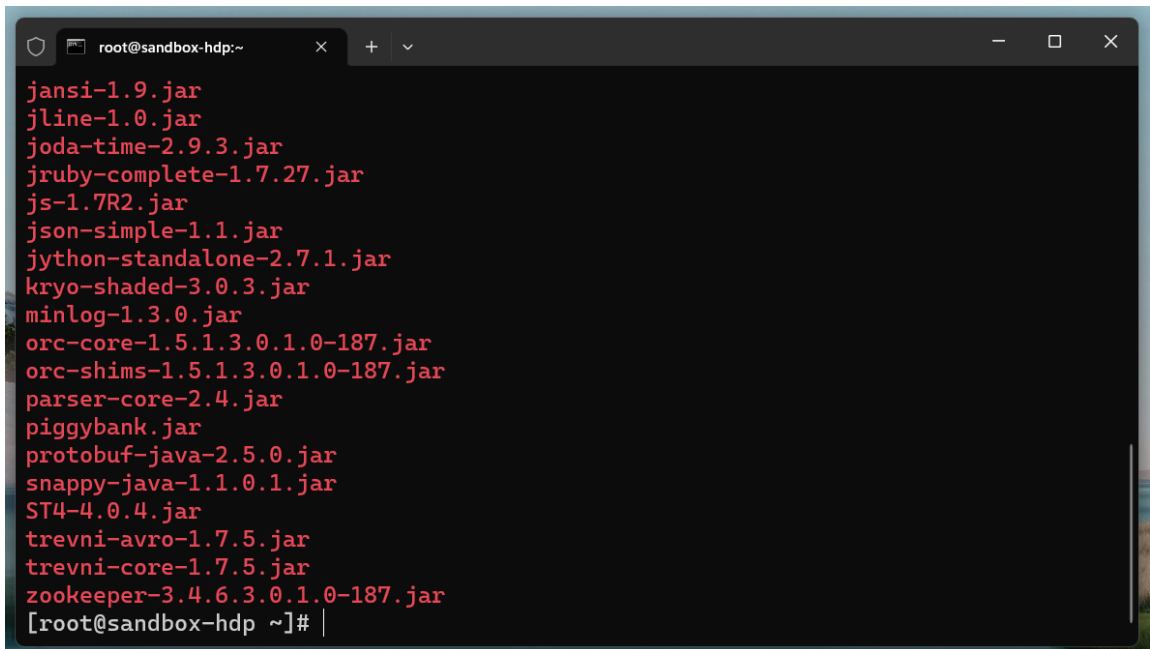


Praktikum Teknologi Perekayasaan Data

Tanggal Praktikum : Monday, June 12, 2023

1. Pastikan terdapat file piggybank.jar di /usr/hdp/3.0.1.0-187/pig/lib



```
root@sandbox-hdp:~  
jansi-1.9.jar  
jline-1.0.jar  
joda-time-2.9.3.jar  
jruby-complete-1.7.27.jar  
js-1.7R2.jar  
json-simple-1.1.jar  
jython-standalone-2.7.1.jar  
kryo-shaded-3.0.3.jar  
minlog-1.3.0.jar  
orc-core-1.5.1.3.0.1.0-187.jar  
orc-shims-1.5.1.3.0.1.0-187.jar  
parser-core-2.4.jar  
piggybank.jar  
protobuf-java-2.5.0.jar  
snappy-java-1.1.0.1.jar  
ST4-4.0.4.jar  
trevni-avro-1.7.5.jar  
trevni-core-1.7.5.jar  
zookeeper-3.4.6.3.0.1.0-187.jar  
[root@sandbox-hdp ~]#
```

Pemrosesan Data Semi-Structured

1. Upload meter_reading.xml

```
root@sandbox-hdp:~  
log_data.txt      pig_1685285337287.log  sampeldata.csv      tryhive  
movies_data.csv   pig_1685285657177.log  script_example.pig  
my_movies        pig_1685285734333.log  script.pig  
[root@sandbox-hdp ~]# nano asd.txt  
-bash: nano: command not found  
[root@sandbox-hdp ~]# vi  
[root@sandbox-hdp ~]# echo "<?xml version =\"1.0\"?>  
>   <feed>  
     rel=\"self\"></link>>   <link href = \"/v1/espi_third_party_batch_feed\" rel=\"self  
\"></link>  
>   <title type =\"text\">Zip Code 93308-6173</title>  
>   <IntervalReading>  
>     <cost>1510</cost>  
>     <timePeriod>  
>       <duration>900</duration>  
>       <start>1301889600</start>  
>     </timePeriod>  
>     <value>102</value>  
>   </IntervalReading>  
>   <IntervalReading>
```

