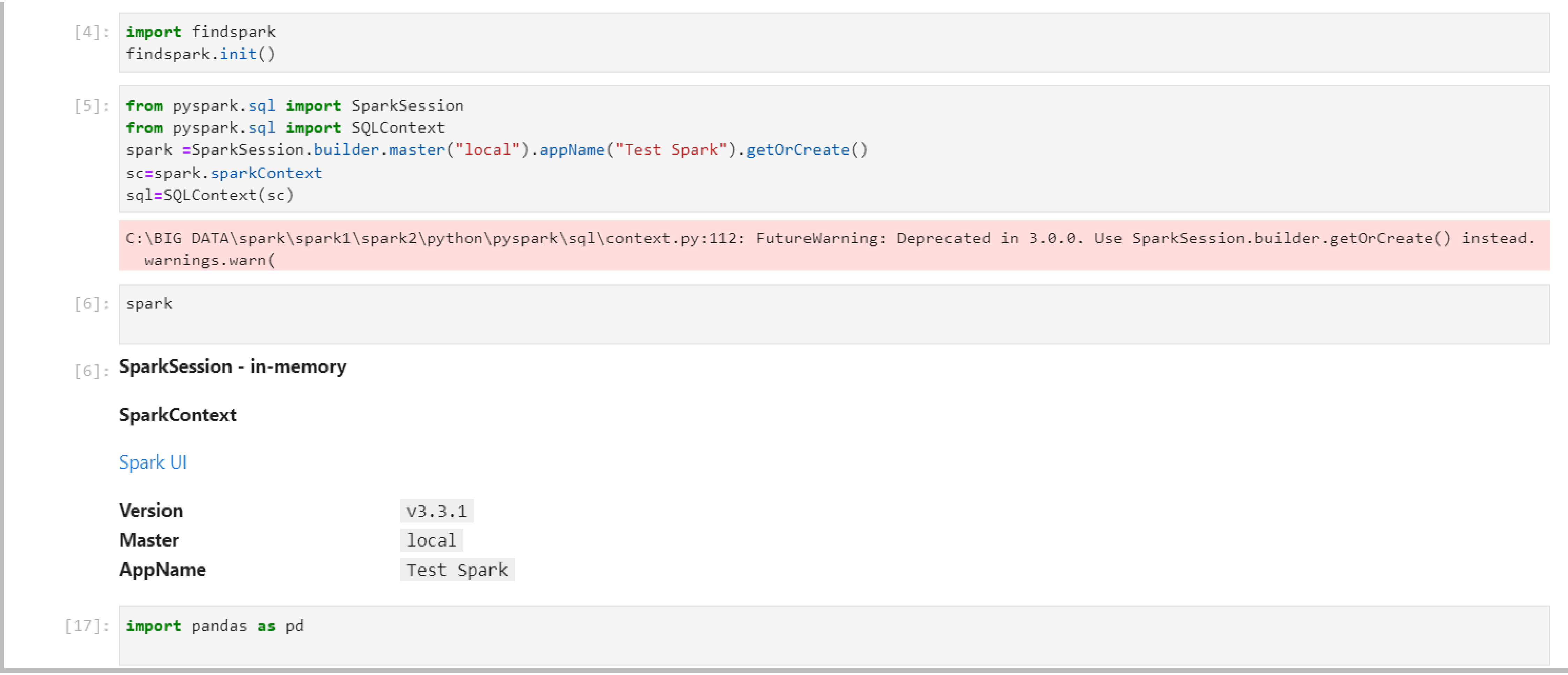


[4]:
```
import findspark
findspark.init()
```

[5]:
```
from pyspark.sql import SparkSession
from pyspark.sql import SQLContext
spark =SparkSession.builder.master("local").appName("Test Spark").getOrCreate()
sc=spark.sparkContext
sql=SQLContext(sc)
```

```
C:\BIG DATA\spark\spark1\spark2\python\pyspark\sql\context.py:112: FutureWarning: Deprecated in 3.0.0. Use SparkSession.builder.getOrCreate() instead.
  warnings.warn(
```

[6]:
```
spark
```

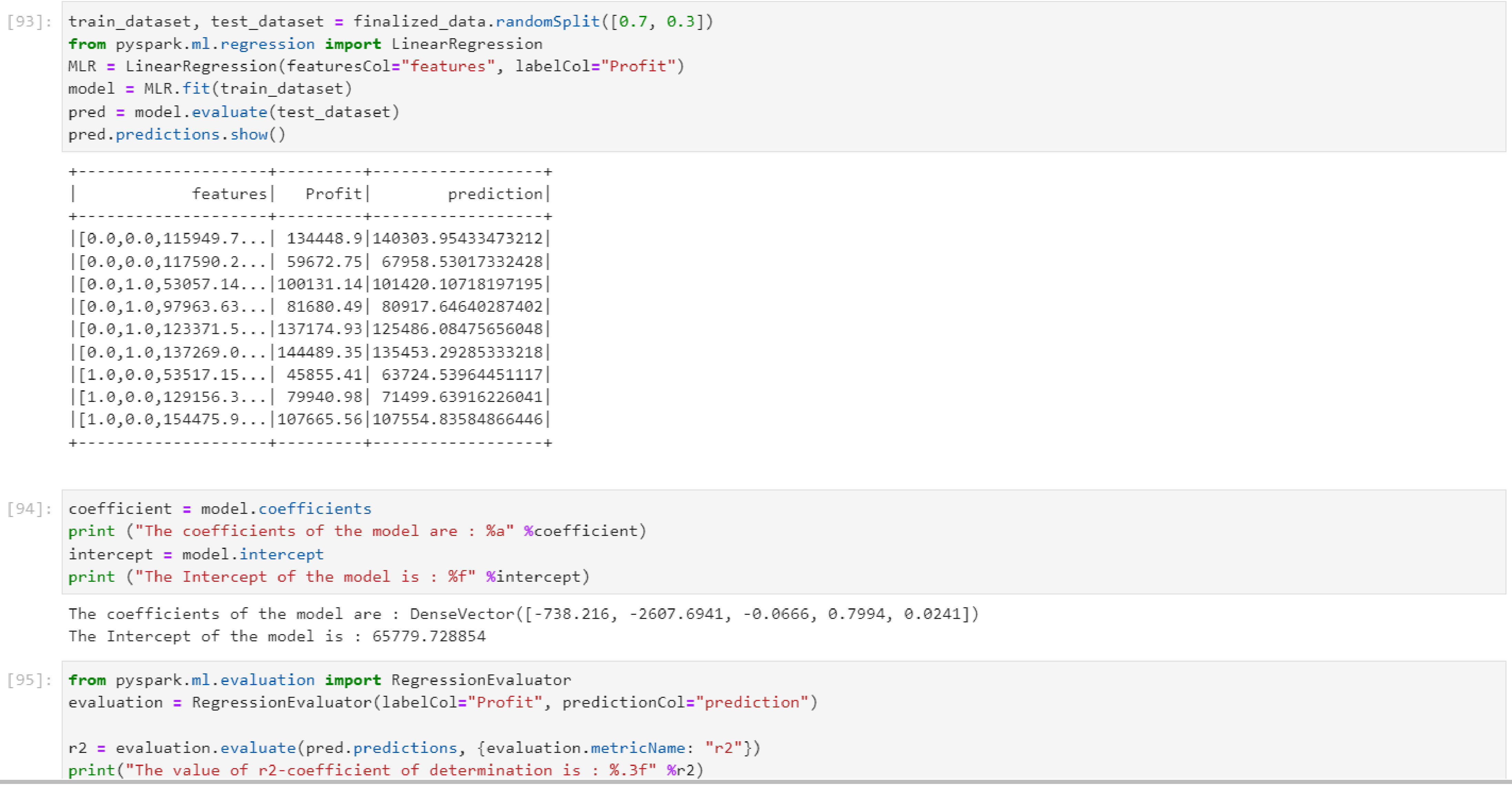
[6]: **SparkSession - in-memory**

SparkContext

Spark UI

Version	`v3.3.1`
Master	`local`
AppName	`Test Spark`

[17]:
```
import pandas as pd
```



[93]:

```
train_dataset, test_dataset = finalized_data.randomSplit([0.7, 0.3])
from pyspark.ml.regression import LinearRegression
MLR = LinearRegression(featuresCol="features", labelCol="Profit")
model = MLR.fit(train_dataset)
pred = model.evaluate(test_dataset)
pred.predictions.show()
```

```
+--------------------+---------+-----------------+
|            features|   Profit|       prediction|
+--------------------+---------+-----------------+
|[0.0,0.0,115949.7...| 134448.9|140303.95433473212|
|[0.0,0.0,117590.2...| 59672.75| 67958.53017332428|
|[0.0,1.0,53057.14...|100131.14|101420.10718197195|
|[0.0,1.0,97963.63...| 81680.49| 80917.64640287402|
|[0.0,1.0,123371.5...|137174.93|125486.08475656048|
|[0.0,1.0,137269.0...|144489.35|135453.29285333218|
|[1.0,0.0,53517.15...| 45855.41| 63724.53964451117|
|[1.0,0.0,129156.3...| 79940.98| 71499.63916226041|
|[1.0,0.0,154475.9...|107665.56|107554.83584866446|
+--------------------+---------+-----------------+
```

