Contents

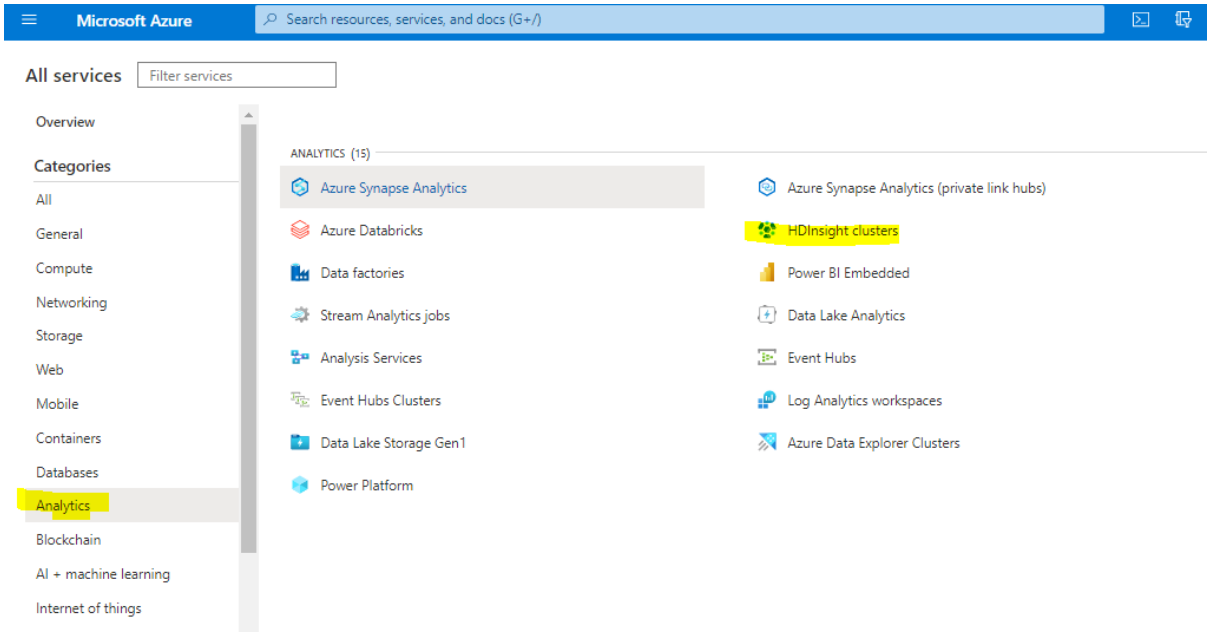
What is Apache Spark?

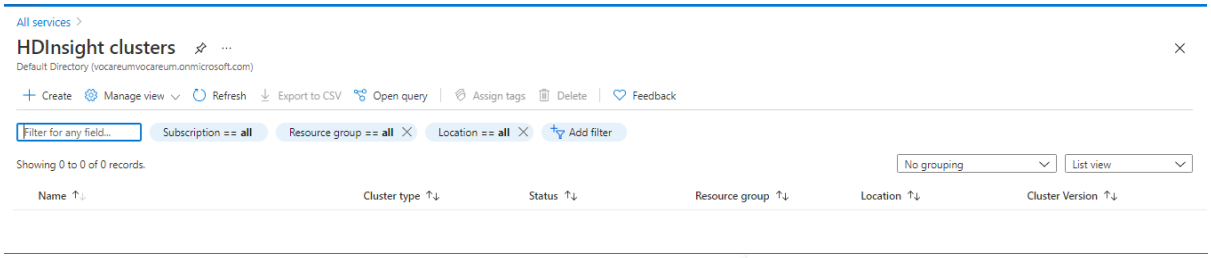
Apache Spark is a parallel processing framework that supports in-memory processing to boost the performance of big-data analytic applications. Apache Spark in Azure HDInsight is the Microsoft implementation of Apache Spark in the cloud.