2. Load Data

```
grunt> data = LOAD 'meter_reading.xml' USING org.apache.pig.piggybank.storage  
.XMLLoader('IntervalReading') as (x:chararray)  
>> ;  
grunt> DEFINE XPath org.apache.pig.piggybank.evaluation.xml.XPath();  
grunt> data_extract = FOREACH data GENERATE  
grunt> (XPath(x, 'IntervalReading/timePeriod/duration')),  
>> (XPath(x, 'IntervalReading/timePeriod/start')),  
>> (XPath(x, 'IntervalReading/value')), (XPath(x,  
>> 'IntervalReading/cost'));  
grunt> dump data_extract  
2023-06-12 00:57:01,242 [main] INFO org.apache.pig.tools.pigstats.ScriptState -  
Pig features used in the script: UNKNOWN  
2023-06-12 00:57:01,348 [main] INFO org.apache.pig.newplan.logical.optimizer.Lo  
gicalPlanOptimizer - {RULES_ENABLED=[AddForEach, ColumnMapKeyPrune, ConstantCalc  
ulator, GroupByConstParallelSetter, LimitOptimizer, LoadTypeCastInserter, MergeF  
ilter, MergeForEach, PartitionFilterOptimizer, PredicatePushdownOptimizer, PushD  
ownForEachFlatten, PushUpFilter, SplitFilter, StreamTypeCastInserter]}  
2023-06-12 00:57:01,457 [main] INFO org.apache.pig.impl.util.SpillableMemoryMan  
ager - Selected heap (PS Old Gen) of size 699400192 to monitor. collectionUsageT  
hreshold = 489580128, usageThreshold = 489580128
```

3. Save Data

```
root@sandbox-hdp:~  
2023-06-12 00:57:14,731 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!  
2023-06-12 00:57:14,740 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!  
2023-06-12 00:57:14,745 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!  
2023-06-12 00:57:14,804 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized  
2023-06-12 00:57:14,967 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input files to process : 1  
2023-06-12 00:57:14,968 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1  
(900,1301889600,102,1510)  
(900,1301890500,102,1510)  
grunt> store data_extract into 'user/root/pig' using PigStorage(',');  
<line 9, column 24> Unexpected character ' '  
2023-06-12 00:59:03,709 [main] ERROR org.apache.pig.tools.grunt.Grunt - ERROR 1200: <line 9, column 24> Unexpected character ' '  
Details at logfile: /root/pig_1686531391722.log  
grunt> |
```

```
root@sandbox-hdp:~  
Spillable Memory Manager spill count : 0  
Total bags proactively spilled: 0  
Total records proactively spilled: 0  
  
Job DAG:  
job_local578076867_0002  
  
2023-06-12 01:01:19,237 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!  
2023-06-12 01:01:19,238 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!  
2023-06-12 01:01:19,239 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!  
2023-06-12 01:01:19,243 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!  
grunt> cat user/root/pig/part-m-000000  
900,1301889600,102,1510  
900,1301890500,102,1510  
grunt> |
```

Pemrosesan Data Unstructured


1. Upload file bigdata.jpg

```

root@sandbox-hdp:~
emImpl - JobTracker metrics system already initialized!
2023-06-12 01:01:19,243 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.mapReduceLayer.MapReduceLauncher - Success!
grunt> cat user/root/pig/part-m-00000
900,1301889600,102,1510
900,1301890500,102,1510
grunt> [root@sandbox-hdp ~]# hdfs dfs -ls user/root
ls: `user/root': No such file or directory
[root@sandbox-hdp ~]# hdfs dfs -ls /user/root
Found 7 items
drwxr-xr-x   - root  hdfs          0 2023-06-12 00:48 /user/root/.sparkStaging
-rw-r--r--   1 admin hdfs    391254 2023-06-12 01:03 /user/root/bigdata.jpg
-rw-r--r--   1 admin hdfs     202 2023-06-04 15:50 /user/root/emp.csv
-rw-r--r--   1 admin hdfs     477 2023-06-12 00:43 /user/root/meter_reading.x
ml
drwxr-xr-x   - root  hdfs          0 2023-06-04 16:19 /user/root/salary_more_tha
n_5000
drwxr-xr-x   - root  hdfs          0 2023-06-05 04:44 /user/root/tpd12
drwxr-xr-x   - root  hdfs          0 2023-05-15 17:34 /user/root/tryhive
[root@sandbox-hdp ~]#

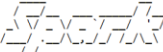
```