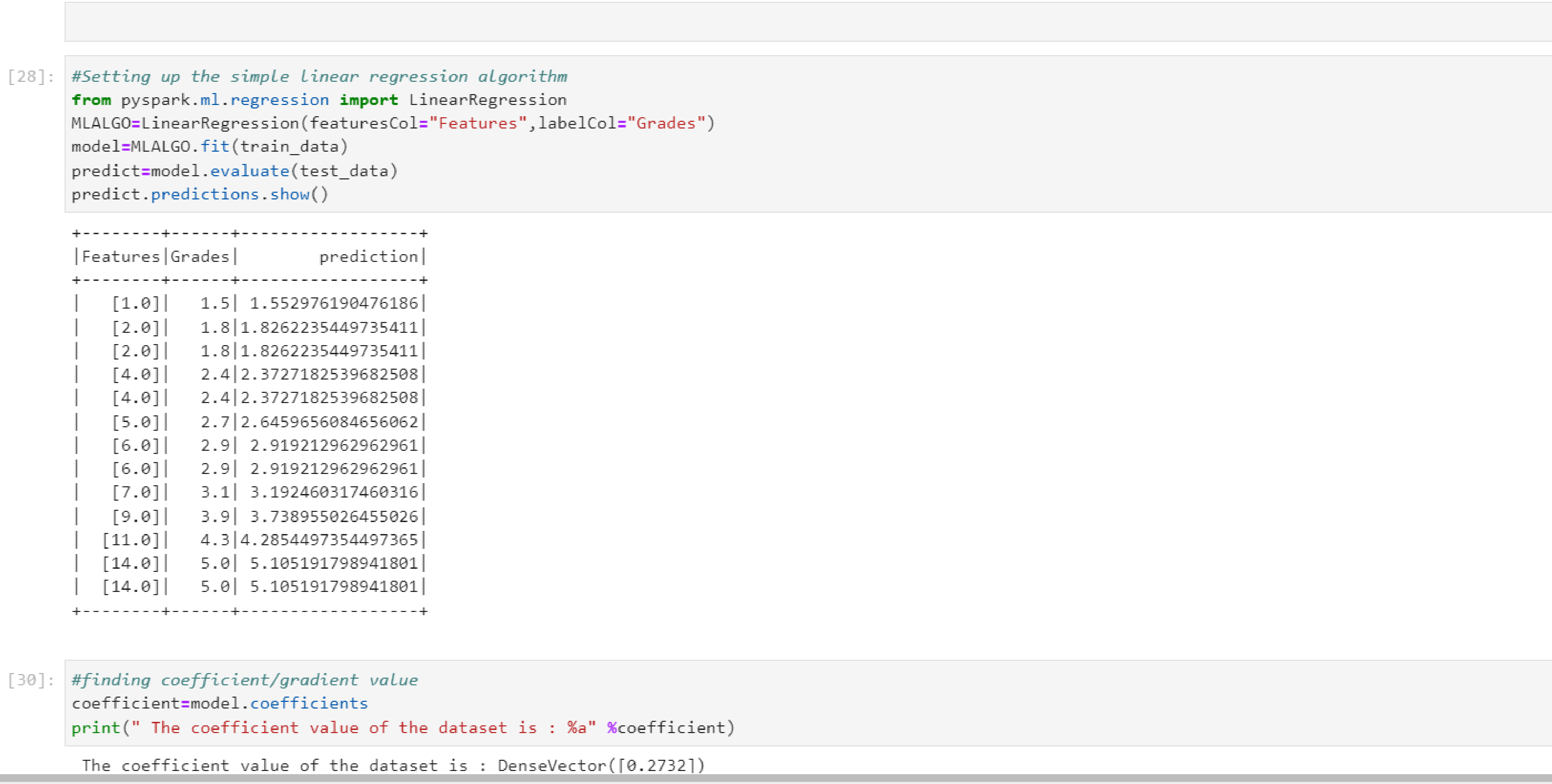
[94]:

```
coefficient = model.coefficients
print ("The coefficients of the model are : %a" %coefficient)
intercept = model.intercept
print ("The Intercept of the model is : %f" %intercept)
```

```
The coefficients of the model are : DenseVector([-738.216, -2607.6941, -0.0666, 0.7994, 0.0241])
The Intercept of the model is : 65779.728854
```

[95]:

```
from pyspark.ml.evaluation import RegressionEvaluator
evaluation = RegressionEvaluator(labelCol="Profit", predictionCol="prediction")

r2 = evaluation.evaluate(pred.predictions, {evaluation.metricName: "r2"})
print("The value of r2-coefficient of determination is : %.3f" %r2)
```

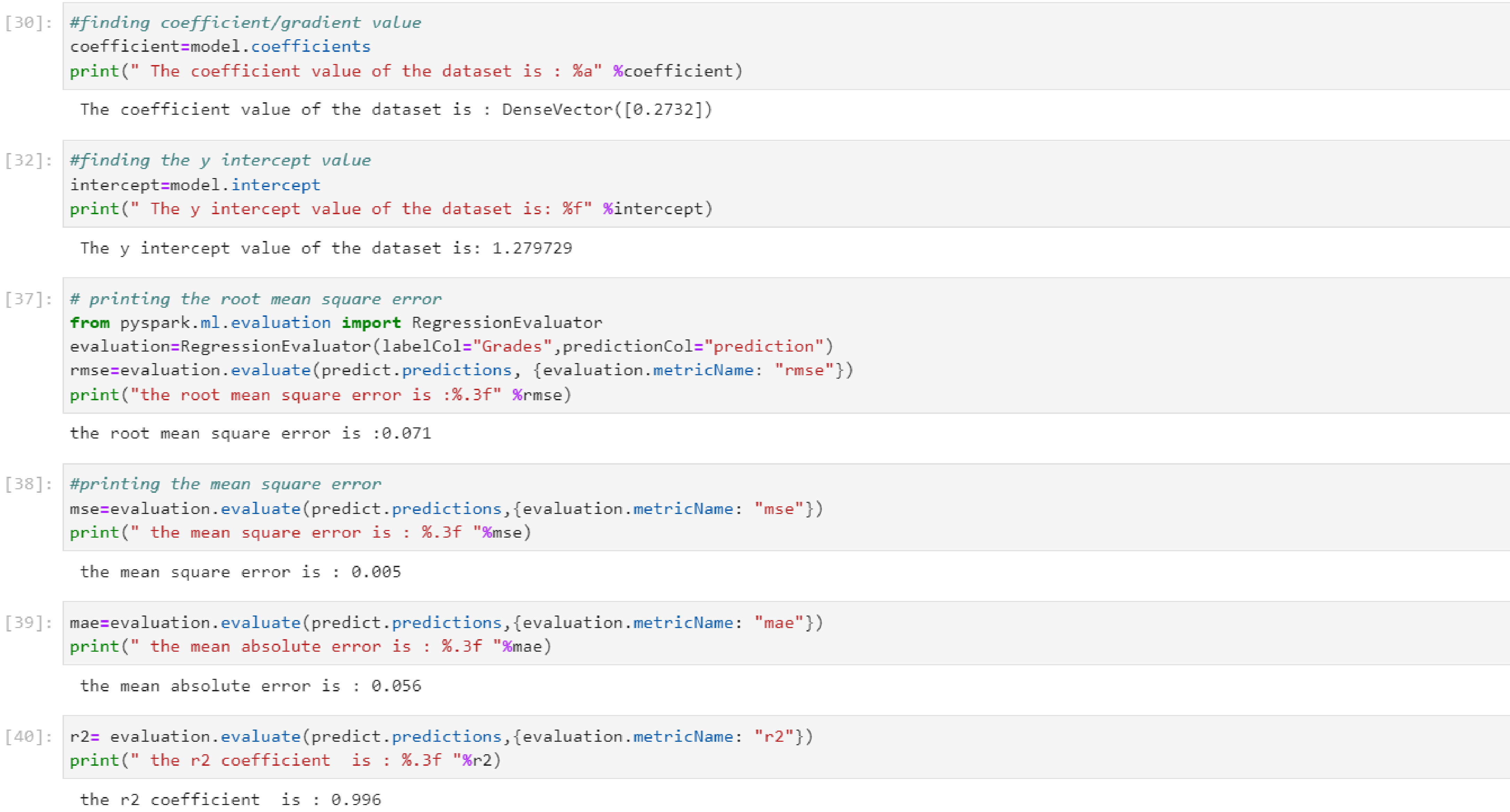


[28]:
```
#Setting up the simple linear regression algorithm
from pyspark.ml.regression import LinearRegression
MLALGO=LinearRegression(featuresCol="Features",labelCol="Grades")
model=MLALGO.fit(train_data)
predict=model.evaluate(test_data)
predict.predictions.show()
```

