

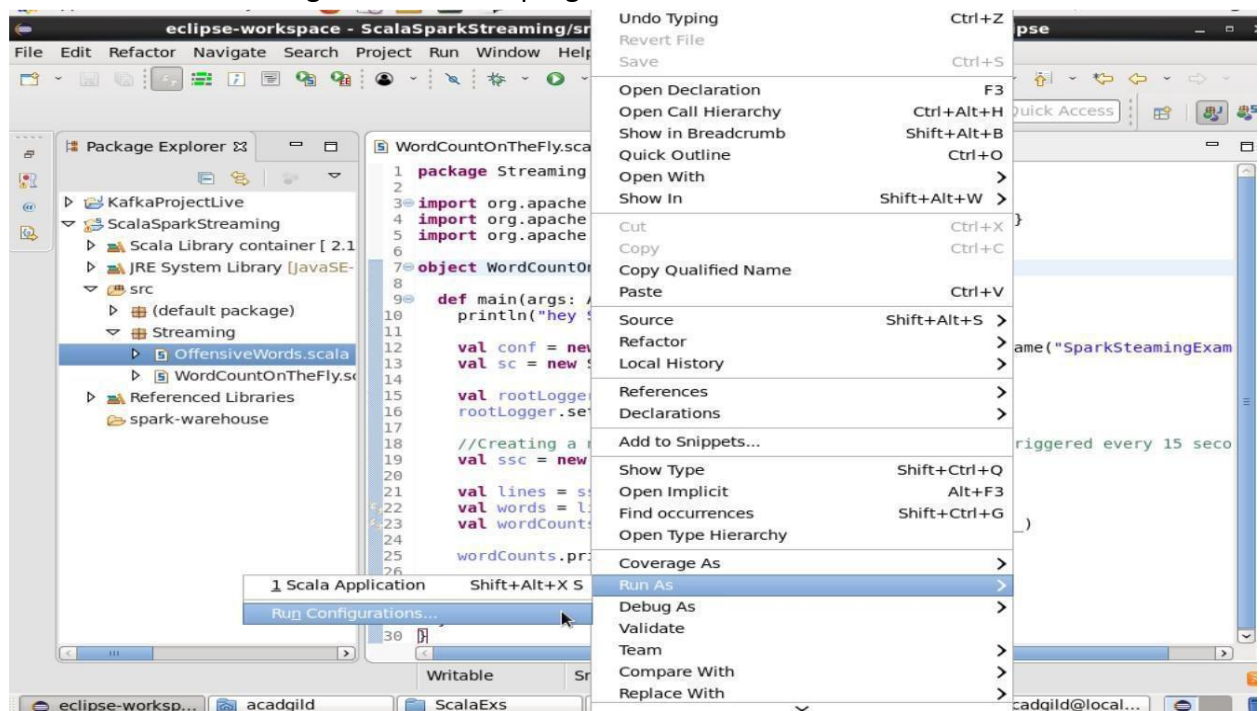
## Case Study- IV Spark Streaming

1. There are two parts in the case study

**First Part** - You have to create a Spark Application which streams data from a file on local directory on your machine and does the word count on the fly. The word should be done by the spark application in such a way that as soon as you drop the file in your local directory, your spark application should immediately do the word count for you.