2. Load Data



```

^C^C[root@sandbox-hdp ~]# spark-shell
SPARK_MAJOR_VERSION is set to 2, using Spark2
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
[root@sandbox-hdp ~]# spark-shell
SPARK_MAJOR_VERSION is set to 2, using Spark2
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Spark context Web UI available at http://sandbox-hdp.hortonworks.com:4040
Spark context available as 'sc' (master = yarn, app id = application_1686535763309_0001).
Spark session available as 'spark'.
Welcome to

 version 2.3.1.3.0.1.0-187

Using Scala version 2.11.8 (OpenJDK 64-Bit Server VM, Java 1.8.0_191)
Type in expressions to have them evaluated.
Type :help for more information.

scala> import org.apache.spark.ml.image.ImageSchema_
<console>:23: error: object ImageSchema_ is not a member of package org.apache.spark.ml.image
import org.apache.spark.ml.image.ImageSchema_
      ^

scala> import org.apache.spark.ml.image.ImageSchema
<console>:23: error: object ImageSchema is not a member of package org.apache.spark.ml.image
import org.apache.spark.ml.image.ImageSchema
      ^

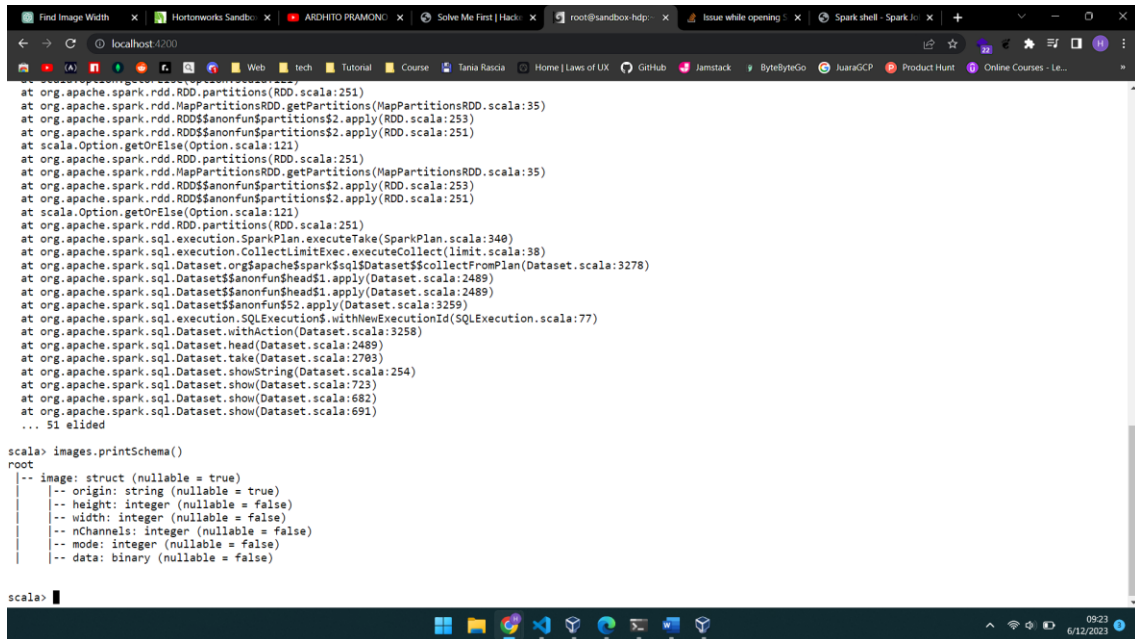
scala> import org.apache.spark.ml.image.ImageSchema._
import org.apache.spark.ml.image.ImageSchema._

scala> import java.nio.file.Paths
import java.nio.file.Paths

scala>