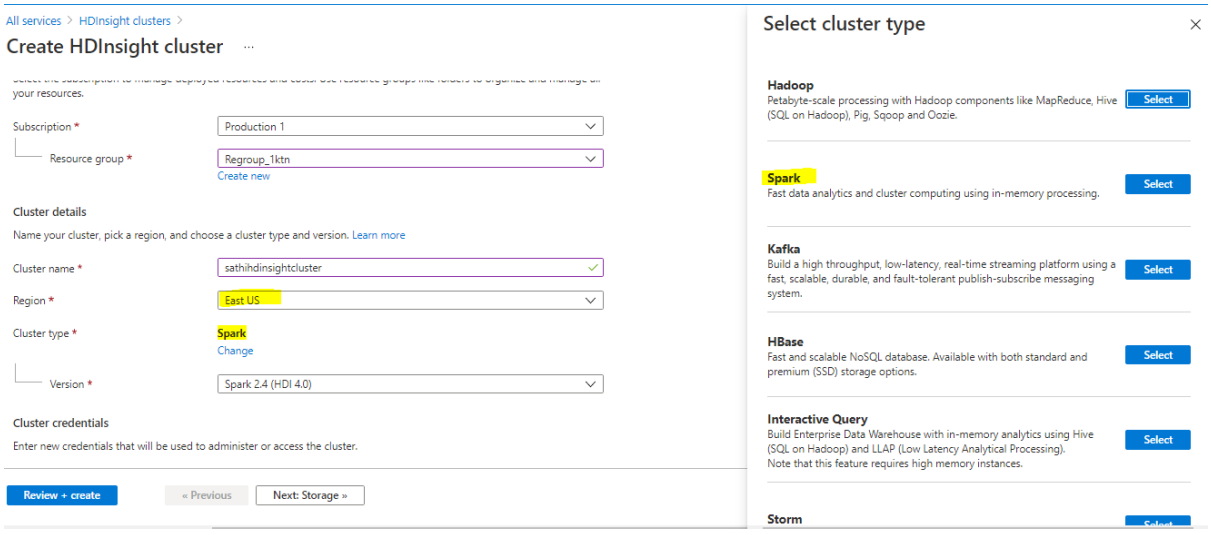
you can use HDInsight Spark clusters to process your data stored in Azure(Azure Blob storage, Azure Data Lake Storage Gen1, or Azure Data Lake Storage Gen2)

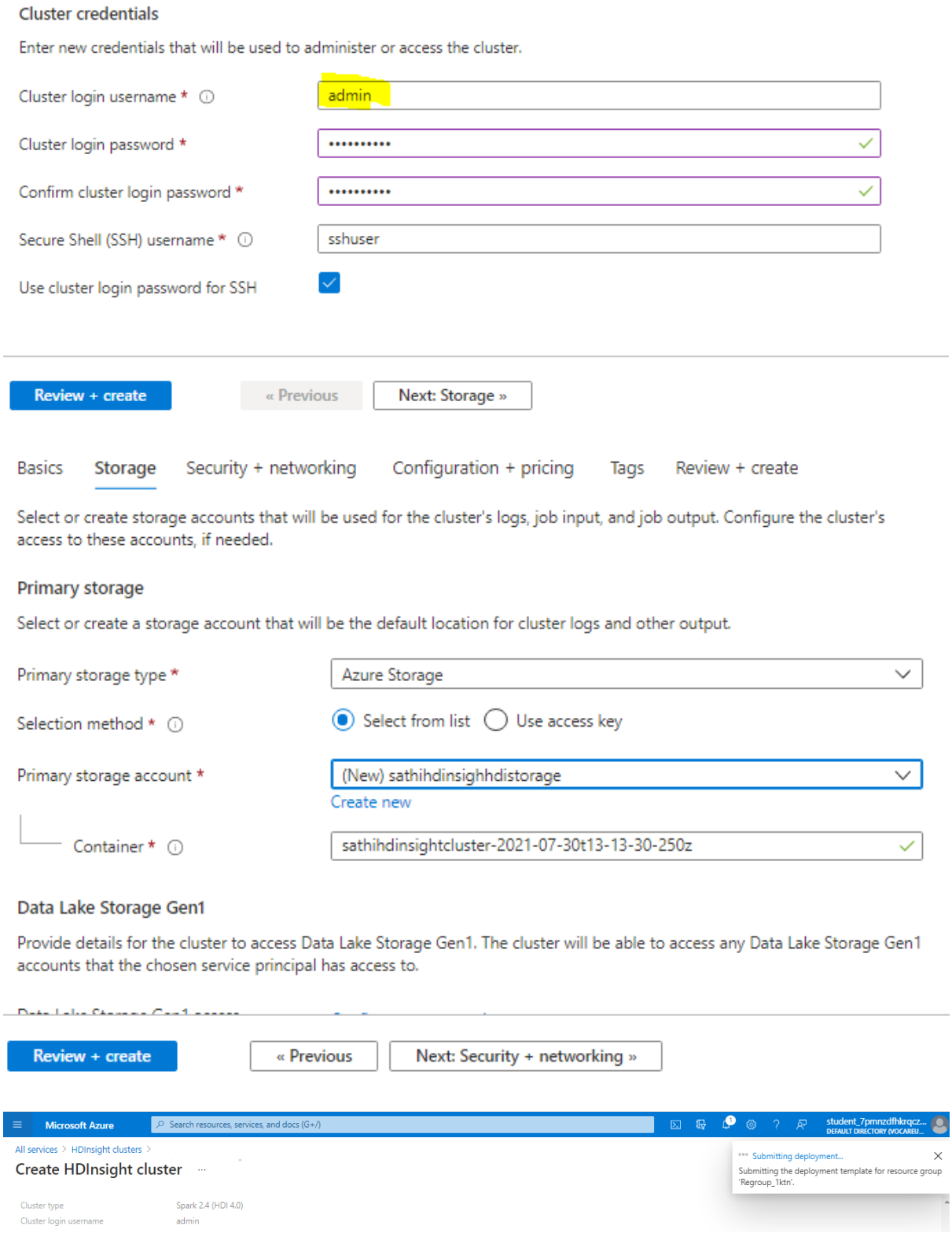
1) Create an Apache Spark Cluster in HDInsight in the East US region