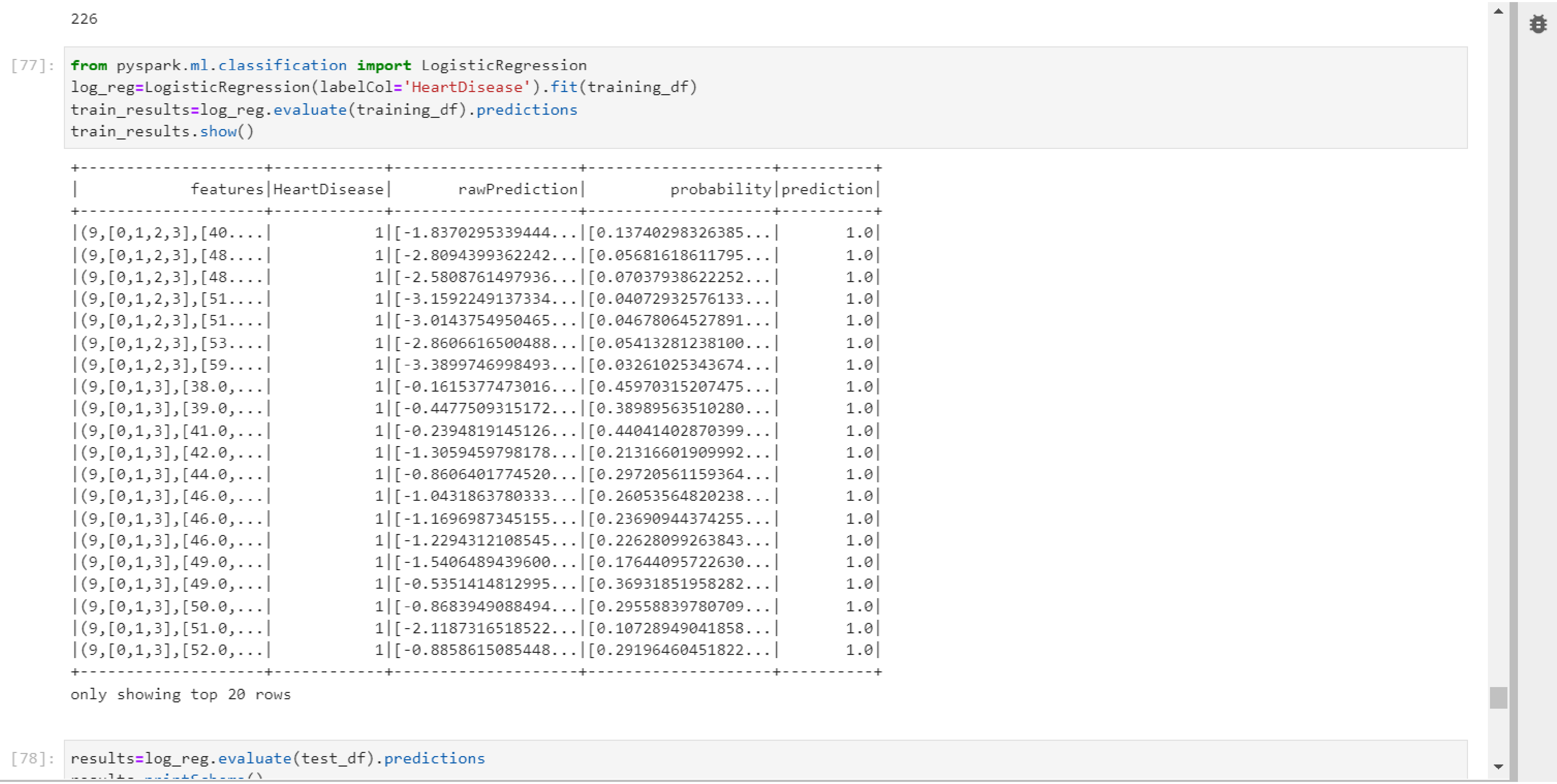
```
+--------+------+------------------+
|Features|Grades|        prediction|
+--------+------+------------------+
|   [1.0]|   1.5| 1.552976190476186|
|   [2.0]|   1.8|1.8262235449735411|
|   [2.0]|   1.8|1.8262235449735411|
|   [4.0]|   2.4|2.3727182539682508|
|   [4.0]|   2.4|2.3727182539682508|
|   [5.0]|   2.7|2.6459656084656062|
|   [6.0]|   2.9| 2.919212962962961|
|   [6.0]|   2.9| 2.919212962962961|
|   [7.0]|   3.1| 3.192460317460316|
|   [9.0]|   3.9| 3.738955026455026|
|  [11.0]|   4.3|4.2854497354497365|
|  [14.0]|   5.0| 5.105191798941801|
|  [14.0]|   5.0| 5.105191798941801|
+--------+------+------------------+
```

[30]:
```
#finding coefficient/gradient value
coefficient=model.coefficients
print(" The coefficient value of the dataset is : %a" %coefficient)
```

```
 The coefficient value of the dataset is : DenseVector([0.2732])
```



```
[30]: #finding coefficient/gradient value
      coefficient=model.coefficients
      print(" The coefficient value of the dataset is : %a" %coefficient)

       The coefficient value of the dataset is : DenseVector([0.2732])

[32]: #finding the y intercept value
      intercept=model.intercept
      print(" The y intercept value of the dataset is: %f" %intercept)

       The y intercept value of the dataset is: 1.279729

[37]: # printing the root mean square error
      from pyspark.ml.evaluation import RegressionEvaluator
      evaluation=RegressionEvaluator(labelCol="Grades",predictionCol="prediction")
      rmse=evaluation.evaluate(predict.predictions, {evaluation.metricName: "rmse"})
      print("the root mean square error is :%.3f" %rmse)

      the root mean square error is :0.071

[38]: #printing the mean square error
      mse=evaluation.evaluate(predict.predictions,{evaluation.metricName: "mse"})
      print(" the mean square error is : %.3f "%mse)

       the mean square error is : 0.005

[39]: mae=evaluation.evaluate(predict.predictions,{evaluation.metricName: "mae"})
      print(" the mean absolute error is : %.3f "%mae)

       the mean absolute error is : 0.056

[40]: r2= evaluation.evaluate(predict.predictions,{evaluation.metricName: "r2"})
      print(" the r2 coefficient  is : %.3f "%r2)

       the r2 coefficient  is : 0.996
```