**Output:**

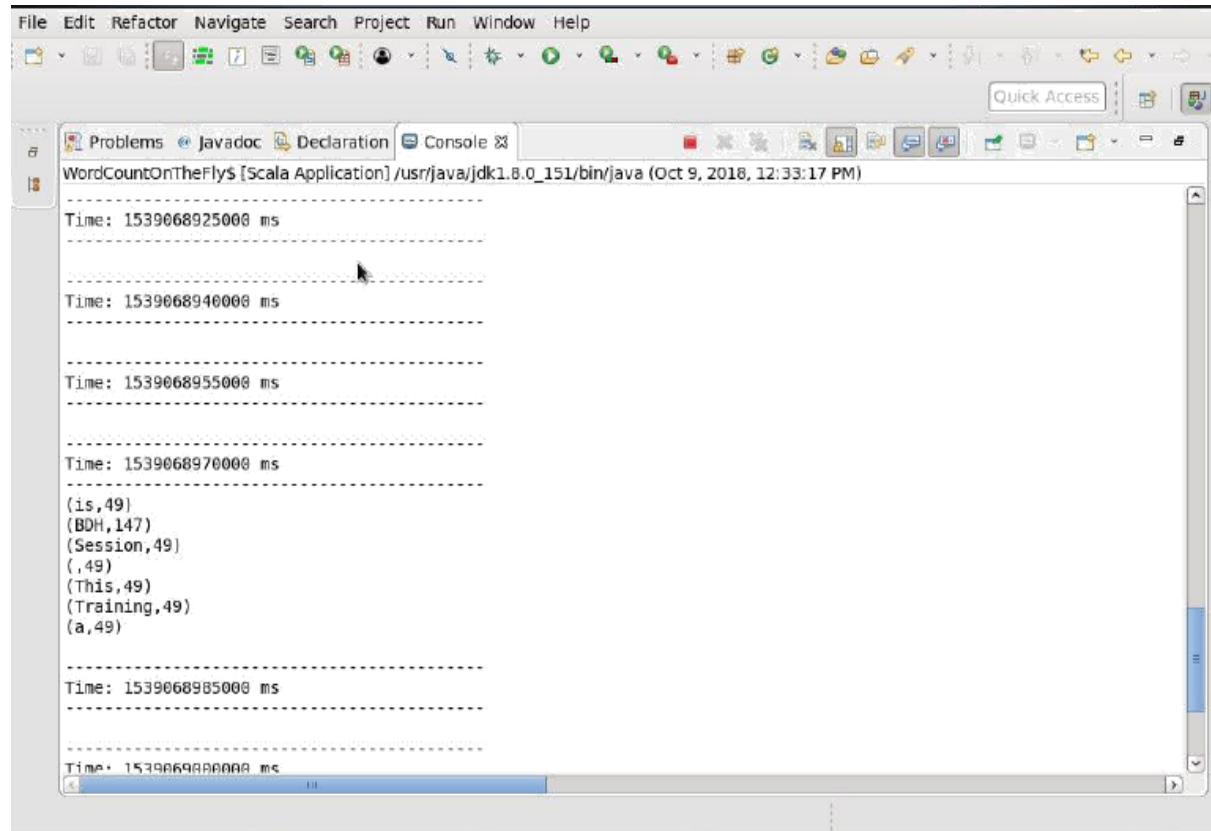
- Go to Run Configurations of the program



- On the “Arguments” tab Pass the arguments and click on “Run” as shown below



- The contents of newFile.txt are being read by Spark Streaming application and is word counting on the fly