Create an HDInsight cluster to process massive amounts of data using popular open-source frameworks such as Hadoop, Spark, Hive, LLAP, Kafka, Storm, ML Services, and more.

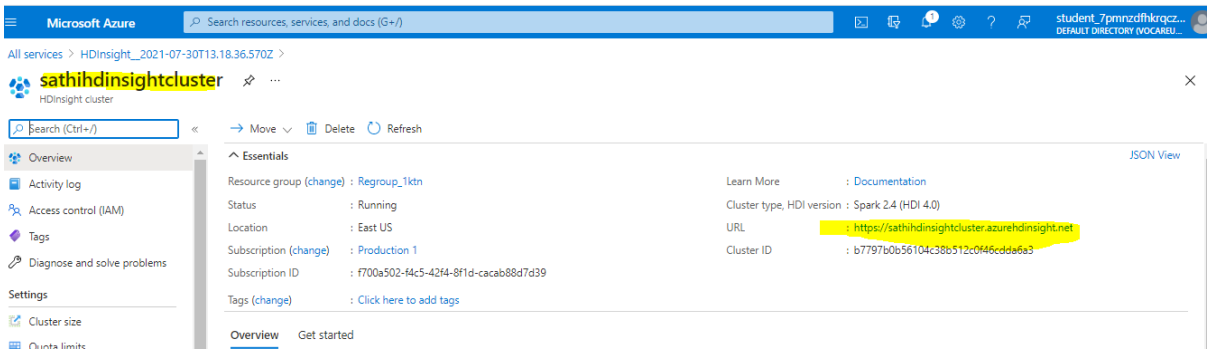

Microsoft Azure
Search resources, services, and docs (G+/)
All services
Filter services
Overview
Categories
All
General
Compute
Networking
Storage
Web
Mobile
Containers
Databases
Analytics
Blockchain
AI + machine learning
Internet of things
ANALYTICS (15)
Azure Synapse Analytics
Azure Databricks
Data factories
Stream Analytics jobs
Analysis Services
Event Hubs Clusters
Data Lake Storage Gen1
Power Platform
Azure Synapse Analytics (private link hubs)
HDInsight clusters
Power BI Embedded
Data Lake Analytics
Event Hubs
Log Analytics workspaces
Azure Data Explorer Clusters


All services
HDInsight clusters
Default Directory (vocareumvocareum.onmicrosoft.com)
Create
Manage view
Refresh
Export to CSV
Open query
Assign tags
Delete
Feedback
Filter for any field...
Subscription == all
Resource group == all
Location == all
Add filter
Showing 0 to 0 of 0 records.
No grouping
List view
Name
Cluster type
Status
Resource group
Location
Cluster Version