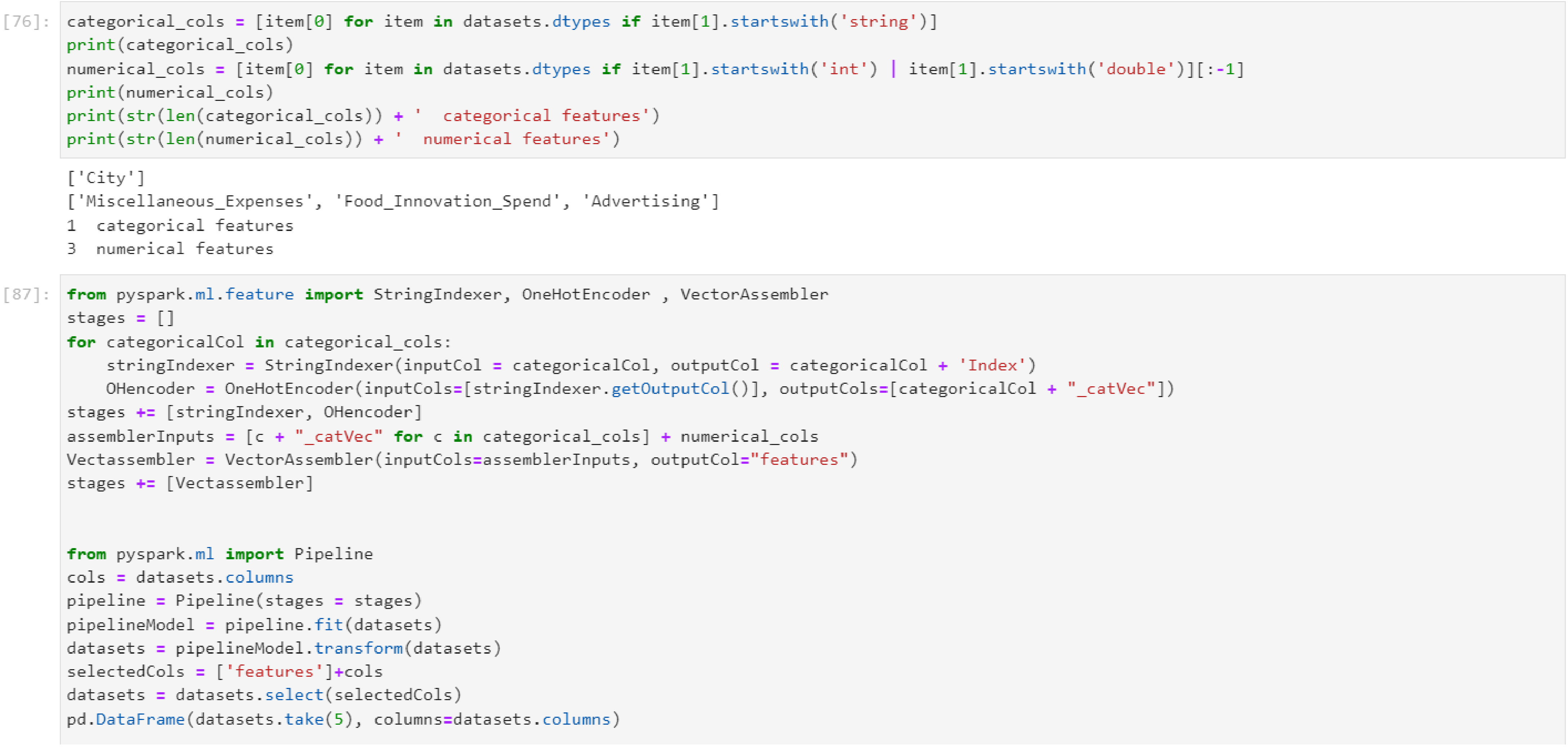


```
226
```

[77]:
```python
from pyspark.ml.classification import LogisticRegression
log_reg=LogisticRegression(labelCol='HeartDisease').fit(training_df)
train_results=log_reg.evaluate(training_df).predictions
train_results.show()
```

```
+--------------------+------------+--------------------+--------------------+----------+
|            features|HeartDisease|       rawPrediction|         probability|prediction|
+--------------------+------------+--------------------+--------------------+----------+
|(9,[0,1,2,3],[40....|           1|[-1.8370295339444...|[0.13740298326385...|       1.0|
|(9,[0,1,2,3],[48....|           1|[-2.8094399362242...|[0.05681618611795...|       1.0|
|(9,[0,1,2,3],[48....|           1|[-2.5808761497936...|[0.07037938622252...|       1.0|
|(9,[0,1,2,3],[51....|           1|[-3.1592249137334...|[0.04072932576133...|       1.0|
|(9,[0,1,2,3],[51....|           1|[-3.0143754950465...|[0.04678064527891...|       1.0|
|(9,[0,1,2,3],[53....|           1|[-2.8606616500488...|[0.05413281238100...|       1.0|
|(9,[0,1,2,3],[59....|           1|[-3.3899746998493...|[0.03261025343674...|       1.0|
|(9,[0,1,3],[38.0,...|           1|[-0.1615377473016...|[0.45970315207475...|       1.0|
|(9,[0,1,3],[39.0,...|           1|[-0.4477509315172...|[0.38989563510280...|       1.0|
|(9,[0,1,3],[41.0,...|           1|[-0.2394819145126...|[0.44041402870399...|       1.0|
|(9,[0,1,3],[42.0,...|           1|[-1.3059459798178...|[0.21316601909992...|       1.0|
|(9,[0,1,3],[44.0,...|           1|[-0.8606401774520...|[0.29720561159364...|       1.0|
|(9,[0,1,3],[46.0,...|           1|[-1.0431863780333...|[0.26053564820238...|       1.0|
|(9,[0,1,3],[46.0,...|           1|[-1.1696987345155...|[0.23690944374255...|       1.0|
|(9,[0,1,3],[46.0,...|           1|[-1.2294312108545...|[0.22628099263843...|       1.0|
|(9,[0,1,3],[49.0,...|           1|[-1.5406489439600...|[0.17644095722630...|       1.0|
|(9,[0,1,3],[49.0,...|           1|[-0.5351414812995...|[0.36931851958282...|       1.0|
|(9,[0,1,3],[50.0,...|           1|[-0.8683949088494...|[0.29558839780709...|       1.0|
|(9,[0,1,3],[51.0,...|           1|[-2.1187316518522...|[0.10728949041858...|       1.0|
|(9,[0,1,3],[52.0,...|           1|[-0.8858615085448...|[0.29196460451822...|       1.0|
+--------------------+------------+--------------------+--------------------+----------+
only showing top 20 rows
```

[78]:
```python
results=log_reg.evaluate(test_df).predictions
results.printSchema()
```



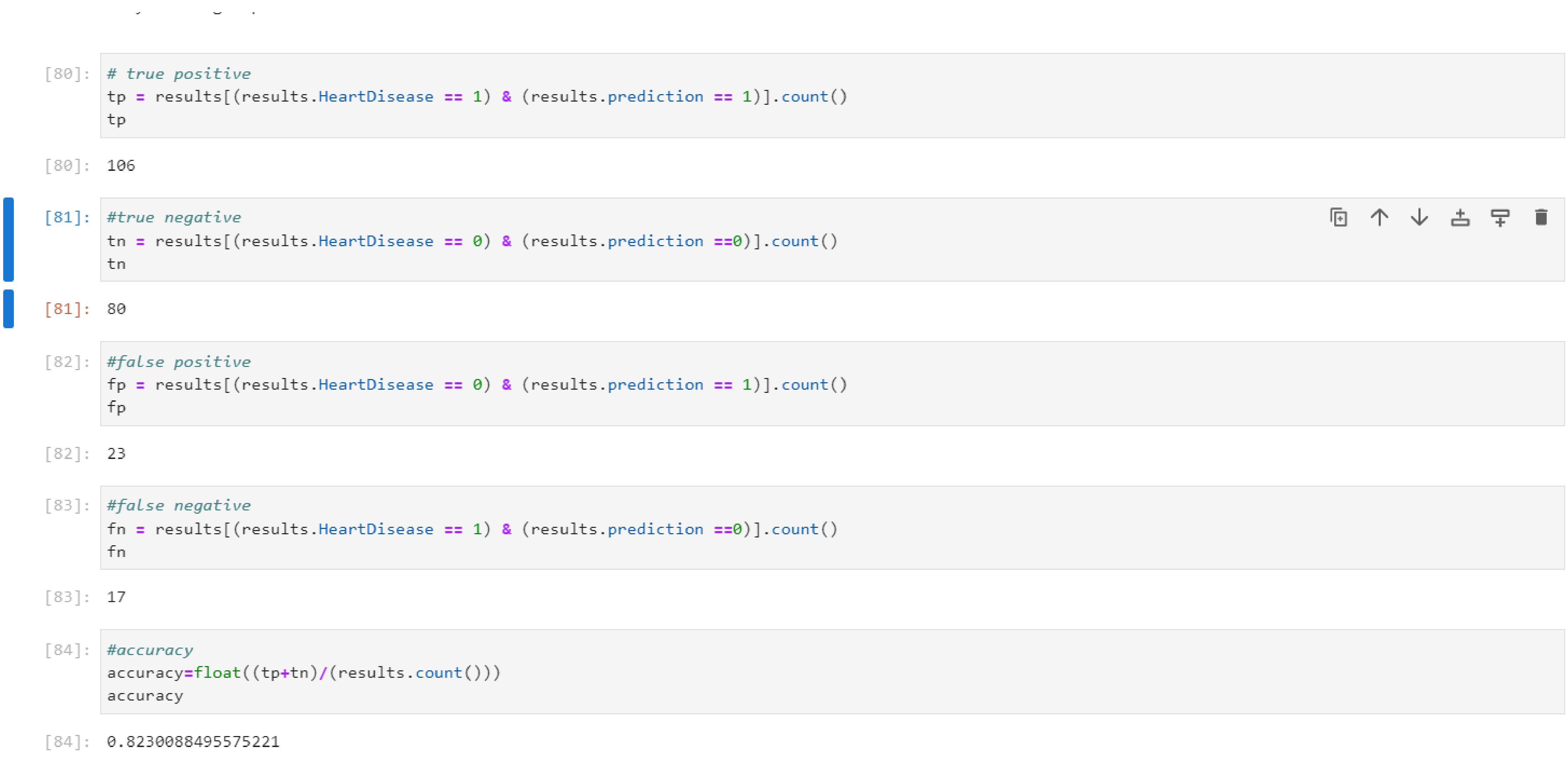
[76]:

```python
categorical_cols = [item[0] for item in datasets.dtypes if item[1].startswith('string')]
print(categorical_cols)
numerical_cols = [item[0] for item in datasets.dtypes if item[1].startswith('int') | item[1].startswith('double')][:-1]
print(numerical_cols)
print(str(len(categorical_cols)) + '  categorical features')
print(str(len(numerical_cols)) + '  numerical features')
```