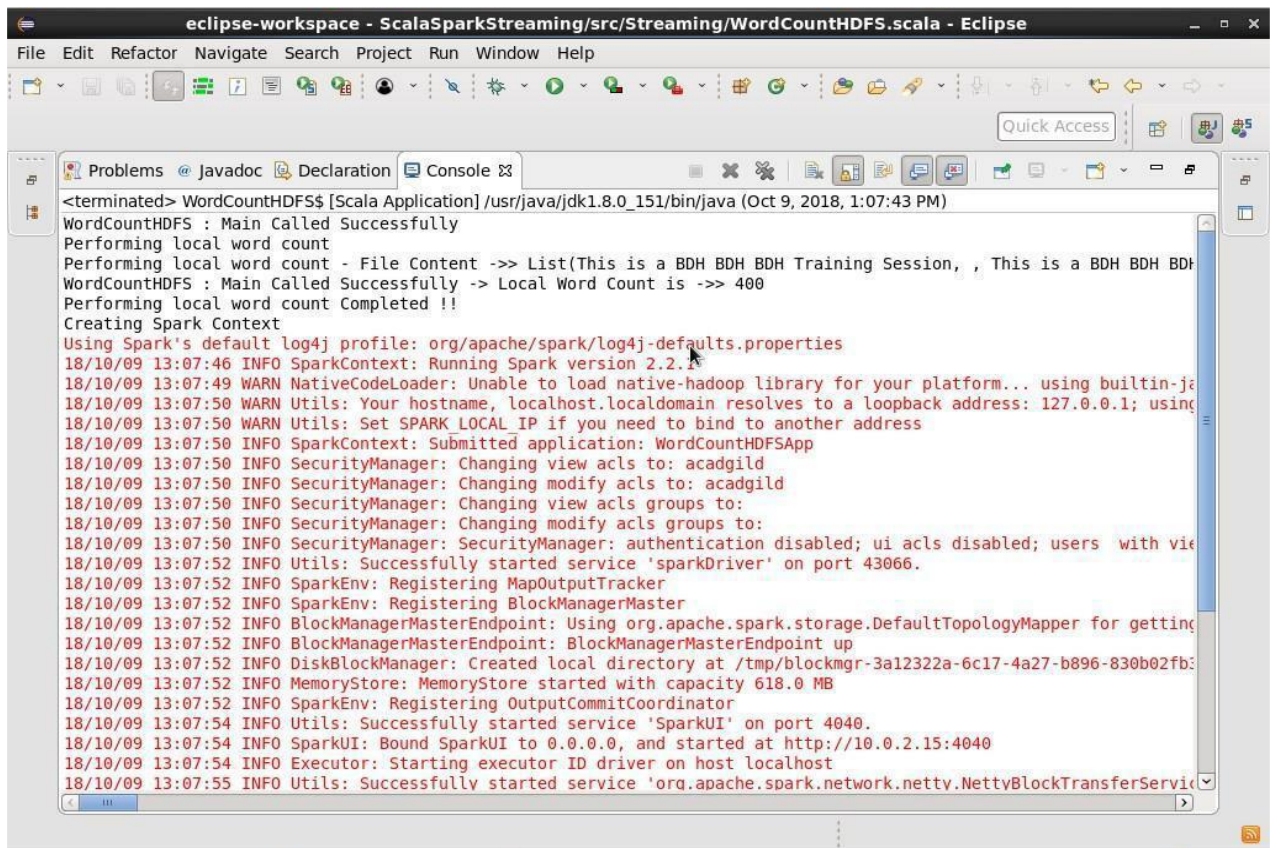
**Second Part** - In this part, you will have to create a Spark Application which should do the following

1. Pick up a file from the local directory and do the word count
2. Then in the same Spark Application, write the code to put the same file on HDFS.
3. Then in same Spark Application, do the word count of the file copied on HDFS in step 2
4. Lastly, compare the word count of step 1 and 2. Both should match, other throw an error

**Output:**

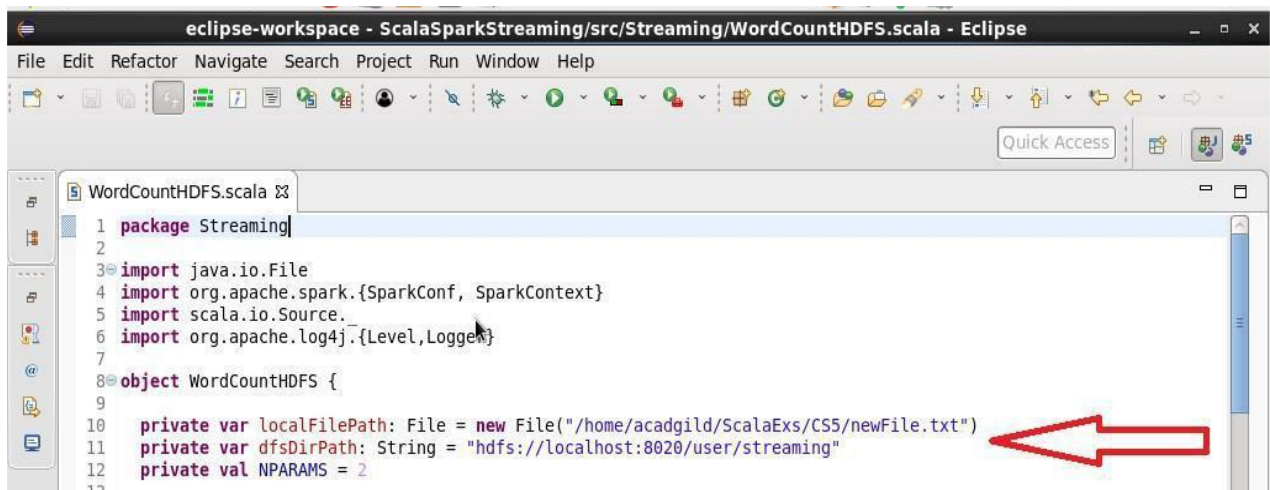
- HDFS does not contain and streaming directory before the application is run

