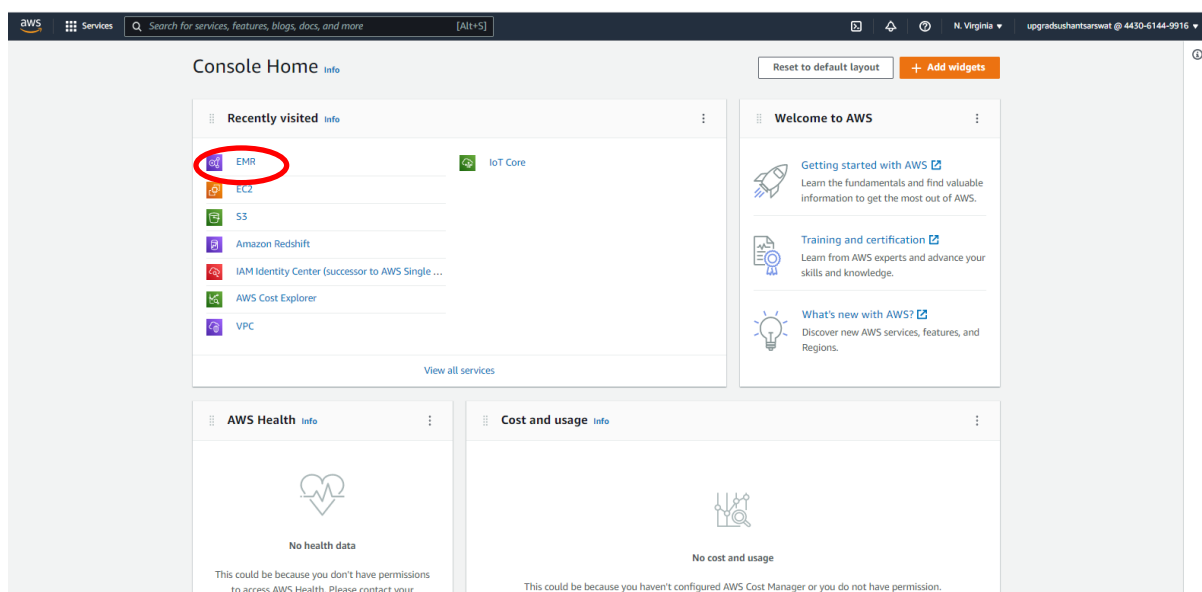


Code Logic - Retail Data Analysis

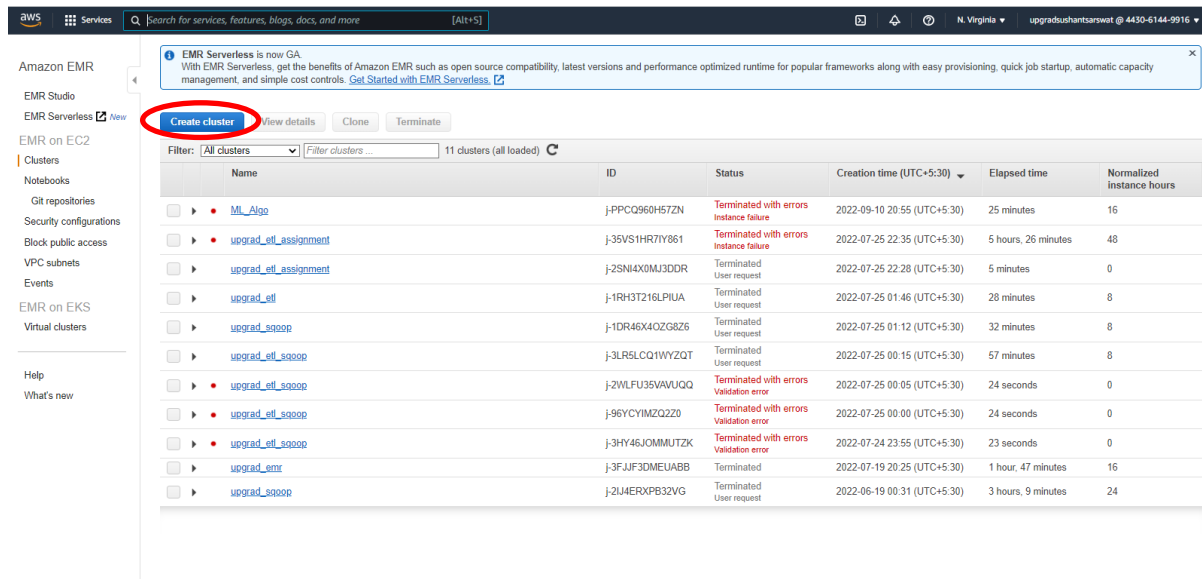
Submitted By : Sushant Sarswat, DE C-38

A step by step logic employed to solve the abovementioned problem is elucidated in the following paragraphs:

