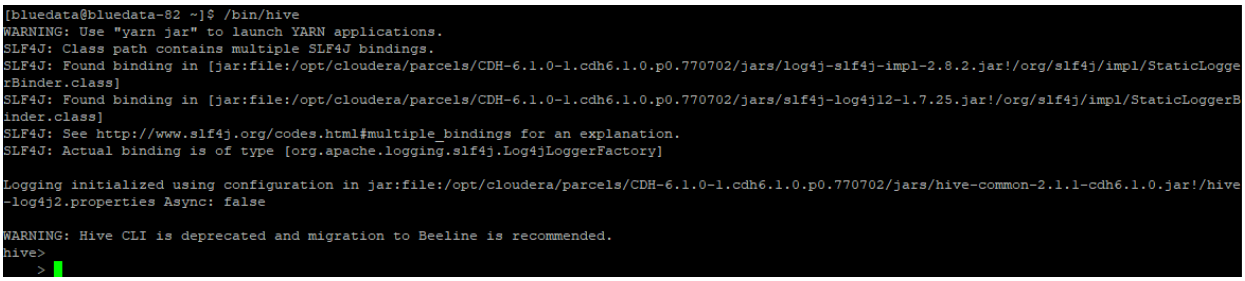
```
[bluedata@bluedata-85 ~]$ hdfs dfs -usage appendToFile
Usage: hadoop fs [generic options] -appendToFile <localsrc> ... <dst>
[bluedata@bluedata-85 ~]$
```

# 2 TESTING HIVE

In this section, we will test Hive.

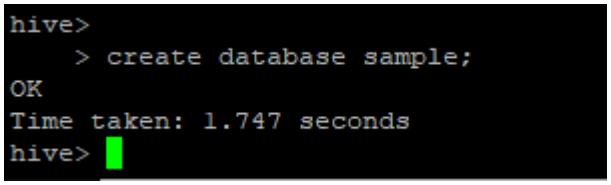
1. To enter into Hive shell prompt, execute

```
/bin/hive
```



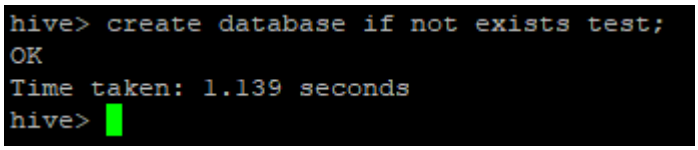

2. To create a database “sample”

```
create database sample;
```




3. Another way to create a database is checking if it exists already

```
create database if not exists test;
```




4. To add description while creating a database, execute

```
create database testing comment "This is just a test database";
```

```
hive> create database testing comment "This is just a test database";
OK
Time taken: 0.073 seconds
hive>
```

5. Using DBPROPERTIES while creating database

```
create database extendedinfo with DBPROPERTIES
('createdby'='Admin','createdfor'='users');
```

```
hive> create database extendedinfo with DBPROPERTIES ('createdby'='Admin','createdfor'='users');
OK
Time taken: 0.07 seconds
hive>
```

6. To check all existing databases, execute

```
show databases;
```

```
hive> show databases;
OK
default
extendedinfo
sample
test
testing
Time taken: 0.127 seconds, Fetched: 5 row(s)
hive>
```

7. To search for databases containing a pattern, execute

```
show databases like 'test*';
```

```
hive> show databases like 'test*';
OK
test
testing
Time taken: 0.082 seconds, Fetched: 2 row(s)
hive>
```

8. To view the description of the database, use the describe command

```
describe database testing;
```

```
hive> describe database testing;
OK
testing This is just a test database    hdfs://bluedata-81.dev.team.bdlocal:8020/user/hive/warehouse/testing.db bluedata        USER
Time taken: 0.03 seconds, Fetched: 1 row(s)
hive>
```

9. Use the extend command to view the other details of the database

```
describe database extended extendedinfo;
```

```
hive> describe database extended extendedinfo;
OK
extendedinfo            hdfs://bluedata-81.dev.team.bdlocal:8020/user/hive/warehouse/extendedinfo.db    bluedata        USER    {createdby=Admin, createdfor=users}
Time taken: 0.034 seconds, Fetched: 1 row(s)
hive>
```

10. To use a database, execute

```
use sample;
```

```
hive> use sample;
OK
Time taken: 0.035 seconds
hive>
```

11. To create a table, execute

```
create table courses(course_id int,course_name
string,students_enrolled int);
```

```
hive> create table courses(course_id int,course_name string,students_enrolled int);
OK
Time taken: 0.305 seconds
hive>
```