**Step 1:** Use newFile.txt from the local directory and do the word count.



```
<terminated> WordCountHDFS$ [Scala Application] /usr/java/jdk1.8.0_151/bin/java (Oct 9, 2018, 1:07:43 PM)
WordCountHDFS : Main Called Successfully
Performing local word count
Performing local word count - File Content ->> List(This is a BDH BDH BDH Training Session, , This is a BDH BDH BDH
WordCountHDFS : Main Called Successfully -> Local Word Count is ->> 400
Performing local word count Completed !!
Creating Spark Context
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
18/10/09 13:07:46 INFO SparkContext: Running Spark version 2.2.1
18/10/09 13:07:49 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-j
18/10/09 13:07:50 WARN Utils: Your hostname, localhost.localdomain resolves to a loopback address: 127.0.0.1; using
18/10/09 13:07:50 WARN Utils: Set SPARK LOCAL IP if you need to bind to another address
18/10/09 13:07:50 INFO SparkContext: Submitted application: WordCountHDFSApp
18/10/09 13:07:50 INFO SecurityManager: Changing view acls to: acadgild
18/10/09 13:07:50 INFO SecurityManager: Changing modify acls to: acadgild
18/10/09 13:07:50 INFO SecurityManager: Changing view acls groups to:
18/10/09 13:07:50 INFO SecurityManager: Changing modify acls groups to:
18/10/09 13:07:52 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with vie
18/10/09 13:07:52 INFO Utils: Successfully started service 'sparkDriver' on port 43066.
18/10/09 13:07:52 INFO SparkEnv: Registering MapOutputTracker
18/10/09 13:07:52 INFO SparkEnv: Registering BlockManagerMaster
18/10/09 13:07:52 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting
18/10/09 13:07:52 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
18/10/09 13:07:52 INFO DiskBlockManager: Created local directory at /tmp/blockmgr-3a12322a-6c17-4a27-b896-830b02fb:
18/10/09 13:07:52 INFO MemoryStore: MemoryStore started with capacity 618.0 MB
18/10/09 13:07:52 INFO SparkEnv: Registering OutputCommitCoordinator
18/10/09 13:07:54 INFO Utils: Successfully started service 'SparkUI' on port 4040.
18/10/09 13:07:54 INFO SparkUI: Bound SparkUI to 0.0.0.0, and started at http://10.0.2.15:4040
18/10/09 13:07:54 INFO Executor: Starting executor ID driver on host localhost
18/10/09 13:07:55 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferServi
```