1. Setting up of the EMR Cluster:



2. Create Cluster



3. Go to Advanced Options

aws Services Search for services, features, blogs, docs, and more [Alt+S]

EMR Serverless is now GA. With EMR Serverless, get the benefits of Amazon EMR such as open source compatibility, latest versions and performance optimized runtime for popular frameworks along with easy provisioning, quick job startup, automatic capacity management, and simple cost controls. [Get Started with EMR Serverless.](#)

Create Cluster - Quick Options [Go to advanced options](#)

General Configuration

Cluster name

☒ Logging ⓘ

S3 folder

Launch mode ☒ Cluster ⓘ ☐ Step execution ⓘ

Software configuration

Release

Applications

- ☒ Core Hadoop: Hadoop 2.10.1, Hive 2.3.9, Hue 4.10.0, Mahout 0.13.0, Pig 0.17.0, and Tez 0.9.2
- ☐ HBase: HBase 1.4.13, Hadoop 2.10.1, Hive 2.3.9, Hue 4.10.0, Phoenix 4.14.3, and ZooKeeper 3.4.14
- ☐ Presto: Presto 0.267 with Hadoop 2.10.1 HDFS and Hive 2.3.9 Metastore
- ☐ Spark: Spark 2.4.8 on Hadoop 2.10.1 YARN and Zeppelin 0.10.0
- ☐ Use AWS Glue Data Catalog for table metadata ⓘ

Hardware configuration

Instance type The selected instance type adds 64 GiB of GP2 EBS storage per instance by default. [Learn more](#)

Number of instances (1 master and 2 core nodes)

Cluster scaling ☐ scale cluster nodes based on workload

Auto-termination ☒ Enable auto-termination. [Learn more](#)

4. Software Configuration

aws Services admin

EMR Serverless is now GA. With EMR Serverless, get the benefits of Amazon EMR such as open source compatibility, latest versions and performance optimized runtime for popular frameworks along with easy provisioning, quick job startup, automatic capacity management, and simple cost controls. [Get Started with EMR Serverless.](#)

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps

Step 2: Hardware

Step 3: General Cluster Settings

Step 4: Security

Software Configuration

Release

<input checked="" type="checkbox"/> Hadoop 2.10.1	<input type="checkbox"/> Zeppelin 0.10.0	<input checked="" type="checkbox"/> Livy 0.7.1
<input checked="" type="checkbox"/> JupyterHub 1.4.1	<input type="checkbox"/> Tez 0.9.2	<input type="checkbox"/> Flink 1.14.2
<input type="checkbox"/> Ganglia 3.7.2	<input type="checkbox"/> HBase 1.4.13	<input checked="" type="checkbox"/> Pig 0.17.0
<input checked="" type="checkbox"/> Hive 2.3.9	<input type="checkbox"/> Presto 0.267	<input type="checkbox"/> ZooKeeper 3.4.14
<input checked="" type="checkbox"/> JupyterEnterpriseGateway 2.1.0	<input type="checkbox"/> MXNet 1.8.0	<input type="checkbox"/> Sqoop 1.4.7
<input type="checkbox"/> Mahout 0.13.0	<input checked="" type="checkbox"/> Hue 4.10.0	<input type="checkbox"/> Phoenix 4.14.3
<input type="checkbox"/> Oozie 5.2.1	<input checked="" type="checkbox"/> Spark 2.4.8	<input type="checkbox"/> HCatalog 2.3.9
<input type="checkbox"/> TensorFlow 2.4.1		

Multiple master nodes (optional)

☐ Use multiple master nodes to improve cluster availability. [Learn more](#)

AWS Glue Data Catalog settings (optional)

☐ Use for Hive table metadata ⓘ

☐ Use for Spark table metadata ⓘ

Edit software settings ⓘ

☒ Enter configuration ☐ Load JSON from S3

Steps (optional)

A step is a unit of work you submit to the cluster. For instance, a step might contain one or more Hadoop or Spark jobs. You can also submit additional steps to a cluster after it is running. [Learn more](#)