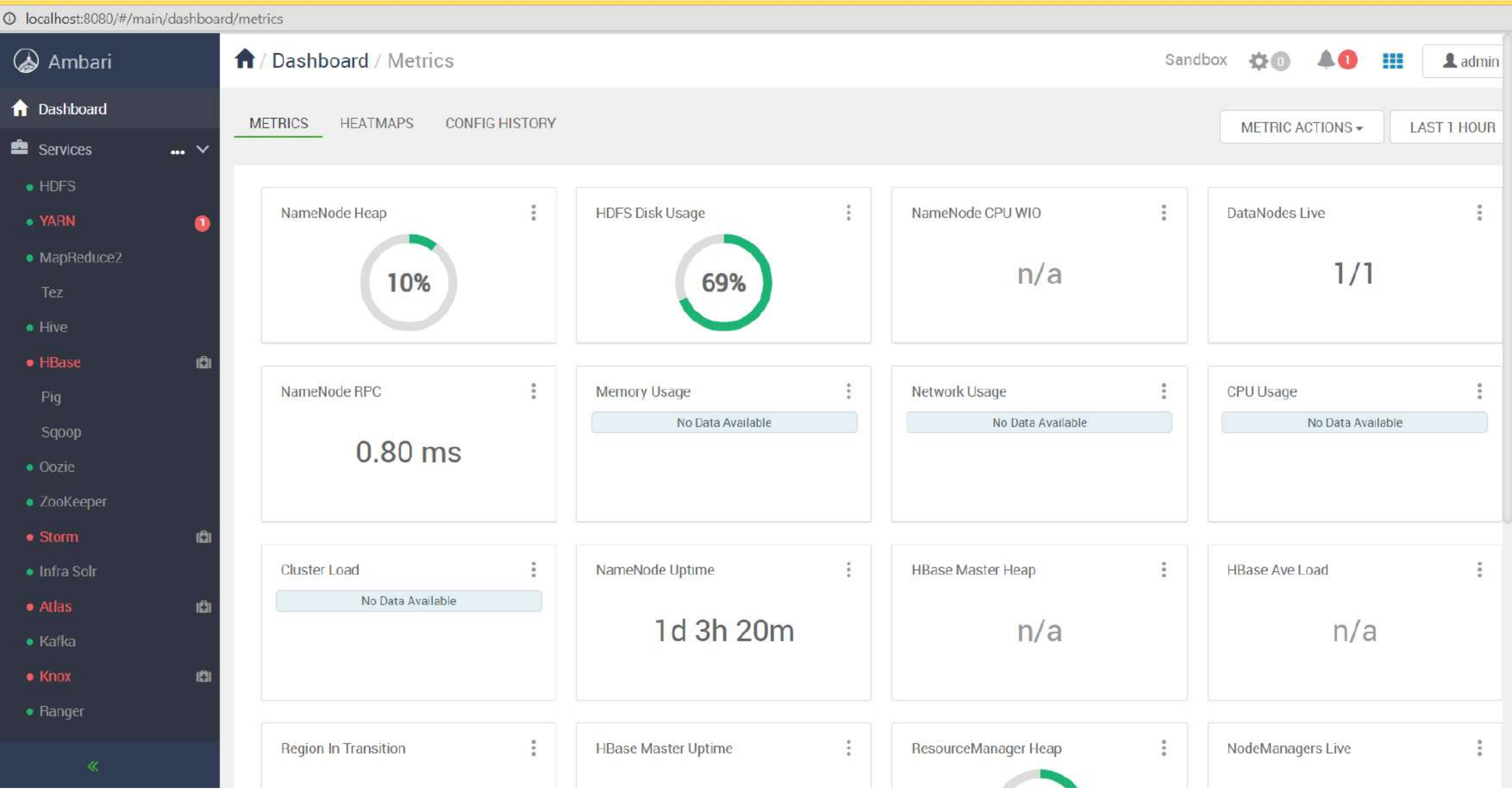
```
['City']
['Miscellaneous_Expenses', 'Food_Innovation_Spend', 'Advertising']
1  categorical features
3  numerical features
```

[87]:

```python
from pyspark.ml.feature import StringIndexer, OneHotEncoder , VectorAssembler
stages = []
for categoricalCol in categorical_cols:
    stringIndexer = StringIndexer(inputCol = categoricalCol, outputCol = categoricalCol + 'Index')
    OHencoder = OneHotEncoder(inputCols=[stringIndexer.getOutputCol()], outputCols=[categoricalCol + "_catVec"])
stages += [stringIndexer, OHencoder]
assemblerInputs = [c + "_catVec" for c in categorical_cols] + numerical_cols
Vectassembler = VectorAssembler(inputCols=assemblerInputs, outputCol="features")
stages += [Vectassembler]


from pyspark.ml import Pipeline
cols = datasets.columns
pipeline = Pipeline(stages = stages)
pipelineModel = pipeline.fit(datasets)
datasets = pipelineModel.transform(datasets)
selectedCols = ['features']+cols
datasets = datasets.select(selectedCols)
pd.DataFrame(datasets.take(5), columns=datasets.columns)
```



```
[80]: # true positive
      tp = results[(results.HeartDisease == 1) & (results.prediction == 1)].count()
      tp

[80]: 106

[81]: #true negative
      tn = results[(results.HeartDisease == 0) & (results.prediction ==0)].count()
      tn

[81]: 80

[82]: #false positive
      fp = results[(results.HeartDisease == 0) & (results.prediction == 1)].count()
      fp

[82]: 23

[83]: #false negative
      fn = results[(results.HeartDisease == 1) & (results.prediction ==0)].count()
      fn

[83]: 17

[84]: #accuracy
      accuracy=float((tp+tn)/(results.count()))
      accuracy

[84]: 0.8230088495575221
```



localhost:8080/#/main/dashboard/metrics

Ambari

Dashboard / Metrics

Sandbox ⚙ 0 🔔 1 admin

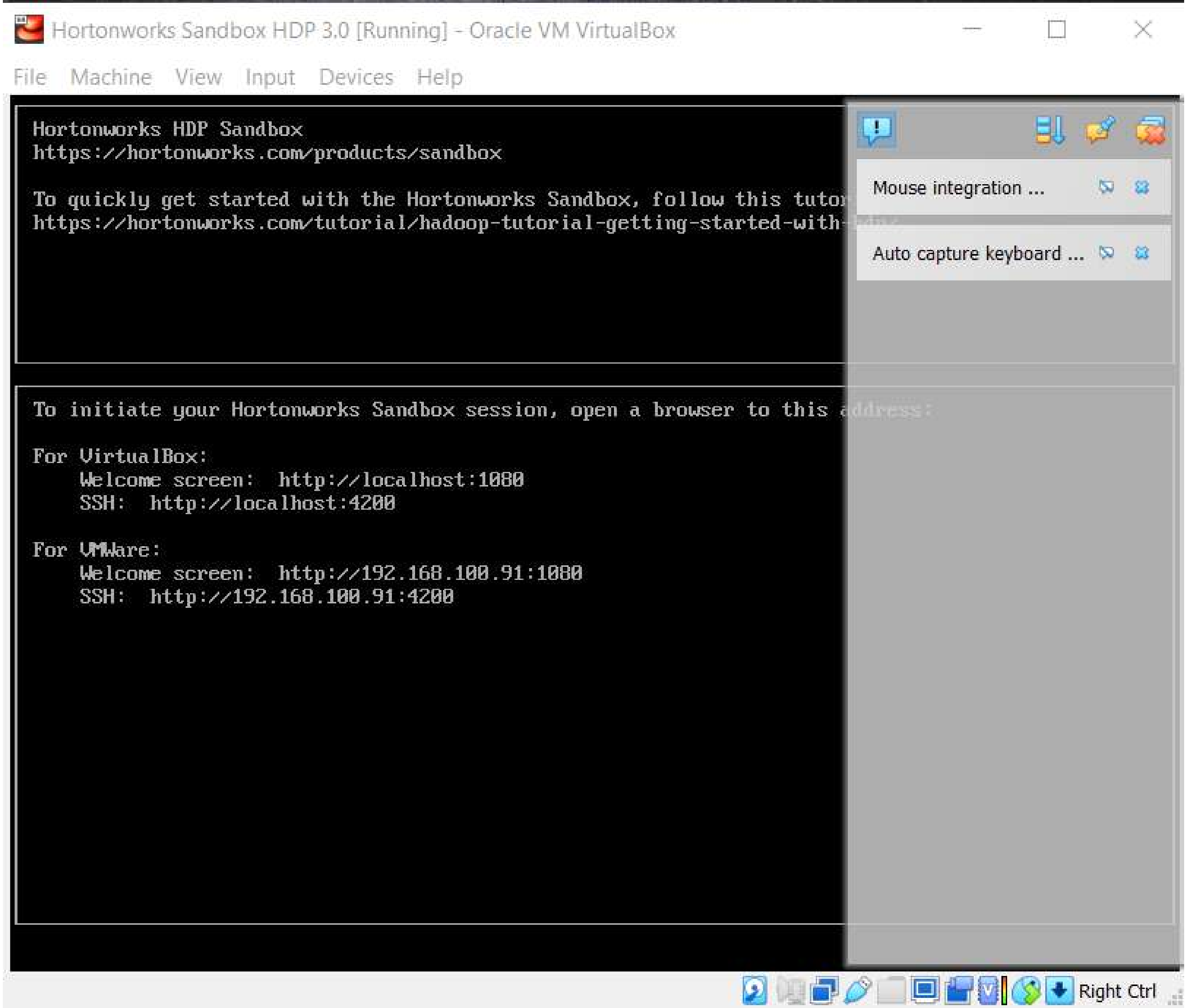
- Dashboard
- Services ...
- HDFS
- YARN 1
- MapReduce2
- Tez
- Hive
- HBase
- Pig
- Sqoop
- Oozie
- ZooKeeper
- Storm
- Infra Solr
- Atlas
- Kafka
- Knox
- Ranger