All services
HDInsight clusters
Create HDInsight cluster
Subscription *
Production 1
Resource group *
Regroup_1ktn
Create new
Cluster details
Name your cluster, pick a region, and choose a cluster type and version. Learn more
Cluster name *
sathihdinsightcluster
Region *
East US
Cluster type *
Spark
Change
Version *
Spark 2.4 (HDI 4.0)
Cluster credentials
Enter new credentials that will be used to administer or access the cluster.
Review + create
« Previous
Next: Storage »
Select cluster type
Hadoop
Petabyte-scale processing with Hadoop components like MapReduce, Hive (SQL on Hadoop), Pig, Sqoop and Oozie.
Select
Spark
Fast data analytics and cluster computing using in-memory processing.
Select
Kafka
Build a high throughput, low-latency, real-time streaming platform using a fast, scalable, durable, and fault-tolerant publish-subscribe messaging system.
Select
HBase
Fast and scalable NoSQL database. Available with both standard and premium (SSD) storage options.
Select
Interactive Query
Build Enterprise Data Warehouse with in-memory analytics using Hive (SQL on Hadoop) and LLAP (Low Latency Analytical Processing).
Note that this feature requires high memory instances.
Select
Storm

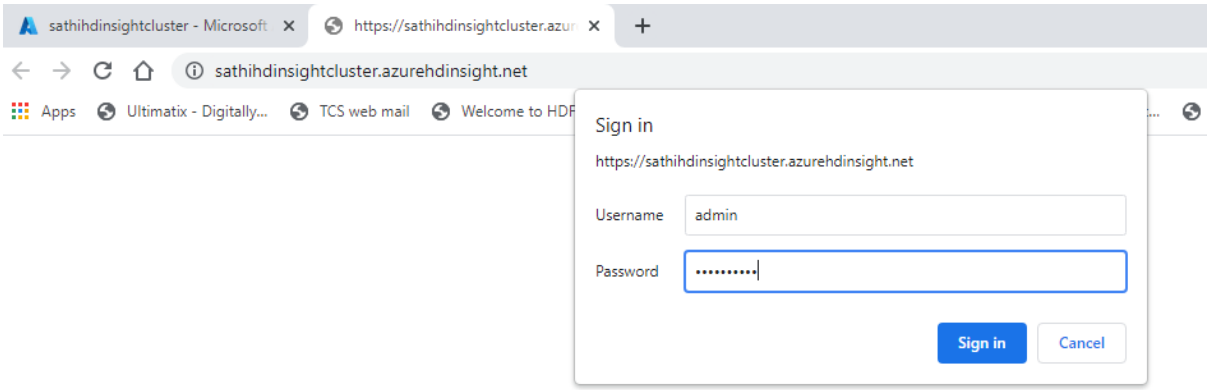


It takes about 20 minutes to create the cluster. The cluster must be created before you can proceed to the next session.