5. Machine Selection

Instance types

<input type="radio"/> m3.2xlarge	8	30	160 SSD
<input type="radio"/> m4.large	2	8	EBS only
<input checked="" type="radio"/> m4.xlarge	4	16	EBS only
<input type="radio"/> m4.2xlarge	8	32	EBS only
<input type="radio"/> m4.4xlarge	16	64	EBS only
<input type="radio"/> m4.10xlarge	40	160	EBS only
<input type="radio"/> m4.16xlarge	64	256	EBS only
<input type="radio"/> m5.xlarge	4	16	EBS only
<input type="radio"/> m5.2xlarge	8	32	EBS only
<input type="radio"/> m5.4xlarge	16	64	EBS only
<input type="radio"/> m5.8xlarge	32	128	EBS only

Cancel
Save

6. Cluster Nodes and instances

Cluster Nodes and Instances

Choose the instance type, number of instances, and a purchasing option. [Learn more about instance purchasing options](#)

Console options for automatic scaling have changed. [Learn more](#)

Node type	Instance type	Instance count	Purchasing option
Master Master - 1	m4.xlarge 4 vCore, 16 GiB memory, EBS only storage EBS Storage: 64 GiB Add configuration settings	1 Instances	<input checked="" type="radio"/> On-demand <input type="radio"/> Spot Use on-demand as max price
Core Core - 2	m4.xlarge 4 vCore, 16 GiB memory, EBS only storage EBS Storage: 64 GiB Add configuration settings	2 Instances	<input checked="" type="radio"/> On-demand <input type="radio"/> Spot Use on-demand as max price
Task Task - 3	m5.xlarge 4 vCore, 16 GiB memory, EBS only storage EBS Storage: 32 GiB Add configuration settings	0 Instances	<input checked="" type="radio"/> On-demand <input type="radio"/> Spot Use on-demand as max price

7. Cluster - Advanced Options

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps

Step 2: Hardware

Step 3: General Cluster Settings

Step 4: Security

General Options

Cluster name upgrad_spark_streaming

☒ Logging

S3 folder s3://aws-logs-443061449916-us-east-1/elasticmap

☐ Log encryption

☒ Debugging

☐ Termination protection

8. Security Options

Create Cluster - Advanced Options

[Go to quick options](#)

[Step 1: Software and Steps](#)

[Step 2: Hardware](#)

[Step 3: General Cluster Settings](#)

Step 4: Security

Security Options

EC2 key pair **RHEL1** ⓘ

☒ Cluster visible to all IAM users in account ⓘ

Permissions ⓘ

☒ Default ☐ Custom

Use default IAM roles. If roles are not present, they will be automatically created for you with managed policies for automatic policy updates.

EMR role [EMR_DefaultRole](#) ☐ Use EMR_DefaultRole_V2 ⓘ

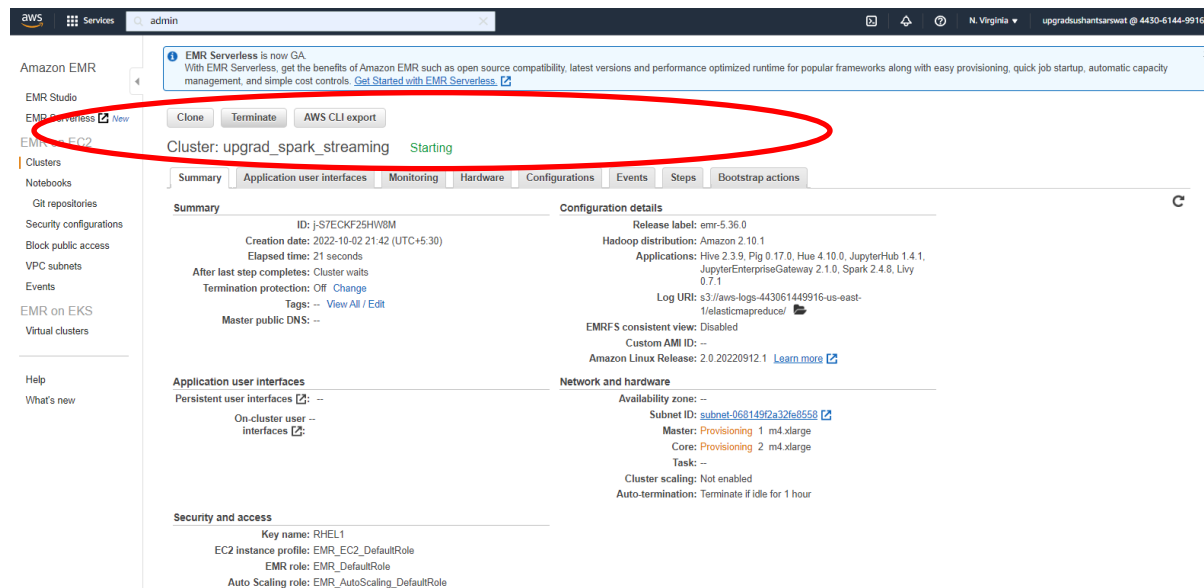
EC2 instance profile [EMR_EC2_DefaultRole](#) ⓘ