```

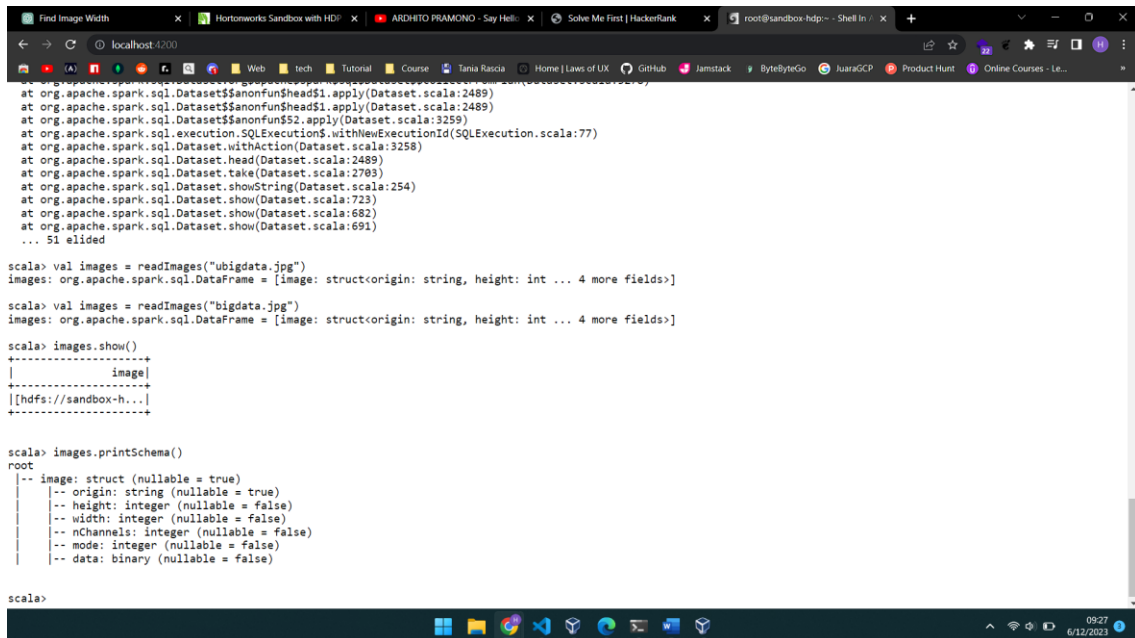
3. Cek File



```
at org.apache.spark.rdd.RDD.partitions(RDD.scala:251)
at org.apache.spark.rdd.MapPartitionsRDD.getPartitions(MapPartitionsRDD.scala:35)
at org.apache.spark.rdd.RDD$$anonfun$partitions$2.apply(RDD.scala:253)
at org.apache.spark.rdd.RDD$$anonfun$partitions$2.apply(RDD.scala:251)
at scala.Option.getOrElse(Option.scala:121)
at org.apache.spark.rdd.RDD.partitions(RDD.scala:251)
at org.apache.spark.rdd.MapPartitionsRDD.getPartitions(MapPartitionsRDD.scala:35)
at org.apache.spark.rdd.RDD$$anonfun$partitions$2.apply(RDD.scala:253)
at org.apache.spark.rdd.RDD$$anonfun$partitions$2.apply(RDD.scala:251)
at scala.Option.getOrElse(Option.scala:121)
at org.apache.spark.rdd.RDD.partitions(RDD.scala:251)
at org.apache.spark.sql.execution.SparkPlan.executeTake(SparkPlan.scala:340)
at org.apache.spark.sql.execution.CollectLimitExec.executeCollect(limit.scala:38)
at org.apache.spark.sql.Dataset.org$apache$spark$sql$Dataset$collectFromPlan(Dataset.scala:3278)
at org.apache.spark.sql.Dataset$$anonfun$head$1.apply(Dataset.scala:2489)
at org.apache.spark.sql.Dataset$$anonfun$head$1.apply(Dataset.scala:2489)
at org.apache.spark.sql.Dataset$$anonfun$head$1.apply(Dataset.scala:2489)
at org.apache.spark.sql.execution.SQLExecution$.withNewExecutionId(SQLExecution.scala:77)
at org.apache.spark.sql.Dataset.withAction(Dataset.scala:3258)
at org.apache.spark.sql.Dataset.head(Dataset.scala:2489)
at org.apache.spark.sql.Dataset.take(Dataset.scala:2703)
at org.apache.spark.sql.Dataset.showString(Dataset.scala:254)
at org.apache.spark.sql.Dataset.show(Dataset.scala:723)
at org.apache.spark.sql.Dataset.show(Dataset.scala:682)
at org.apache.spark.sql.Dataset.show(Dataset.scala:691)
... 51 elided

scala> images.printSchema()
root
|-- image: struct (nullable = true)
|   |-- origin: string (nullable = true)
|   |-- height: integer (nullable = false)
|   |-- width: integer (nullable = false)
|   |-- nchannels: integer (nullable = false)
|   |-- mode: integer (nullable = false)
|   |-- data: binary (nullable = false)
|
```