«

METRICS HEATMAPS CONFIG HISTORY

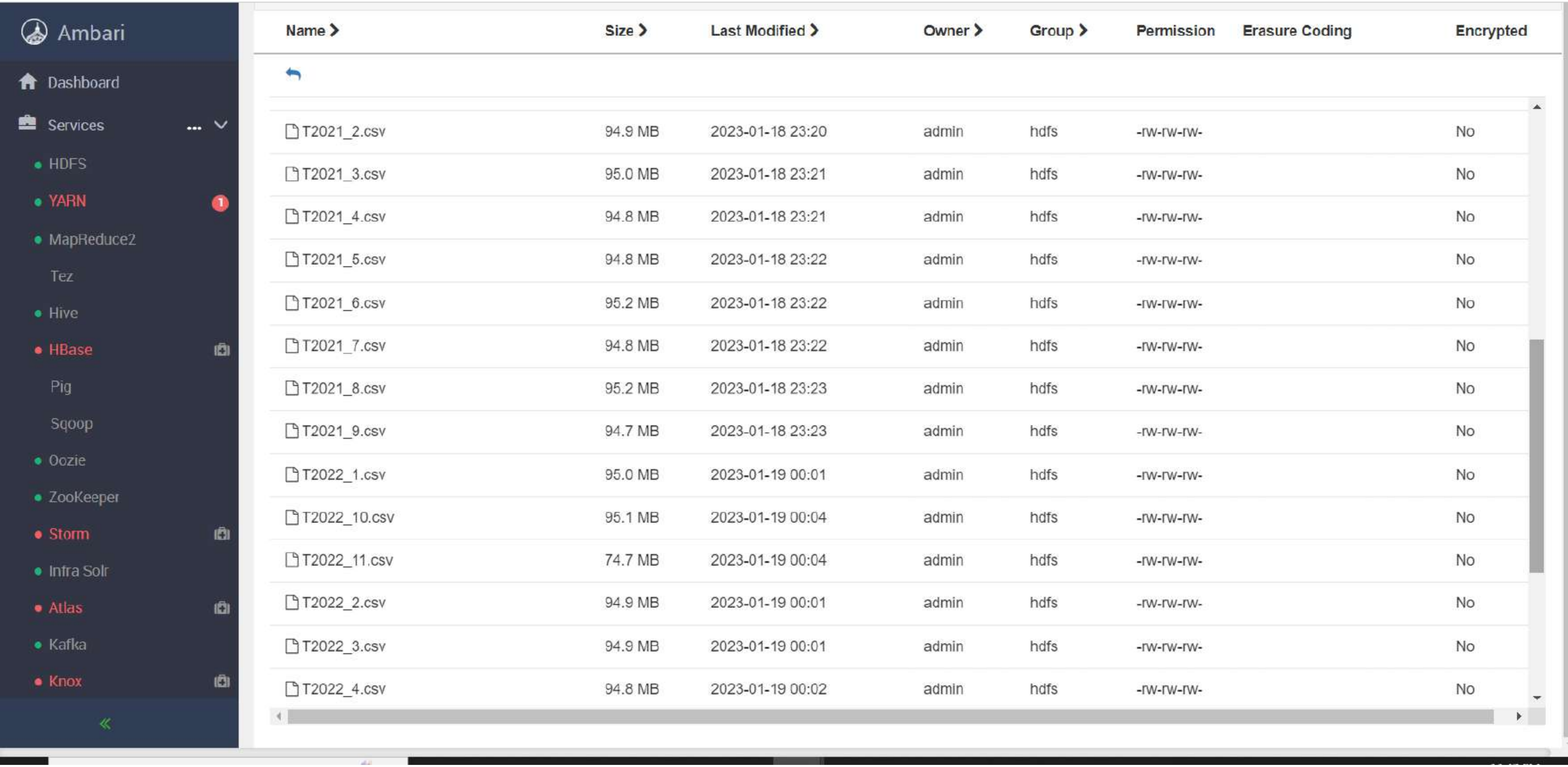
METRIC ACTIONS ▾ LAST 1 HOUR

NameNode Heap 10%	HDFS Disk Usage 69%	NameNode CPU WIO n/a	DataNodes Live 1/1
NameNode RPC 0.80 ms	Memory Usage No Data Available	Network Usage No Data Available	CPU Usage No Data Available
Cluster Load No Data Available	NameNode Uptime 1d 3h 20m	HBase Master Heap n/a	HBase Ave Load n/a
Region In Transition	HBase Master Uptime	ResourceManager Heap	NodeManagers Live