Auto Scaling role [EMR_AutoScaling_DefaultRole](#) ⓘ

► Security Configuration

► EC2 security groups

9. Starting the cluster



The screenshot shows the AWS Management Console interface for an Amazon EMR cluster. The cluster name is 'upgrad_spark_streaming' and its status is 'Starting'. The cluster is highlighted with a red oval. The console displays various tabs for the cluster, including Summary, Application user interfaces, Monitoring, Hardware, Configurations, Events, Steps, and Bootstrap actions. The Summary tab is selected, showing details such as ID, Creation date, Elapsed time, After last step completes, Termination protection, Tags, Master public DNS, Release label, Hadoop distribution, Applications, Log URI, EMRFS consistent view, Custom AMI ID, Amazon Linux Release, Network and hardware, Availability zone, Subnet ID, Master, Core, Task, Cluster scaling, and Auto-termination. The Security and access section shows the Key name (RHEL1), EC2 instance profile (EMR_EC2_DefaultRole), EMR role (EMR_DefaultRole), and Auto Scaling role (EMR_AutoScaling_DefaultRole).

10. Cluster Ready

Amazon EMR

EMR Studio

EMR Serverless **New**

EMR on EC2

Clusters

Notebooks

Git repositories

Security configurations

Block public access

VPC subnets

Events

EMR on EKS

Virtual clusters

Help

What's new

Cluster: **upgrad_spark_streaming** **Waiting** Cluster ready after last step completed.

Summary Application user interfaces Monitoring Hardware Configurations Events Steps Bootstrap actions

Summary

ID: j-S7ECKF25H4W8M
 Creation date: 2022-10-02 21:42 (UTC+5:30)
 Elapsed time: 18 minutes
 After last step completes: Cluster waits
 Termination protection: Off [Change](#)
 Tags: -- [View All](#) / [Edit](#)
 Master public DNS: ec2-3-81-142-164.compute-1.amazonaws.com [Connect to the Master Node Using SSH](#)

Configuration details

Release label: emr-5.36.0
 Hadoop distribution: Amazon 2.10.1
 Applications: Hive 2.3.9, Pig 0.17.0, Hue 4.10.0, JupyterHub 1.4.1, JupyterEnterpriseGateway 2.1.0, Spark 2.4.8, Livy 0.7.1
 Log URI: s3://aws-logs-443061449916-us-east-1/elasticmapreduce/
 EMRFS consistent view: Disabled
 Custom AMI ID: --
 Amazon Linux Release: 2.0 20220912.1 [Learn more](#)

Application user interfaces

Persistent user interfaces: [Spark history server](#), [YARN timeline server](#), [Tez UI](#)
 On-cluster user interfaces: Not Enabled [Enable an SSH Connection](#)

Network and hardware

Availability zone: us-east-1a
 Subnet ID: [subnet-0681492a32e8558](#)
 Master: [Bootstrapping](#) 1 m4.xlarge
 Core: [Provisioning](#) 2 m4.xlarge
 Task: --
 Cluster scaling: Not enabled
 Auto-termination: Terminate if idle for 1 hour

Security and access

Key name: RHEL1
 EC2 instance profile: EMR_EC2_DefaultRole
 EMR role: EMR_DefaultRole
 Auto Scaling role: EMR_AutoScaling_DefaultRole

11. Creating Notebook

Amazon EMR

EMR Studio

EMR Serverless **New**

EMR on EC2

Clusters

Notebooks

Git repositories

Security configurations

Block public access

VPC subnets

Events

EMR on EKS

Virtual clusters

Help

What's new

Notebooks

Use EMR notebooks based on Jupyter to analyze data interactively with live code, narrative text, visualizations, and more. Create and attach notebooks to Amazon EMR clusters running Hadoop, Spark, and Livy. Notebooks run free of charge and are saved in Amazon S3 independently of clusters. Standard billing for clusters and Amazon S3 apply. [Learn more](#)

[Create notebook](#) [View details](#) [Open in JupyterLab](#) [Open in Jupyter](#) [Start](#) [Stop](#) [Delete](#)

Filter: **All notebooks** 4 notebooks (all loaded)