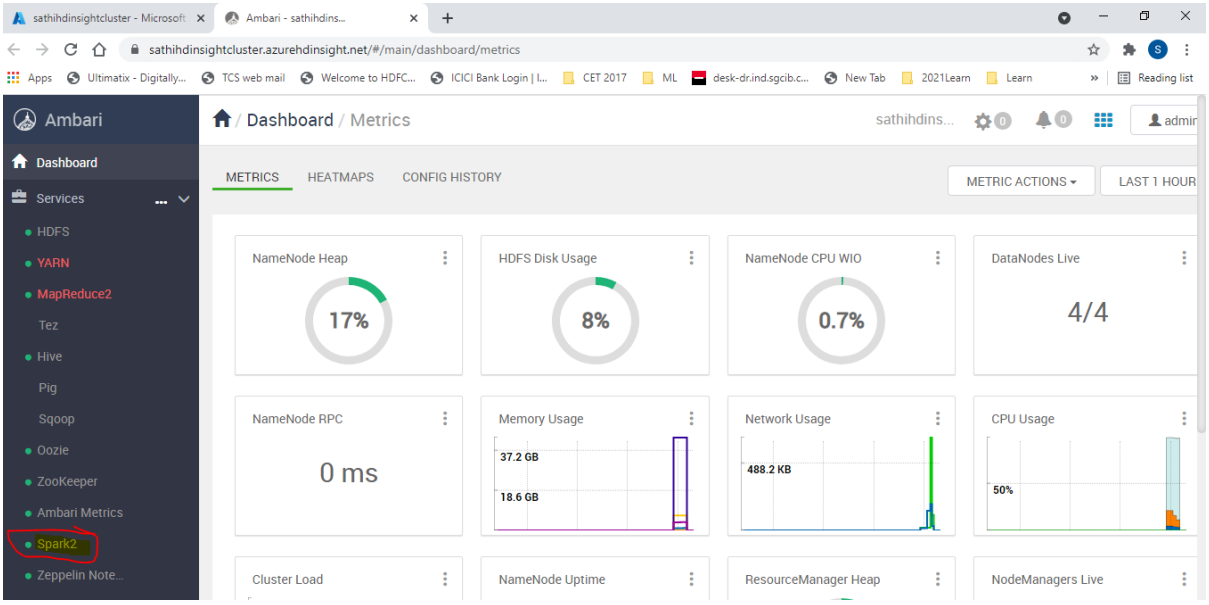


2) Create a Jupyter Notebook in the Spark cluster

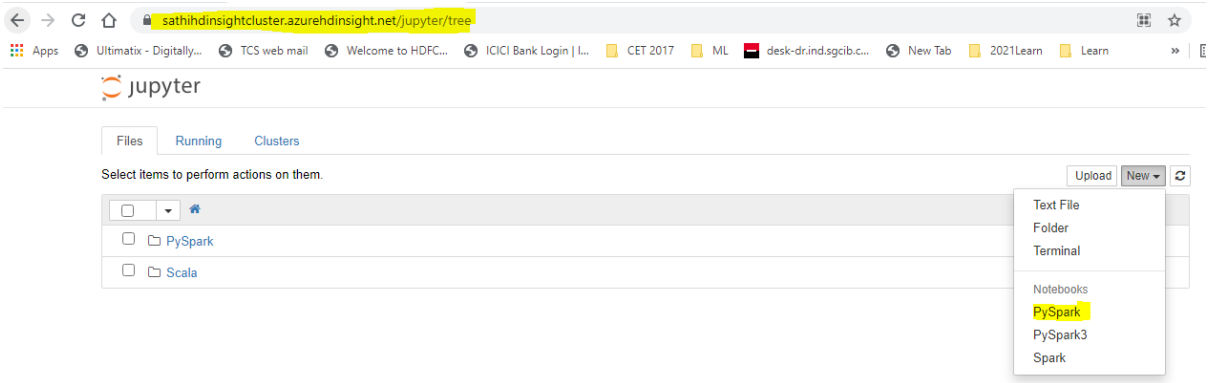
For Jupyter notebook web browser click on sathihdinsightcluster URL or navigate to `https://CLUSTERNAME.azurehdinsight.net/jupyter`, where `CLUSTERNAME` is the name of your cluster. If prompted, enter the cluster login credentials for the cluster