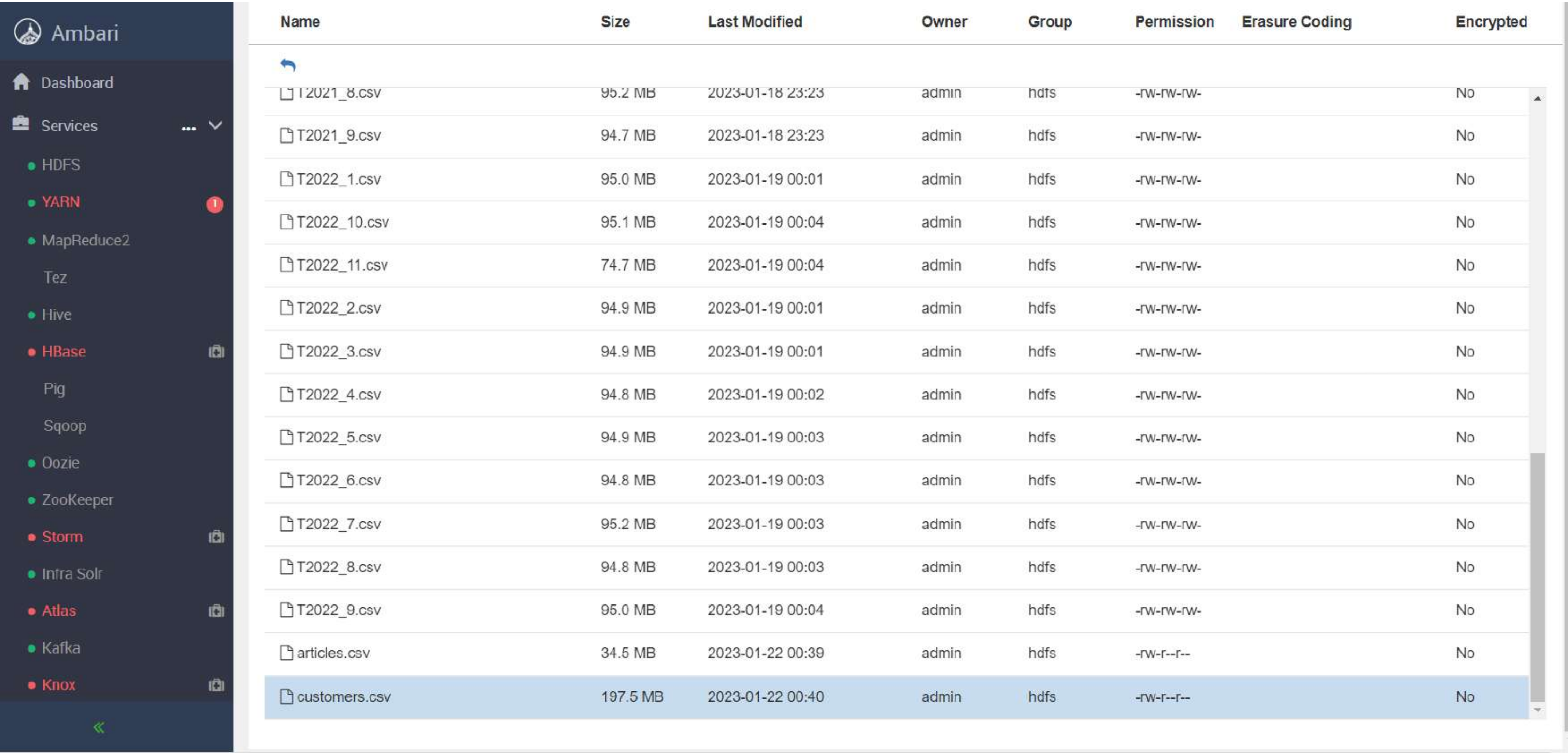

Hortonworks Sandbox HDP 3.0 [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Mouse integration ...
Auto capture keyboard ...
Right Ctrl

```
Hortonworks HDP Sandbox
https://hortonworks.com/products/sandbox

To quickly get started with the Hortonworks Sandbox, follow this tutor
https://hortonworks.com/tutorial/hadoop-tutorial-getting-started-with-hdp/
```

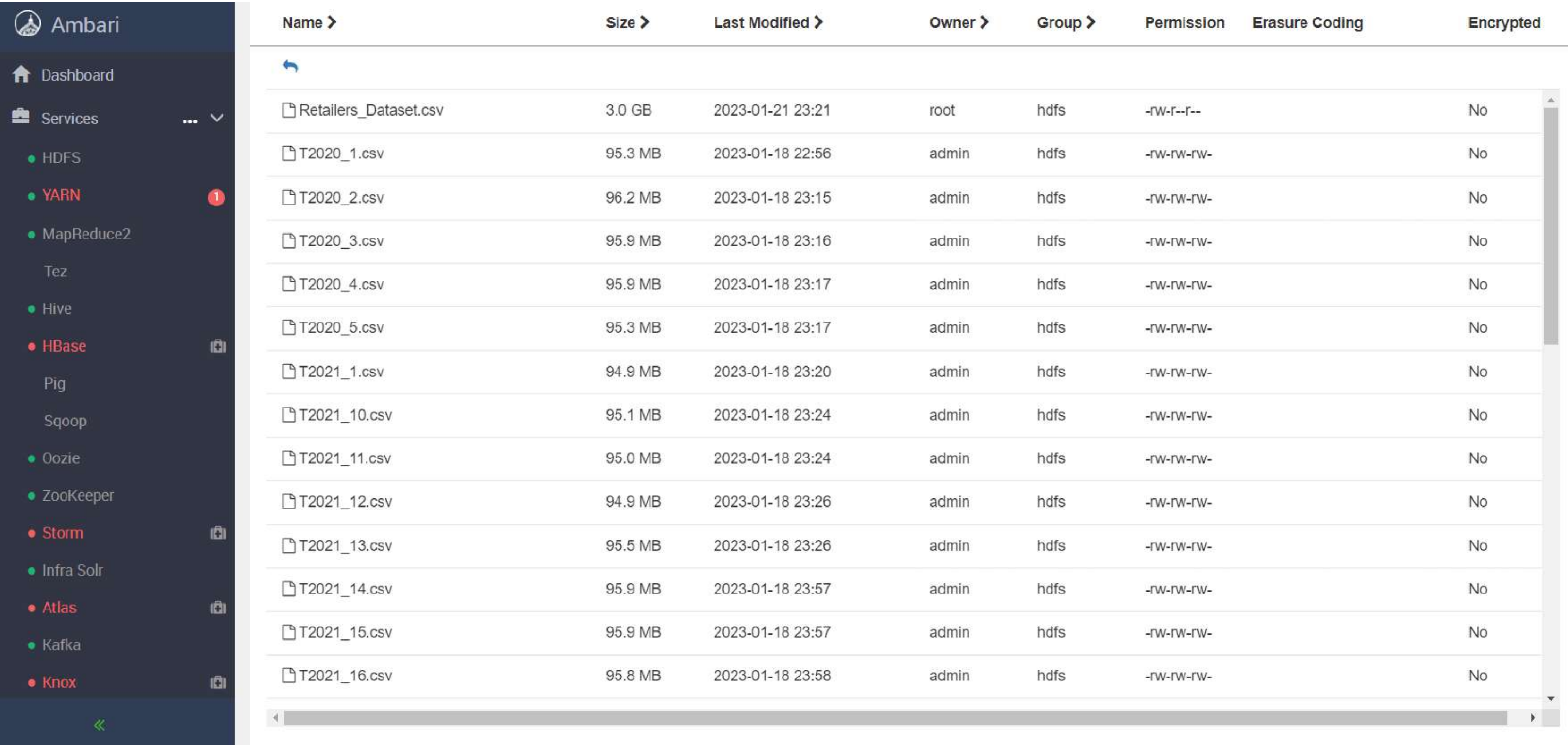
```
To initiate your Hortonworks Sandbox session, open a browser to this address:

For VirtualBox:
    Welcome screen:  http://localhost:1080
    SSH:  http://localhost:4200

For VMWare:
    Welcome screen:  http://192.168.100.91:1080
    SSH:  http://192.168.100.91:4200
```



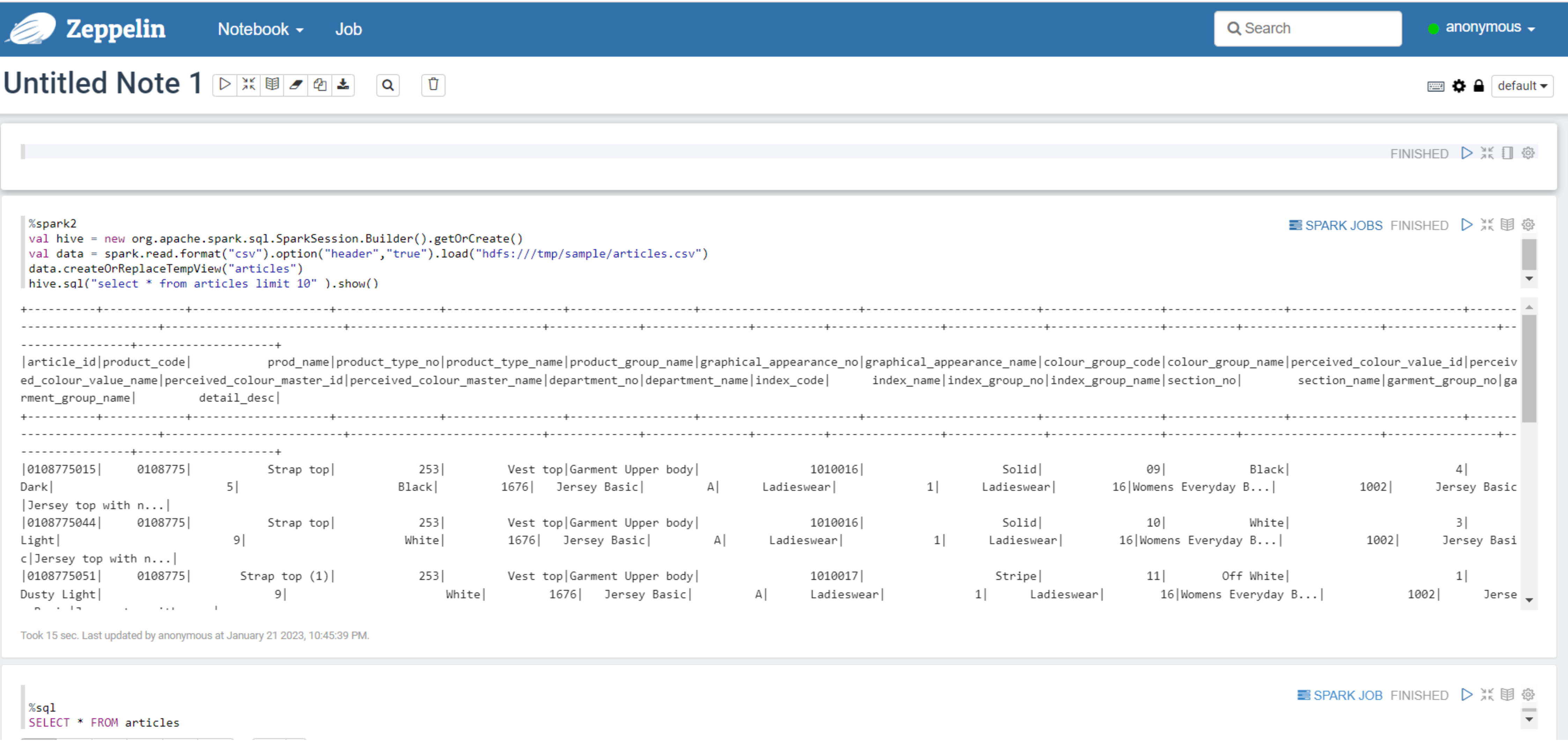
Name	Size	Last Modified	Owner	Group	Permission	Erasure Coding	Encrypted
T2021_2.csv	94.9 MB	2023-01-18 23:20	admin	hdfs	-rw-rw-rw-		No
T2021_3.csv	95.0 MB	2023-01-18 23:21	admin	hdfs	-rw-rw-rw-		No
T2021_4.csv	94.8 MB	2023-01-18 23:21	admin	hdfs	-rw-rw-rw-		No
T2021_5.csv	94.8 MB	2023-01-18 23:22	admin	hdfs	-rw-rw-rw-		No
T2021_6.csv	95.2 MB	2023-01-18 23:22	admin	hdfs	-rw-rw-rw-		No
T2021_7.csv	94.8 MB	2023-01-18 23:22	admin	hdfs	-rw-rw-rw-		No
T2021_8.csv	95.2 MB	2023-01-18 23:23	admin	hdfs	-rw-rw-rw-		No
T2021_9.csv	94.7 MB	2023-01-18 23:23	admin	hdfs	-rw-rw-rw-		No
T2022_1.csv	95.0 MB	2023-01-19 00:01	admin	hdfs	-rw-rw-rw-		No
T2022_10.csv	95.1 MB	2023-01-19 00:04	admin	hdfs	-rw-rw-rw-		No
T2022_11.csv	74.7 MB	2023-01-19 00:04	admin	hdfs	-rw-rw-rw-		No
T2022_2.csv	94.9 MB	2023-01-19 00:01	admin	hdfs	-rw-rw-rw-		No
T2022_3.csv	94.9 MB	2023-01-19 00:01	admin	hdfs	-rw-rw-rw-		No
T2022_4.csv	94.8 MB	2023-01-19 00:02	admin	hdfs	-rw-rw-rw-		No