Name	Status	Cluster	Creation time (UTC+5:30)	Last modified
Spark_ML	Stopped	j-PPCQ960H57ZN	2022-09-10 20:56 (UTC+5:30)	3 weeks ago
upgrad_etl_notebook	Stopped	j-3ANQAM48T593	2022-07-28 09:57 (UTC+5:30)	9 weeks ago
upgrad_etl_notebook	Stopped	j-36VS1HR7IY861	2022-07-25 23:07 (UTC+5:30)	9 weeks ago
upgrad_spark	Stopped	j-3FJF3DMFUAB8	2022-07-19 21:24 (UTC+5:30)	10 weeks ago

12. Configuring Notebook

Amazon EMR

EMR Studio

EMR Serverless [New](#)

EMR on EC2

Clusters

Notebooks

Git repositories

Security configurations

Block public access

VPC subnets

Events

EMR on EKS

Virtual clusters

Help

What's new

EMR Serverless is now GA.
With EMR Serverless, get the benefits of Amazon EMR such as open source compatibility, latest versions and performance optimized runtime for popular frameworks along with easy provisioning, quick job startup, automatic capacity management, and simple cost controls. [Get Started with EMR Serverless](#)

Create notebook

Name and configure your notebook

Name your notebook, choose a cluster or create one, and customize configuration options if desired. [Learn more](#)

Notebook name*
Names may only contain alphanumeric characters, hyphens (-), or underscores (_).

Description
255 characters max.

Cluster* ☒ Choose an existing cluster

☐ Create a cluster

Security groups ☒ Use default security groups ☐ Choose security groups

AWS service role*

Notebook location* Choose an S3 location where files for this notebook are saved.
☒ Use the default S3 location
s3://aws-emr-resources-443061449916-us-east-1/notebooks/
☐ Choose an existing S3 location in us-east-1

Git repository

Tags


13. Connecting with existing cluster

Choose a cluster

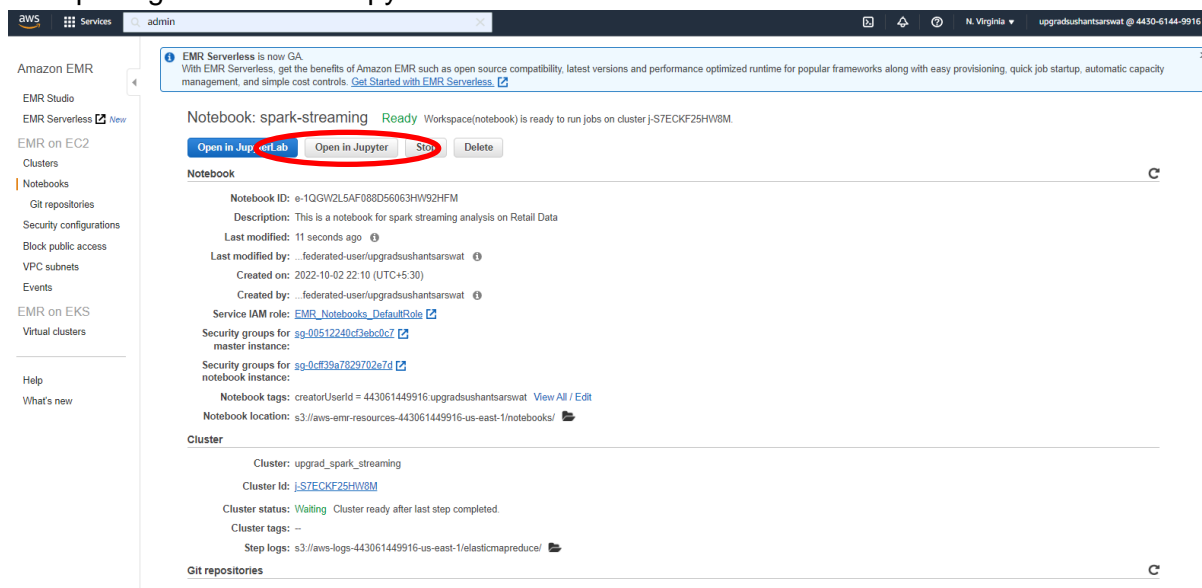
The listed clusters meet notebook requirements. They are in an EC2-VPC, running EMR 5.18.0 or later, and have Hadoop, Spark, and Livy installed. [Learn more](#)

The notebook can be opened once the cluster is in Waiting or Running status.

Filter: 1 cluster (all loaded)

Name	ID	Status
 upgrad_spark_streaming	j-S7ECKF25HW8M	Waiting Cluster ready

14. Opening Notebook in Jupyter



Amazon EMR console showing the 'Notebook: spark-streaming' page. The 'Open in Jupyter' button is circled in red.