12. Insert data in the created table

```
INSERT INTO TABLE courses VALUES (1,'Hadoop',5500);
```

```
hive> INSERT INTO TABLE courses VALUES (1,'Hadoop',5500);
Query ID = bluedata_20190930075947_7729a197-124e-455e-95e8-56b8d345946b
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
19/09/30 07:59:48 INFO client.RMProxy: Connecting to ResourceManager at bluedata-81.dev.team.bdlocal/172.18.0.27:8032
19/09/30 07:59:48 INFO client.RMProxy: Connecting to ResourceManager at bluedata-81.dev.team.bdlocal/172.18.0.27:8032
Starting Job = job_1569562019118_0001, Tracking URL = http://bluedata-81.dev.team.bdlocal:8088/proxy/application_1569562019118_0001/
Kill Command = /opt/cloudera/parcels/CDH-6.1.0-1.cdh6.1.0.p0.770702/lib/hadoop/bin/hadoop job  -kill job_1569562019118_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2019-09-30 07:59:58,971 Stage-1 map = 0%,  reduce = 0%
2019-09-30 08:00:07,254 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.33 sec
MapReduce Total cumulative CPU time: 2 seconds 330 msec
Ended Job = job_1569562019118_0001
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://bluedata-81.dev.team.bdlocal:8020/user/hive/warehouse/sample.db/courses/.hive-staging_hive_2019-09-30_
2-1/-ext-10000
Loading data to table sample.courses
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Cumulative CPU: 2.33 sec   HDFS Read: 4724 HDFS Write: 84 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 330 msec
OK
Time taken: 21.375 seconds
hive>
```

13. To view all inserted records, use select command

```
select * from courses;
```

```
hive> select * from courses;
OK
1       Hadoop  5500
Time taken: 0.107 seconds, Fetched: 1 row(s)
hive>
```

14. View the schema of the table

```
describe courses;
```

```
hive> describe courses;
OK
course_id               int
course_name             string
students_enrolled       int
Time taken: 0.066 seconds, Fetched: 3 row(s)
hive>
```

# 3 TESTING YARN

In this section, we will test Yarn.

1. Check version

```
yarn version
```

```
[bluedata@bluedata-81 ~]$ yarn version
WARNING: YARN_OPTS has been replaced by HADOOP_OPTS. Using value of YARN_OPTS.
Hadoop 3.0.0-cdh6.1.0
Source code repository http://github.com/cloudera/hadoop -r b8dd3044ff414ac0bf14b77ab23d55ca291464a9
Compiled by jenkins on 2018-12-07T01:00Z
Compiled with protoc 2.5.0
From source with checksum 25f1e186cc43e44704f8d99c6c7bec
This command was run using /opt/cloudera/parcels/CDH-6.1.0-1.cdh6.1.0.p0.770702/jars/hadoop-common-3.0.0-cdh6.1.0.jar
[bluedata@bluedata-81 ~]$
```

2. To list all nodes in Yarn

```
yarn node -list
```

```
[bluedata@bluedata-81 ~]$ yarn node -list
WARNING: YARN_OPTS has been replaced by HADOOP_OPTS. Using value of YARN_OPTS.
19/09/30 08:31:40 INFO client.RMProxy: Connecting to ResourceManager at bluedata-81.dev.team.bdlocal/172.18.0.27:8032
Total Nodes:3
         Node-Id             Node-State Node-Http-Address       Number-of-Running-Containers
bluedata-83.dev.team.bdlocal:8041              RUNNING bluedata-83.dev.team.bdlocal:8042                                0
bluedata-84.dev.team.bdlocal:8041              RUNNING bluedata-84.dev.team.bdlocal:8042                                0
bluedata-85.dev.team.bdlocal:8041              RUNNING bluedata-85.dev.team.bdlocal:8042                                0
[bluedata@bluedata-81 ~]$
```