**Step 2:** Then in the same spark Application, write the code to put the same file on HDFS

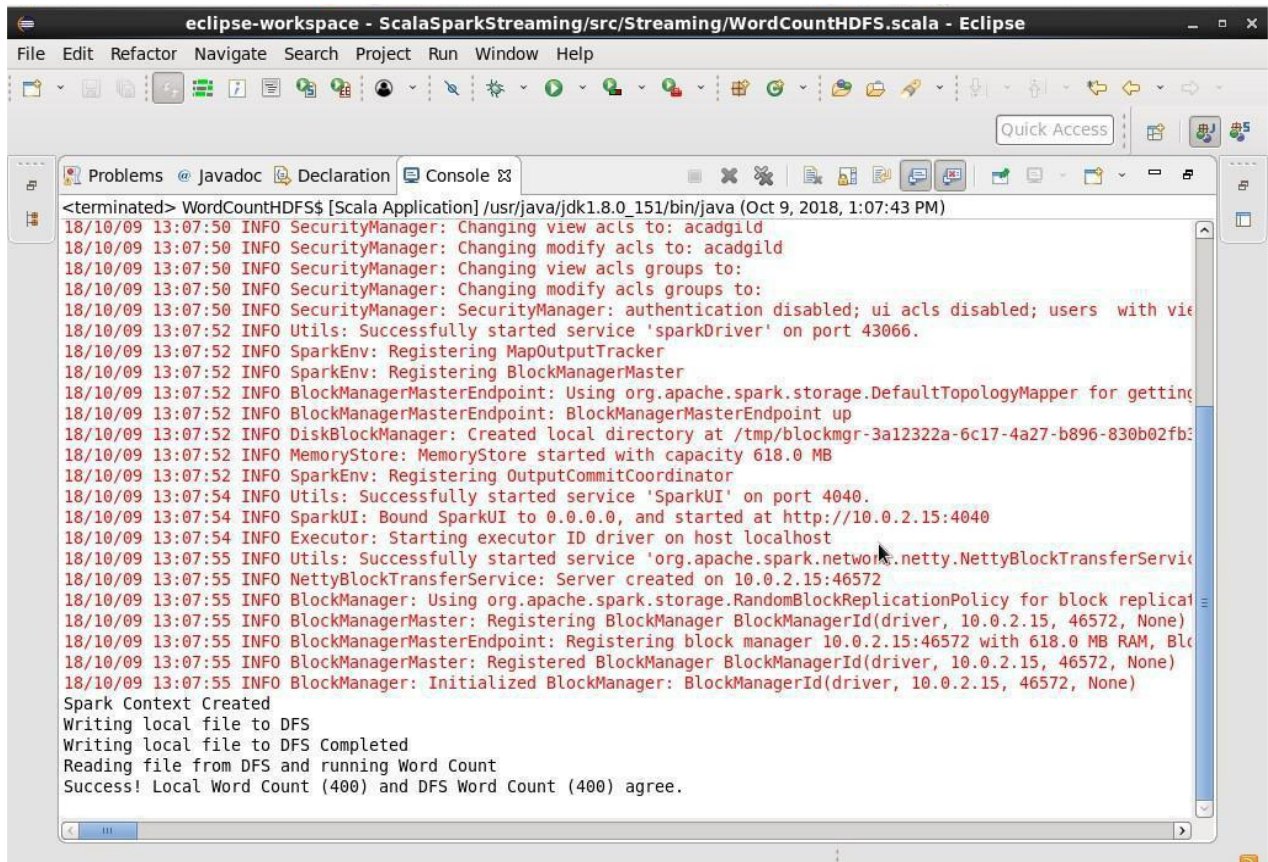


```
1 package Streaming
2
3 import java.io.File
4 import org.apache.spark.{SparkConf, SparkContext}
5 import scala.io.Source
6 import org.apache.log4j.{Level, Logger}
7
8 object WordCountHDFS {
9
10   private var localFilePath: File = new File("/home/acadgild/ScalaExs/CS5/newFile.txt")
11   private var dfsDirPath: String = "hdfs://localhost:8020/user/streaming"
12   private val NPARAMS = 2
13 }
```

Then in same Spark Application, do the word count of the file copied on to HDFS in step 2



Lastly, compare the word count of step 1 and step 2. Both should match, otherwise throw an error