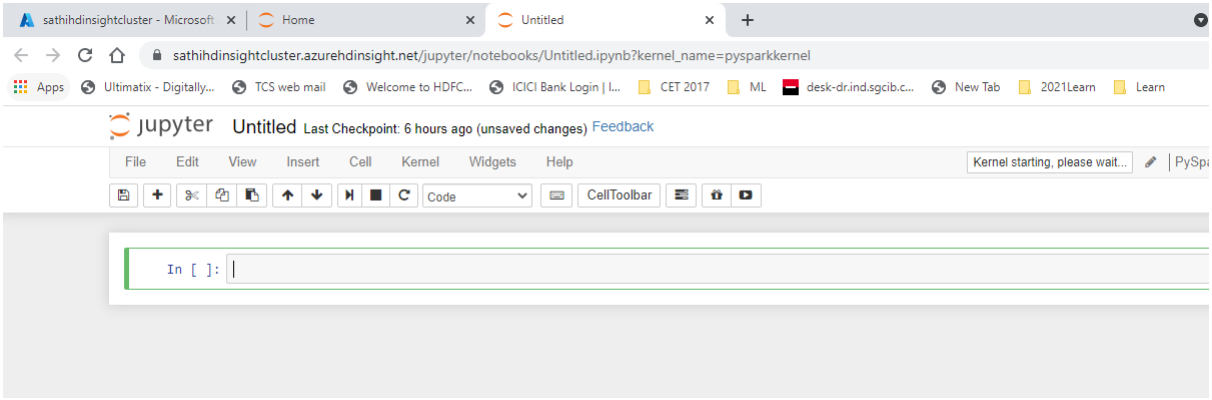


Ambari View

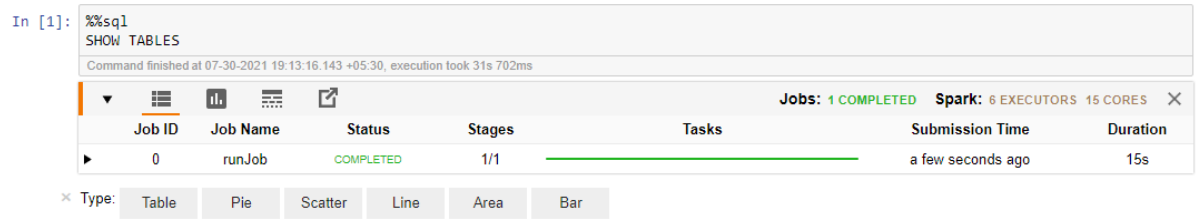


https://sathihdinsightcluster.azurehdinsight.net/jupyter





Run Apache Spark SQL statements and verify



database	tableName	isTemporary
default	hivesampletable	False

```
Starting Spark application
```

ID	YARN Application ID	Kind	State	Spark UI	Driver log	Current session?
0	application_1627651608052_0004	pyspark	idle	Link	Link	✓

3) Import the following dataset into the Jupyter notebook using Spark SQL

https://github.com/MicrosoftLearning/20775_Performing-Data-Engineering-on-Microsoft-HDInsight/blob/master/Allfiles/Demofiles/Mod04/clidata/hvac/HVAC.csv

```
from pyspark.sql import *
from pyspark.sql.types import *
```

```
In [1]: from pyspark.sql import *
        from pyspark.sql.types import *
Command finished at 07-30-2021 20:07:12.563 +05:30, execution took 45ms

Starting Spark application
```

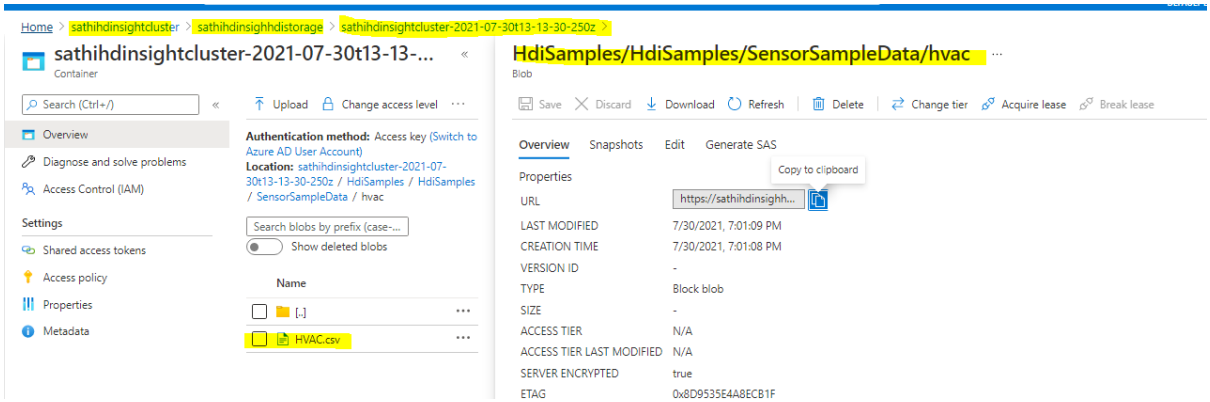
ID	YARN Application ID	Kind	State	Spark UI	Driver log	Current session?
1	application_1627651608052_0005	pyspark	idle	Link	Link	✓

```
SparkSession available as 'spark'.
/usr/bin/anaconda/lib/python2.7/site-packages/matplotlib/font_manager.py:273: UserWarning: Matplotlib is building the font cach
e using fc-list. This may take a moment.
  warnings.warn('Matplotlib is building the font cache using fc-list. This may take a moment.')

In [2]: csvfile = spark.read.csv("/HdiSamples/HdiSamples/SensorSampleData/HVAC.csv")
        csvfile.write.saveAsTable("hvac")
Command finished at 07-30-2021 20:07:47.780 +05:30, execution took 3s 286ms

u'Path does not exist: wasb://sathihdinsightcluster-2021-07-30t13-13-30-250z@sathihdinsighhdistorage.blob.core.windows.net/HdiS
amples/HdiSamples/SensorSampleData/HVAC.csv;'
Traceback (most recent call last):
  File "/usr/hdp/current/spark2-client/python/pyspark/sql/readwriter.py", line 476, in csv
    return self._df(self._jreader.csv(self._spark._sc._jvm.PythonUtils.toSeq(path)))
  File "/usr/hdp/current/spark2-client/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
    answer, self.gateway_client, self.target_id, self.name)
  File "/usr/hdp/current/spark2-client/python/pyspark/sql/utils.py", line 69, in deco
```