EMR Serverless is now GA. With EMR Serverless, get the benefits of Amazon EMR such as open source compatibility, latest versions and performance optimized runtime for popular frameworks along with easy provisioning, quick job startup, automatic capacity management, and simple cost controls. [Get Started with EMR Serverless](#)

Notebook: spark-streaming Ready Workspace(notebook) is ready to run jobs on cluster j-S7ECKF25HW8M.

Buttons: Open in JupyterLab, Open in Jupyter (circled in red), Stop, Delete

Notebook details:

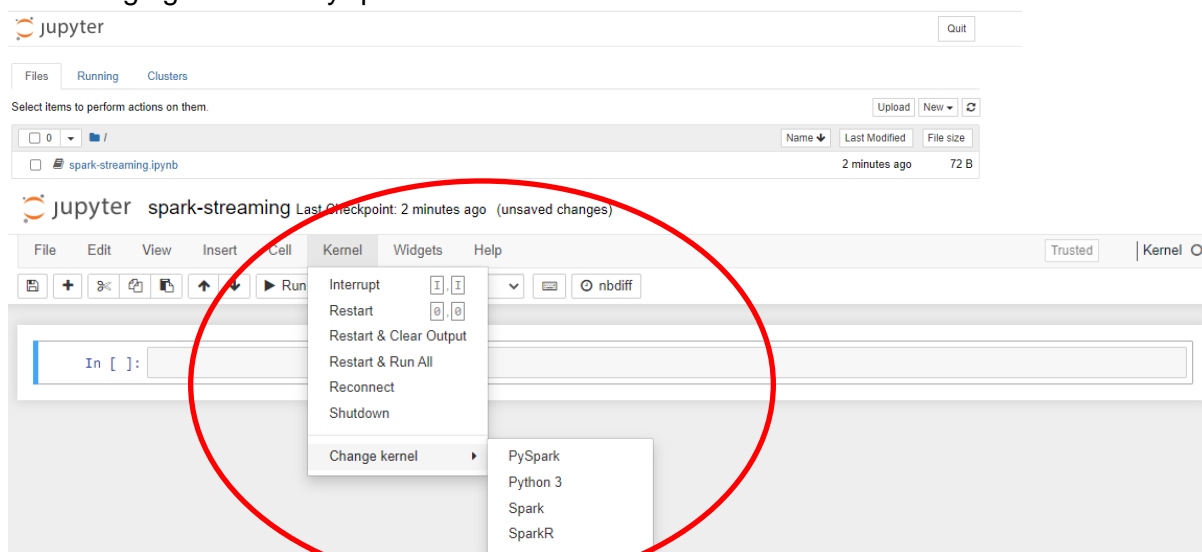
- Notebook ID: e-1QGIV2L5AF088D56063HV92HFM
- Description: This is a notebook for spark streaming analysis on Retail Data
- Last modified: 11 seconds ago
- Last modified by: ...federated-user/upgradushantsarwat
- Created on: 2022-10-02 22:10 (UTC+5:30)
- Created by: ...federated-user/upgradushantsarwat
- Service IAM role: EMR_Notebooks_DefaultRole
- Security groups for master instance: sg-00512240c3ebc0c7
- Security groups for notebook instance: sg-0cf39a7829702e7d
- Notebook tags: creator/UserId = 443061449916 upgradushantsarwat
- Notebook location: s3://aws-emr-resources-443061449916-us-east-1/notebooks/

Cluster details:

- Cluster: upgrad_spark_streaming
- Cluster Id: j-S7ECKF25HW8M
- Cluster status: Waiting Cluster ready after last step completed
- Cluster tags: --
- Step logs: s3://aws-logs-443061449916-us-east-1/elasticsearchpreduce/

Git repositories

15. Changing kernel to PySpark



JupyterLab interface showing the 'Kernel' menu with 'Change kernel' option selected, and the 'PySpark' option highlighted in the submenu.

Files: Running Clusters

Select items to perform actions on them.