The screenshot shows the Eclipse IDE interface with the console window open. The title bar indicates the workspace is 'eclipse-workspace - ScalaSparkStreaming/src/Streaming/WordCountHDFS.scala - Eclipse'. The console displays the output of a Scala application, showing various INFO logs from the Spark ecosystem, including SecurityManager, SparkEnv, BlockManagerMasterEndpoint, DiskBlockManager, MemoryStore, SparkUI, and BlockManager. The final output indicates the Spark Context was created, files were written to and read from DFS, and the word count was successfully calculated as 400, matching the local word count.

```
<terminated> WordCountHDFS$ [Scala Application] /usr/java/jdk1.8.0_151/bin/java (Oct 9, 2018, 1:07:43 PM)
18/10/09 13:07:50 INFO SecurityManager: Changing view acls to: acadgild
18/10/09 13:07:50 INFO SecurityManager: Changing modify acls to: acadgild
18/10/09 13:07:50 INFO SecurityManager: Changing view acls groups to:
18/10/09 13:07:50 INFO SecurityManager: Changing modify acls groups to:
18/10/09 13:07:50 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with vie
18/10/09 13:07:52 INFO Utils: Successfully started service 'sparkDriver' on port 43066.
18/10/09 13:07:52 INFO SparkEnv: Registering MapOutputTracker
18/10/09 13:07:52 INFO SparkEnv: Registering BlockManagerMaster
18/10/09 13:07:52 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting
18/10/09 13:07:52 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
18/10/09 13:07:52 INFO DiskBlockManager: Created local directory at /tmp/blockmgr-3a12322a-6c17-4a27-b896-830b02fb:
18/10/09 13:07:52 INFO MemoryStore: MemoryStore started with capacity 618.0 MB
18/10/09 13:07:52 INFO SparkEnv: Registering OutputCommitCoordinator
18/10/09 13:07:54 INFO Utils: Successfully started service 'SparkUI' on port 4040.
18/10/09 13:07:54 INFO SparkUI: Bound SparkUI to 0.0.0.0, and started at http://10.0.2.15:4040
18/10/09 13:07:54 INFO Executor: Starting executor ID driver on host localhost
18/10/09 13:07:55 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferServic
18/10/09 13:07:55 INFO NettyBlockTransferService: Server created on 10.0.2.15:46572
18/10/09 13:07:55 INFO BlockManager: Using org.apache.spark.storage.RandomBlockReplicationPolicy for block replicat
18/10/09 13:07:55 INFO BlockManagerMaster: Registering BlockManager BlockManagerId(driver, 10.0.2.15, 46572, None)
18/10/09 13:07:55 INFO BlockManagerMasterEndpoint: Registering block manager 10.0.2.15:46572 with 618.0 MB RAM, Blc
18/10/09 13:07:55 INFO BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 10.0.2.15, 46572, None)
18/10/09 13:07:55 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 10.0.2.15, 46572, None)
Spark Context Created
Writing local file to DFS
Writing local file to DFS Completed
Reading file from DFS and running Word Count
Success! Local Word Count (400) and DFS Word Count (400) agree.
```

Here we see that a directory “Streaming” was created in HDFS, which contains another directory “dsf\_read\_write\_test” which contains 2 files as a result of the job performed by Spark Streaming program

```
Applications Places System Tue Oct 9, 2:11 PM Acadgild
acadgild@localhost:~/ScalaExs/CS5
File Edit View Search Terminal Help
[acadgild@localhost CS5]$ hadoop fs -ls /user
18/10/09 13:07:19 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cl
asses where applicable
Found 1 items
drwxr-xr-x - acadgild supergroup 0 2018-02-09 14:50 /user/hive
[acadgild@localhost CS5]$ hadoop fs -ls /user
18/10/09 14:09:50 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cl
asses where applicable
Found 2 items
drwxr-xr-x - acadgild supergroup 0 2018-02-09 14:50 /user/hive
drwxr-xr-x - acadgild supergroup 0 2018-10-09 13:08 /user/streaming
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost CS5]$ hadoop fs -ls /user/streaming
18/10/09 14:10:14 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cl
asses where applicable
Found 1 items
drwxr-xr-x - acadgild supergroup 0 2018-10-09 13:08 /user/streaming/dfs_read_write_test
[acadgild@localhost CS5]$ hadoop fs -ls /user/streaming/dfs_read_write_test
18/10/09 14:10:52 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cl
asses where applicable
Found 3 items
-rw-r--r-- 3 acadgild supergroup 0 2018-10-09 13:08 /user/streaming/dfs_read_write_test/ SUCCESS
-rw-r--r-- 3 acadgild supergroup 999 2018-10-09 13:08 /user/streaming/dfs_read_write_test/part-00000
-rw-r--r-- 3 acadgild supergroup 1000 2018-10-09 13:08 /user/streaming/dfs_read_write_test/part-00001
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost CS5]$
```