4. Cek Data



```
at org.apache.spark.sql.Dataset$$anonfun$head$1.apply(Dataset.scala:2489)
at org.apache.spark.sql.Dataset$$anonfun$head$1.apply(Dataset.scala:2489)
at org.apache.spark.sql.Dataset$$anonfun$head$1.apply(Dataset.scala:2489)
at org.apache.spark.sql.execution.SQLExecution$.withNewExecutionId(SQLExecution.scala:77)
at org.apache.spark.sql.Dataset.withAction(Dataset.scala:3258)
at org.apache.spark.sql.Dataset.head(Dataset.scala:2489)
at org.apache.spark.sql.Dataset.take(Dataset.scala:2703)
at org.apache.spark.sql.Dataset.showString(Dataset.scala:254)
at org.apache.spark.sql.Dataset.show(Dataset.scala:723)
at org.apache.spark.sql.Dataset.show(Dataset.scala:682)
at org.apache.spark.sql.Dataset.show(Dataset.scala:691)
... 51 elided

scala> val images = readImages("ubidata.jpg")
images: org.apache.spark.sql.DataFrame = [image: struct<origin: string, height: int ... 4 more fields>]

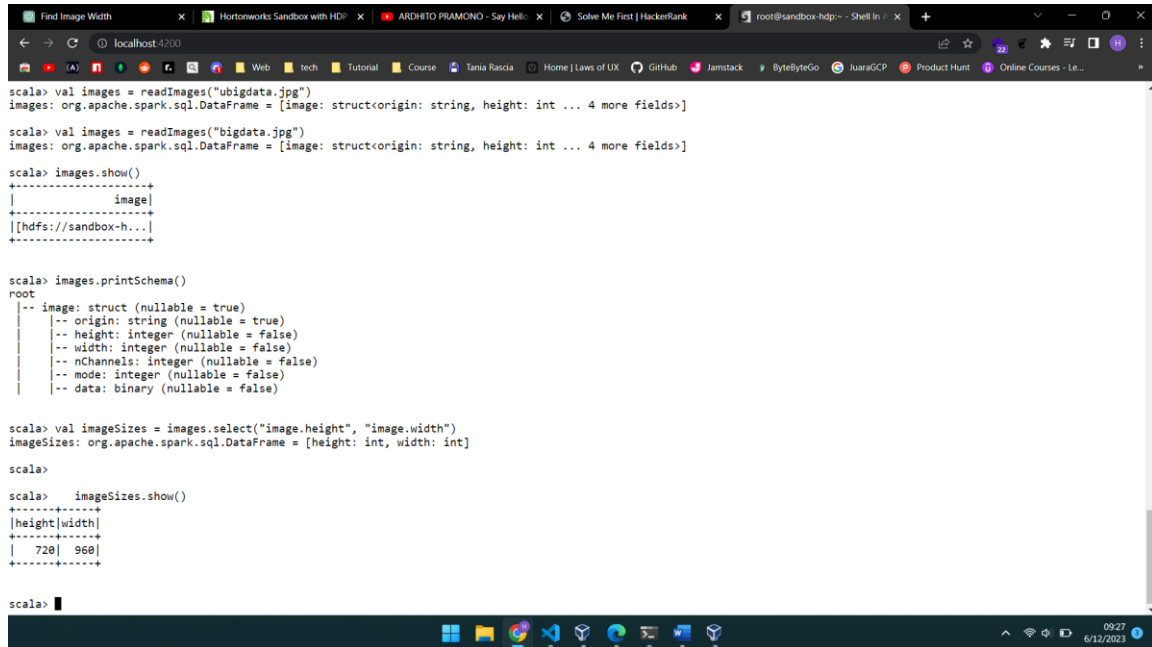
scala> val images = readImages("bigdata.jpg")
images: org.apache.spark.sql.DataFrame = [image: struct<origin: string, height: int ... 4 more fields>]

scala> images.show()
+-----+
|          image|
+-----+
|[hdfs://sandbox-h...|
+-----+

scala> images.printSchema()
root
|-- image: struct (nullable = true)
|   |-- origin: string (nullable = true)
|   |-- height: integer (nullable = false)
|   |-- width: integer (nullable = false)
|   |-- nchannels: integer (nullable = false)
|   |-- mode: integer (nullable = false)
|   |-- data: binary (nullable = false)
|
```