3. To view more details of each node, execute

```
yarn node -list -showDetails
```

```
[bluedata@bluedata-81 ~]$ yarn node -list -showDetails
WARNING: YARN_OPTS has been replaced by HADOOP_OPTS. Using value of YARN_OPTS.
19/09/30 08:34:50 INFO client.RMProxy: Connecting to ResourceManager at bluedata-81.dev.team.bdlocal/172.18.0.27:8032
19/09/30 08:34:50 INFO conf.Configuration: resource-types.xml not found
19/09/30 08:34:50 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
Total Nodes:3
         Node-Id             Node-State Node-Http-Address       Number-of-Running-Containers
bluedata-83.dev.team.bdlocal:8041              RUNNING bluedata-83.dev.team.bdlocal:8042                                0
Detailed Node Information :
        Configured Resources : <memory:6144, vCores:4>
        Allocated Resources : <memory:0, vCores:0>
        Resource Utilization by Node : PMem:48258 MB, VMem:48285 MB, VCores:0.40653116
        Resource Utilization by Containers : PMem:0 MB, VMem:0 MB, VCores:0.0
        Node-Labels :
bluedata-84.dev.team.bdlocal:8041              RUNNING bluedata-84.dev.team.bdlocal:8042                                0
Detailed Node Information :
        Configured Resources : <memory:6144, vCores:4>
        Allocated Resources : <memory:0, vCores:0>
        Resource Utilization by Node : PMem:32847 MB, VMem:32848 MB, VCores:0.35988003
        Resource Utilization by Containers : PMem:0 MB, VMem:0 MB, VCores:0.0
        Node-Labels :
bluedata-85.dev.team.bdlocal:8041              RUNNING bluedata-85.dev.team.bdlocal:8042                                0
Detailed Node Information :
        Configured Resources : <memory:6144, vCores:4>
        Allocated Resources : <memory:0, vCores:0>
        Resource Utilization by Node : PMem:48258 MB, VMem:48285 MB, VCores:0.40653116
        Resource Utilization by Containers : PMem:0 MB, VMem:0 MB, VCores:0.0
        Node-Labels :
[bluedata@bluedata-81 ~]$
```

4. To filter nodes on the basis on state, execute

```
yarn node -list -states RUNNING
```

```
[bluedata@bluedata-81 ~]$ yarn node -list -states RUNNING
WARNING: YARN_OPTS has been replaced by HADOOP_OPTS. Using value of YARN_OPTS.
19/09/30 08:37:48 INFO client.RMProxy: Connecting to ResourceManager at bluedata-81.dev.team.bdlocal/172.18.0.27:8032
Total Nodes:3
         Node-Id             Node-State Node-Http-Address       Number-of-Running-Containers
bluedata-83.dev.team.bdlocal:8041              RUNNING bluedata-83.dev.team.bdlocal:8042                             0
bluedata-84.dev.team.bdlocal:8041              RUNNING bluedata-84.dev.team.bdlocal:8042                             0
bluedata-85.dev.team.bdlocal:8041              RUNNING bluedata-85.dev.team.bdlocal:8042                             0
[bluedata@bluedata-81 ~]$
```

**Note:** States can be: NEW, RUNNING, UNHEALTHY, DECOMMISSIONED, LOST, REBOOTED, DECOMMISSIONING, SHUTDOWN.

5. To view the status information of any node, execute (Use Node ID, E.g. bluedata-83.dev.team.bdlocal:8041)

```
yarn node -status bluedata-83.dev.team.bdlocal:8041
```

```
[bluedata@bluedata-81 ~]$ yarn node -status bluedata-83.dev.team.bdlocal:8041
WARNING: YARN_OPTS has been replaced by HADOOP_OPTS. Using value of YARN_OPTS.
19/09/30 08:41:36 INFO client.RMProxy: Connecting to ResourceManager at bluedata-81.dev.team.bdlocal/172.18.0.27:8032
19/09/30 08:41:37 INFO conf.Configuration: resource-types.xml not found
19/09/30 08:41:37 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
Node Report :
        Node-Id : bluedata-83.dev.team.bdlocal:8041
        Rack : /default
        Node-State : RUNNING
        Node-Http-Address : bluedata-83.dev.team.bdlocal:8042
        Last-Health-Update : Mon 30/Sep/19 08:40:59:247PDT
        Health-Report :
        Containers : 0
        Memory-Used : 0MB
        Memory-Capacity : 6144MB
        CPU-Used : 0 vcores
        CPU-Capacity : 4 vcores
        Node-Labels :
        Resource Utilization by Node : PMem:48274 MB, VMem:48300 MB, VCores:0.6764412
        Resource Utilization by Containers : PMem:0 MB, VMem:0 MB, VCores:0.0

[bluedata@bluedata-81 ~]$
```