Files list:

Name	Last Modified	File size
spark-streaming.ipynb	2 minutes ago	72 B

JupyterLab spark-streaming Last Checkpoint: 2 minutes ago (unsaved changes)

Kernel menu options:

- Interrupt
- Restart
- Restart & Clear Output
- Restart & Run All
- Reconnect
- Shutdown
- Change kernel (selected)
 - PySpark (highlighted)
 - Python 3
 - Spark
 - SparkR

16. Code explanation:

- (a) **Step 1:** All the required libraries were imported from PySpark.
- (b) **Step 2:** The Spark session was established.
- (c) **Step 3:** Connection to provided Kafka Server and Topic was established using given IP address and port and data was read into 'lines'.
- (d) **Step 4:** Schema is defined to make the raw data more readable.
- (e) **Step 5:** Four utility functions are created for 'Order', 'Return', 'Calculation of Total no. of Items' and 'Cost of all products in Invoice'.
- (f) **Step 6:** User Defined Functions (UDFs) are created with utility functions created above.
- (g) **Step 7:** Order Query Data is displayed by providing output to Console.
- (h) **Step 8:** Calculation of Time Based KPIs is undertaken and alias are provided.
- (i) **Step 10:** Calculation of Time and Country Based KPIs is undertaken and alias are provided.
- (j) **Step 11:** Time based KPI values are appended in a JSON file to HDFS in path location 'timeKPI' at an interval of one minute.
- (k) **Step 12:** Time and Country based KPI values are appended in a JSON file to HDFS in path location 'time_countryKPI' at an interval of one minute.

17. On completion of the code, we open the Terminal window and login as Hadoop user.

```
[hadoop@ip-172-31-42-115 ~]$ vi spark-streaming.py  
[hadoop@ip-172-31-42-115 ~]$ spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.8 spark-streaming.py 18.211.252.152 9092 real-time-project
```

- (a) We can use the vi editor to access the spark-streaming.py file for any changes to the code.
- (b) A spark-submit job is submitted using the following code:


```
spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.8 spark-streaming.py 18.211.252.152 9092 real-time-project
```

20. We can witness the following display on the Console:

```
Batch: 0
-----
+-----+-----+-----+-----+-----+-----+-----+-----+
|invoice_no|country|timestamp|type|total_items|total_cost|is_order|is_return|
+-----+-----+-----+-----+-----+-----+-----+-----+

Batch: 1
-----
+-----+-----+-----+-----+-----+-----+-----+-----+
|invoice_no|country|timestamp|type|total_items|total_cost|is_order|is_return|
+-----+-----+-----+-----+-----+-----+-----+-----+
|154132551513585|United Kingdom|2022-10-05 20:16:28|ORDER|24|59.04|1|0|
|154132551513586|United Kingdom|2022-10-05 20:16:30|ORDER|14|128.04|1|0|
|154132551513587|United Kingdom|2022-10-05 20:16:38|ORDER|74|194.09999|1|0|
|154132551513588|United Kingdom|2022-10-05 20:16:41|ORDER|18|30.3|1|0|
|154132551513589|United Kingdom|2022-10-05 20:16:46|ORDER|35|130.25|1|0|
|154132551513590|United Kingdom|2022-10-05 20:16:46|ORDER|4|10.58|1|0|
|154132551513591|Portugal|2022-10-05 20:16:50|ORDER|29|13.9|1|0|
|154132551513592|United Kingdom|2022-10-05 20:17:06|ORDER|3|6.24|1|0|
|154132551513593|United Kingdom|2022-10-05 20:17:16|ORDER|19|66.270004|1|0|
|154132551513594|United Kingdom|2022-10-05 20:17:23|ORDER|2|2.9|1|0|
+-----+-----+-----+-----+-----+-----+-----+-----+

Batch: 2
-----
+-----+-----+-----+-----+-----+-----+-----+-----+
|invoice_no|country|timestamp|type|total_items|total_cost|is_order|is_return|
+-----+-----+-----+-----+-----+-----+-----+-----+
|154132551513595|United Kingdom|2022-10-05 20:17:28|ORDER|7|38.49|1|0|
|154132551513596|France|2022-10-05 20:17:28|ORDER|125|142.99|1|0|
|154132551513597|United Kingdom|2022-10-05 20:17:40|ORDER|3|4.59|1|0|
|154132551513598|United Kingdom|2022-10-05 20:17:40|ORDER|4|35.82|1|0|
|154132551513599|United Kingdom|2022-10-05 20:17:41|ORDER|38|188.6|1|0|
|154132551513600|United Kingdom|2022-10-05 20:17:47|ORDER|28|58.18|1|0|
|154132551513601|United Kingdom|2022-10-05 20:17:48|ORDER|12|15.0|1|0|
|154132551513602|United Kingdom|2022-10-05 20:17:49|ORDER|12|15.0|1|0|
|154132551513603|United Kingdom|2022-10-05 20:17:52|ORDER|1|8.5|1|0|
|154132551513604|United Kingdom|2022-10-05 20:17:53|ORDER|23|50.010002|1|0|
|154132551513605|United Kingdom|2022-10-05 20:17:55|ORDER|11|33.35|1|0|
|154132551513606|United Kingdom|2022-10-05 20:18:08|ORDER|8|32.98|1|0|
|154132551513607|United Kingdom|2022-10-05 20:18:10|ORDER|33|47.05|1|0|
|154132551513608|United Kingdom|2022-10-05 20:18:11|ORDER|14|25.33|1|0|
|154132551513609|United Kingdom|2022-10-05 20:18:12|ORDER|3|17.4|1|0|
|154132551513610|United Kingdom|2022-10-05 20:18:17|ORDER|2|19.92|1|0|
+-----+-----+-----+-----+-----+-----+-----+-----+
```

