Data Engineering

Yarn 2

BY MARTIN LE FORMAL

DE1 PROMO 2023

15/07/2023

Installation & Uploading

Being a vim developer I decided to use Maven from the command line and not use IntelliJ so first I compile the Java code using Maven to generate Jar binary files:

```
❯ mvn package
[INFO] Scanning for projects...
[INFO]
[INFO] ----------------< com.opstty:hadoop-examples-mapreduce >----------------
[INFO] Building hadoop-examples-mapreduce 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
```

```
[INFO]
[INFO] --- assembly:3.6.0:single (default) @ hadoop-examples-mapreduce ---
[INFO] Building jar: /Users/martin.le.formalefrei.net/project/data-engineering/y
s-mapreduce-1.0-SNAPSHOT-jar-with-dependencies.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  7.711 s
[INFO] Finished at: 2023-07-25T16:35:07+02:00
[INFO] ------------------------------------------------------------------------

~/project/data-engineering/yarn_2/hadoop-examples-mapreduce 〉main
❯ ls
README.md                   pom.xml                     target
hadoop-examples-mapreduce.iml src

~/project/data-engineering/yarn_2/hadoop-examples-mapreduce 〉main
❯ find . -name '*.jar'
./target/hadoop-examples-mapreduce-1.0-SNAPSHOT-jar-with-dependencies.jar
./target/hadoop-examples-mapreduce-1.0-SNAPSHOT.jar
```

Now I upload the compiled files files to the edge node using scp:

```
❯ scp ./target/hadoop-examples-mapreduce-1.0-SNAPSHOT-jar-with-depen
ome/martin.le.formal/
(martin.le.formal@master01.hadoop.efrei.clemlab.com) Password:
(martin.le.formal@master01.hadoop.efrei.clemlab.com) Password:
hadoop-examples-mapreduce-1.0-SNAPSHOT-jar-with-dependencies.jar
```

```
-bash-4.2$ ls
create_internal_tree_table.sql  ebook.txt
create_tree_table.sql           hadoop-examples-mapreduce-1.0-SNAPSHOT-jar-with-dependencies.jar
demat.txt                       local.txt
```

The jar file has been uploaded successfully.

Now let's run the job word count, I renamed the jar file to lib for easier manipulation and I uploaded it as well as the ebook.txt on pdfs:

```
-bash-4.2$ yarn jar lib.jar wordcount ebook.txt wordcount
```

```
-bash-4.2$ hdfs dfs -get wordcount
-bash-4.2$ ls
create_internal_tree_table.sql  demat.txt  lib.jar    mapper.py  sudoku.dta      wordcount
create_tree_table.sql           ebook.txt  local.txt  queries    trees_external
-bash-4.2$ vi wordcount/
part-r-00000  _SUCCESS
-bash-4.2$ vi wordcount/_SUCCESS
-bash-4.2$ vi wordcount/
part-r-00000  _SUCCESS
-bash-4.2$ vi wordcount/part-r-00000
```

```
"A      4
"But    1
"Cannibal!"     1
"Dear   1
"Go,    1
"He     1
"Here,  1
"I      3
"I'm    1
"It     1
"Let's  1
"Masai! 1
"No     1
"No!"   2
"No,"   1
"Not    1
"Oh,    2
"One    1
"Ride   1
"Senecoza       2
"Steve!"        1
```

For the rest of the questions you can see it on the following GitHub link:

https://github.com/ChessMartin/yarn2