Penugasan

1. Menghitung tinggi dan lebar gambar



```
scala> val images = readImages("ubigdata.jpg")
images: org.apache.spark.sql.DataFrame = [image: struct<origin: string, height: int ... 4 more fields>]

scala> val images = readImages("bigdata.jpg")
images: org.apache.spark.sql.DataFrame = [image: struct<origin: string, height: int ... 4 more fields>]

scala> images.show()
+-----+
|          image          |
+-----+
|[hdfs://sandbox-h...]|
+-----+

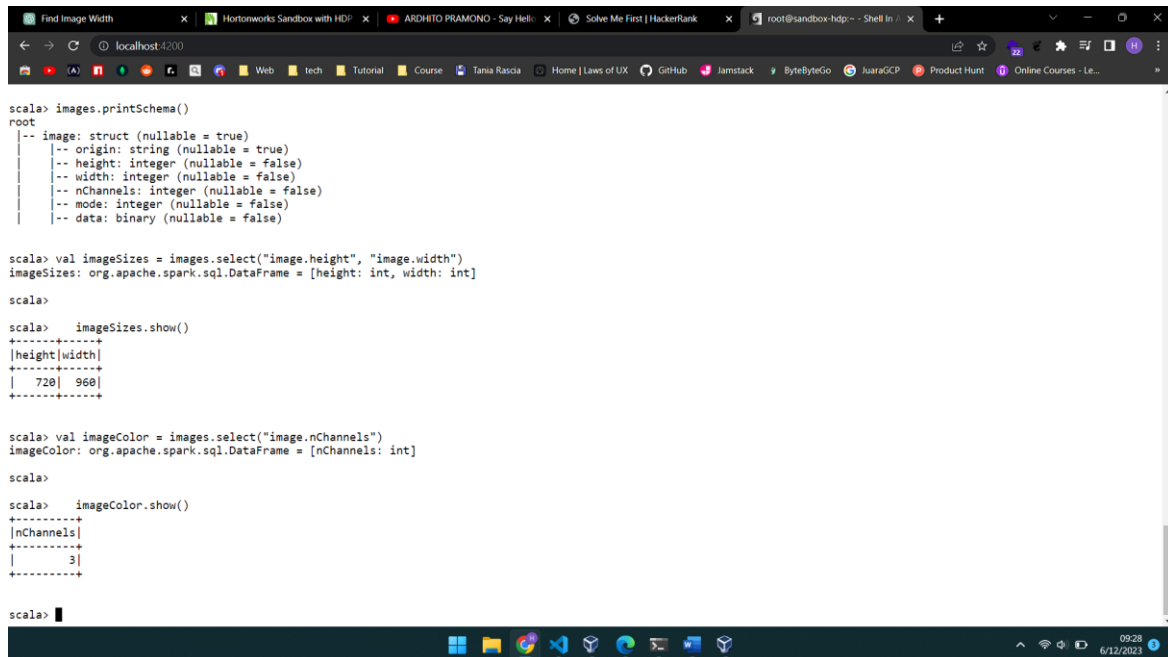
scala> images.printSchema()
root
|-- image: struct (nullable = true)
|   |-- origin: string (nullable = true)
|   |-- height: integer (nullable = false)
|   |-- width: integer (nullable = false)
|   |-- nChannels: integer (nullable = false)
|   |-- mode: integer (nullable = false)
|   |-- data: binary (nullable = false)

scala> val imageSizes = images.select("image.height", "image.width")
imageSizes: org.apache.spark.sql.DataFrame = [height: int, width: int]

scala>
scala> imageSizes.show()
+-----+
|height|width|
+-----+
|   720|  960|
+-----+

scala>
```

2. Melakukan deteksi apakah gambar tersebut berwarna atau tidak



```
scala> images.printSchema()
root
|-- image: struct (nullable = true)
|   |-- origin: string (nullable = true)
|   |-- height: integer (nullable = false)
|   |-- width: integer (nullable = false)
|   |-- nChannels: integer (nullable = false)
|   |-- mode: integer (nullable = false)
|   |-- data: binary (nullable = false)

scala> val imageSizes = images.select("image.height", "image.width")
imageSizes: org.apache.spark.sql.DataFrame = [height: int, width: int]

scala>
scala> imageSizes.show()
+-----+
|height|width|
+-----+
|   720|  960|
+-----+

scala> val imageColor = images.select("image.nChannels")
imageColor: org.apache.spark.sql.DataFrame = [nChannels: int]

scala>
scala> imageColor.show()
+-----+
|nChannels|
+-----+
|         3|
+-----+

scala>
```