21. Press Ctrl+Z to exit the scrolling display.

22. On inputting `hadoop fs-ls` we will observed the following list

```
[hadoop@ip-172-31-42-115 ~]$ hadoop fs -ls
Found 3 items
drwxr-xr-x - hadoop hdfsadmingroup 0 2022-10-05 20:21 .sparkStaging
drwxr-xr-x - hadoop hdfsadmingroup 0 2022-10-05 20:28 timeKPI
drwxr-xr-x - hadoop hdfsadmingroup 0 2022-10-05 20:28 time_countryKPI
```

23. On inputting `hadoop fs -ls timeKPI`, we will find the JSON files stored in the location.

```
[hadoop@ip-172-31-42-115 ~]$ hadoop fs -ls timeKPI
Found 13 items
drwxr-xr-x - hadoop hdfsadmingroup 0 2022-10-05 20:28 timeKPI/_spark_
metadata
drwxr-xr-x - hadoop hdfsadmingroup 0 2022-10-05 20:22 timeKPI/cp
-rw-r--r-- 1 hadoop hdfsadmingroup 0 2022-10-05 20:27 timeKPI/part-00
000-105ee2ae-421f-492c-ac6f-b0394a419c23-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 0 2022-10-05 20:28 timeKPI/part-00
000-3eca9c99-a925-44e1-b909-bd011521bf1a-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 0 2022-10-05 20:26 timeKPI/part-00
000-506f8dcf-23f3-44b9-9468-1f0b525bab3d-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 0 2022-10-05 20:25 timeKPI/part-00
000-7fab06d5-2efa-4534-a11f-a42801d7a2a9-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 0 2022-10-05 20:22 timeKPI/part-00
000-8cb61551-9391-4029-b521-01566113c6ca-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 0 2022-10-05 20:24 timeKPI/part-00
000-c85571b5-8e43-4801-bc83-3d33d157982b-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 0 2022-10-05 20:23 timeKPI/part-00
000-d12f1fa7-923f-4d41-b42d-f68358d9aaal-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 178 2022-10-05 20:26 timeKPI/part-00
072-38e009bb-a85f-41da-a881-88dcc70da680-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 178 2022-10-05 20:25 timeKPI/part-00
115-0773f031-cd27-4965-9029-f968b8adc856-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 178 2022-10-05 20:27 timeKPI/part-00
169-18c0cde0-7b0a-441c-92d2-ae5bd88e78cc-c000.json
-rw-r--r-- 1 hadoop hdfsadmingroup 182 2022-10-05 20:28 timeKPI/part-00
181-af1a3eef-d7d0-45d9-b5e0-6ef769906201-c000.json
```

24. On inputting `hadoop fs -cat timeKPI/<file name>`, we can observe the KPI values.

```
[hadoop@ip-172-31-42-115 ~]$ hadoop fs -cat timeKPI/part-00072-38e009bb-a85f-41da-a881-88dcc70da680-c000.json
{"start":"2022-10-05T20:17:00.000Z","end":"2022-10-05T20:18:00.000Z","total_volume_of_sales":665.9400181770325,"average_transaction_size":47.56714415550232,"rate_of_return":0.0}
```

25. The same commands as mentioned in Para 23 and 24 modified for time_countryKPI will result in the following output.

```
[hadoop@ip-172-31-42-115 ~]$ hadoop fs -cat time_countryKPI/part-00020-e873fd0c-3dd4-4502-868f-b8f7c672741f-c000.json
{"start":"2022-10-05T20:19:00.000Z","end":"2022-10-05T20:20:00.000Z","country":"United Kingdom","OEM":8,"total_volume_of_sales":1355.3499084711075,"rate_of_return":0.125}
```

26. Thereafter, on execution of following code we will be able to get the summarised output table to Console-output File:

```
spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.8 spark-streaming.py 18.211.252.152 9092 real-time-project > Console-output
```

27. WinSCP was used thereafter to extract the JSON files of timeKPI and time_countryKPI to respective folders as well as Console-output file.