6. To view Hadoop Environment Variable details, execute

```
yarn envvars
```

```
[bluedata@bluedata-81 ~]$  yarn envvars
WARNING: YARN_OPTS has been replaced by HADOOP_OPTS. Using value of YARN_OPTS.
JAVA_HOME='/opt/jdk'
HADOOP_YARN_HOME='/opt/cloudera/parcels/CDH-6.1.0-1.cdh6.1.0.p0.770702/lib/hadoop/libexec/../../hadoop-yarn'
YARN_DIR='./'
YARN_LIB_JARS_DIR='lib'
HADOOP_CONF_DIR='/etc/hadoop/conf'
HADOOP_TOOLS_HOME='/opt/cloudera/parcels/CDH-6.1.0-1.cdh6.1.0.p0.770702/lib/hadoop'
HADOOP_TOOLS_DIR='share/hadoop/tools'
HADOOP_TOOLS_LIB_JARS_DIR='share/hadoop/tools/lib'
[bluedata@bluedata-81 ~]$
```

7. To get list of all application, execute

```
yarn application -list
```

```
[bluedata@bluedata-81 ~]$ yarn application -list
WARNING: YARN_OPTS has been replaced by HADOOP_OPTS. Using value of YARN_OPTS.
19/09/30 09:05:32 INFO client.RMProxy: Connecting to ResourceManager at bluedata-81.dev.team.bdlocal/172.18.0.27:8032
Total number of applications (application-types: [], states: [SUBMITTED, ACCEPTED, RUNNING] and tags: []):0
                Application-Id      Application-Name        Application-Type          User           Queue                   State
   Final-State         Progress                        Tracking-URL
[bluedata@bluedata-81 ~]$
```

**Note:** By default, no application running.

# 4 TESTING SPARK

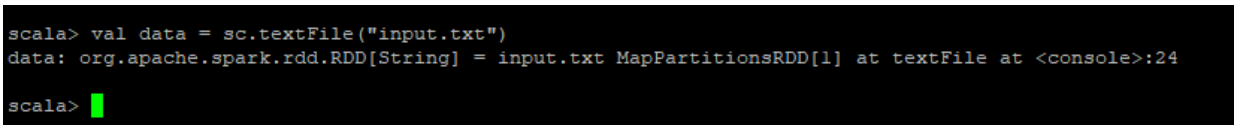
In this section, we will test Spark.

1. Enter the Spark shell

```
spark-shell
```

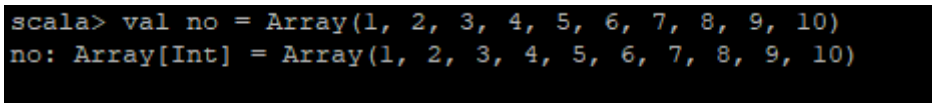

2. To create a new RDD, execute

```
val data = sc.textFile("input.txt")
```

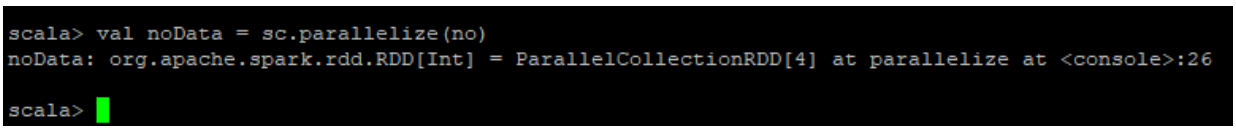



3. Create RDD using Parallelized Collection

```
val no = Array(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
```




```
val noData = sc.parallelize(no)
```




4. Creating a new RDD using existing RDD

```
val newRDD = no.map(data => (data * 2))
```

```
scala> val newRDD = no.map(data => (data * 2))
newRDD: Array[Int] = Array(2, 4, 6, 8, 10, 12, 14, 16, 18, 20)

scala>
```

5. To count items in RDD, follow:
   a. First create a RDD:

```
val num = Array(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
val NewData = sc.parallelize(num)
```

```
scala> val num = Array(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
num: Array[Int] = Array(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

scala> val NewData = sc.parallelize(num)
NewData: org.apache.spark.rdd.RDD[Int] = ParallelCollectionRDD[4] at parallelize at <console>:26
```

   b. Let's check the count now

```
NewData.count()
```

```
scala> NewData.count()
res0: Long = 10
```

   c. Read the first item from the RDD

```
NewData.first()
```

```
scala> NewData.first()
res4: Int = 1
```

   d. Read first 5 items from the RDD

```
NewData.take(5)
```

```
scala> NewData.take(5)
res5: Array[Int] = Array(1, 2, 3, 4, 5)
```

e. To count the number of partitions, execute

```
NewData.partitions.length
```

```
scala> NewData.partitions.length
res6: Int = 4
```

f. To cache the file

```
NewData.cache()
```

```
scala> NewData.cache()
res7: NewData.type = ParallelCollectionRDD[4] at parallelize at <console>:26
```

g. To collect, execute

```
NewData.collect()
```

```
scala> NewData.collect()
res8: Array[Int] = Array(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
```