Name	Size	Last Modified	Owner	Group	Permission	Erasure Coding	Encrypted
T2021_8.csv	95.2 MB	2023-01-18 23:23	admin	hdfs	-rw-rw-rw-		No
T2021_9.csv	94.7 MB	2023-01-18 23:23	admin	hdfs	-rw-rw-rw-		No
T2022_1.csv	95.0 MB	2023-01-19 00:01	admin	hdfs	-rw-rw-rw-		No
T2022_10.csv	95.1 MB	2023-01-19 00:04	admin	hdfs	-rw-rw-rw-		No
T2022_11.csv	74.7 MB	2023-01-19 00:04	admin	hdfs	-rw-rw-rw-		No
T2022_2.csv	94.9 MB	2023-01-19 00:01	admin	hdfs	-rw-rw-rw-		No
T2022_3.csv	94.9 MB	2023-01-19 00:01	admin	hdfs	-rw-rw-rw-		No
T2022_4.csv	94.8 MB	2023-01-19 00:02	admin	hdfs	-rw-rw-rw-		No
T2022_5.csv	94.9 MB	2023-01-19 00:03	admin	hdfs	-rw-rw-rw-		No
T2022_6.csv	94.8 MB	2023-01-19 00:03	admin	hdfs	-rw-rw-rw-		No
T2022_7.csv	95.2 MB	2023-01-19 00:03	admin	hdfs	-rw-rw-rw-		No
T2022_8.csv	94.8 MB	2023-01-19 00:03	admin	hdfs	-rw-rw-rw-		No
T2022_9.csv	95.0 MB	2023-01-19 00:04	admin	hdfs	-rw-rw-rw-		No
articles.csv	34.5 MB	2023-01-22 00:39	admin	hdfs	-rw-r--r--		No
customers.csv	197.5 MB	2023-01-22 00:40	admin	hdfs	-rw-r--r--		No



Name	Size	Last Modified	Owner	Group	Permission	Erasure Coding	Encrypted
Retailers_Dataset.csv	3.0 GB	2023-01-21 23:21	root	hdfs	-rw-r--r--		No
T2020_1.csv	95.3 MB	2023-01-18 22:56	admin	hdfs	-rw-rw-rw-		No
T2020_2.csv	96.2 MB	2023-01-18 23:15	admin	hdfs	-rw-rw-rw-		No
T2020_3.csv	95.9 MB	2023-01-18 23:16	admin	hdfs	-rw-rw-rw-		No
T2020_4.csv	95.9 MB	2023-01-18 23:17	admin	hdfs	-rw-rw-rw-		No
T2020_5.csv	95.3 MB	2023-01-18 23:17	admin	hdfs	-rw-rw-rw-		No
T2021_1.csv	94.9 MB	2023-01-18 23:20	admin	hdfs	-rw-rw-rw-		No
T2021_10.csv	95.1 MB	2023-01-18 23:24	admin	hdfs	-rw-rw-rw-		No
T2021_11.csv	95.0 MB	2023-01-18 23:24	admin	hdfs	-rw-rw-rw-		No
T2021_12.csv	94.9 MB	2023-01-18 23:26	admin	hdfs	-rw-rw-rw-		No
T2021_13.csv	95.5 MB	2023-01-18 23:26	admin	hdfs	-rw-rw-rw-		No
T2021_14.csv	95.9 MB	2023-01-18 23:57	admin	hdfs	-rw-rw-rw-		No
T2021_15.csv	95.9 MB	2023-01-18 23:57	admin	hdfs	-rw-rw-rw-		No
T2021_16.csv	95.8 MB	2023-01-18 23:58	admin	hdfs	-rw-rw-rw-		No



```
%spark2
val hive = new org.apache.spark.sql.SparkSession.Builder().getOrCreate()
val data = spark.read.format("csv").option("header","true").load("hdfs:///tmp/sample/articles.csv")
data.createOrReplaceTempView("articles")
hive.sql("select * from articles limit 10" ).show()
```

```
+----------+------------+--------------------+---------------+-----------------+------------------+-----------------------+-------------------------+-----------------+-----------------+-------------------------+-------
--------------------+--------------------------+----------------------------+-------------+---------------+----------+----------------+--------------+----------------+----------+--------------------+----------------+--
----------------+--------------------+
|article_id|product_code|           prod_name|product_type_no|product_type_name|product_group_name|graphical_appearance_no|graphical_appearance_name|colour_group_code|colour_group_name|perceived_colour_value_id|perceiv
ed_colour_value_name|perceived_colour_master_id|perceived_colour_master_name|department_no|department_name|index_code|      index_name|index_group_no|index_group_name|section_no|        section_name|garment_group_no|ga
rment_group_name|         detail_desc|
+----------+------------+--------------------+---------------+-----------------+------------------+-----------------------+-------------------------+-----------------+-----------------+-------------------------+-------
--------------------+--------------------------+----------------------------+-------------+---------------+----------+----------------+--------------+----------------+----------+--------------------+----------------+--
----------------+--------------------+
|0108775015|     0108775|           Strap top|            253|         Vest top|Garment Upper body|                1010016|                    Solid|               09|            Black|                        4|
Dark|                         5|                       Black|         1676|   Jersey Basic|         A|      Ladieswear|             1|      Ladieswear|        16|Womens Everyday B...|            1002|      Jersey Basic
|Jersey top with n...|
|0108775044|     0108775|           Strap top|            253|         Vest top|Garment Upper body|                1010016|                    Solid|               10|            White|                        3|
Light|                         9|                       White|         1676|   Jersey Basic|         A|      Ladieswear|             1|      Ladieswear|        16|Womens Everyday B...|            1002|      Jersey Basi
c|Jersey top with n...|
|0108775051|     0108775|       Strap top (1)|            253|         Vest top|Garment Upper body|                1010017|                   Stripe|               11|        Off White|                        1|
Dusty Light|                         9|                       White|         1676|   Jersey Basic|         A|      Ladieswear|             1|      Ladieswear|        16|Womens Everyday B...|            1002|      Jerse
```

Took 15 sec. Last updated by anonymous at January 21 2023, 10:45:39 PM.

```
%sql
SELECT * FROM articles
```