Navigate to sample document stored in azurehdinsightcluster storage



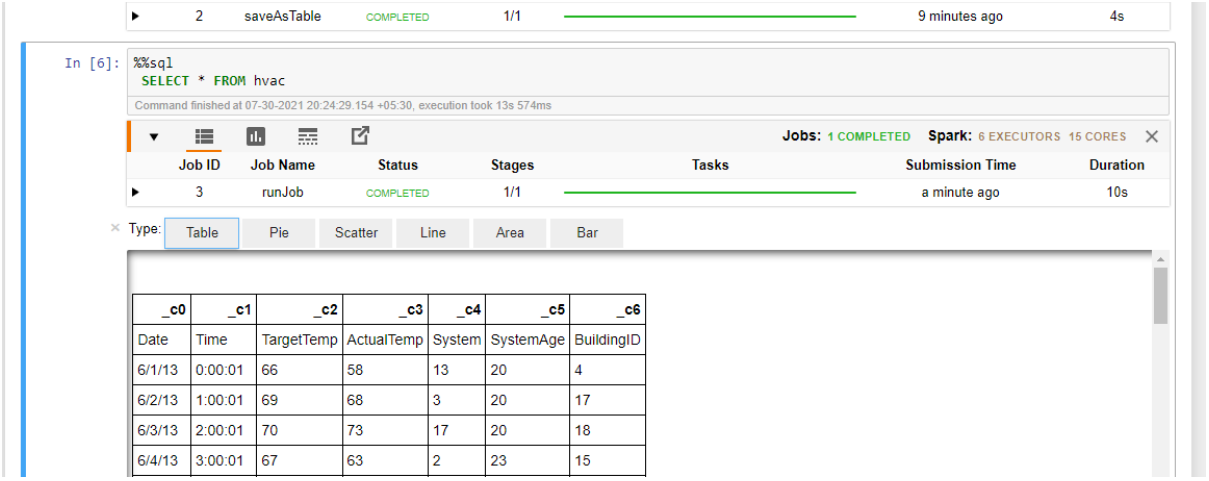
csvfile = spark.read.csv("/HdiSamples/HdiSamples/SensorSampleData/hvac/HVAC.csv")
ccsvfile.write.saveAsTable("hvac")

```
In [4]: csvfile = spark.read.csv("/HdiSamples/HdiSamples/SensorSampleData/hvac/HVAC.csv")
        csvfile.write.saveAsTable("hvac")
Command finished at 07-30-2021 20:16:31.967 +05:30, execution took 21s 374ms
```

Jobs: 2 COMPLETED Spark: 6 EXECUTORS 15 CORES

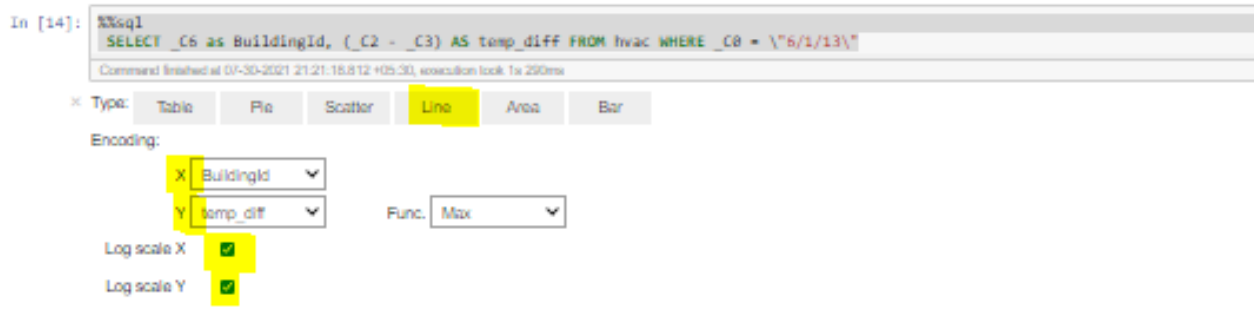
Job ID	Job Name	Status	Stages	Tasks	Submission Time	Duration
1	csv	COMPLETED	1/1		a minute ago	2s
2	saveAsTable	COMPLETED	1/1		a few seconds ago	4s

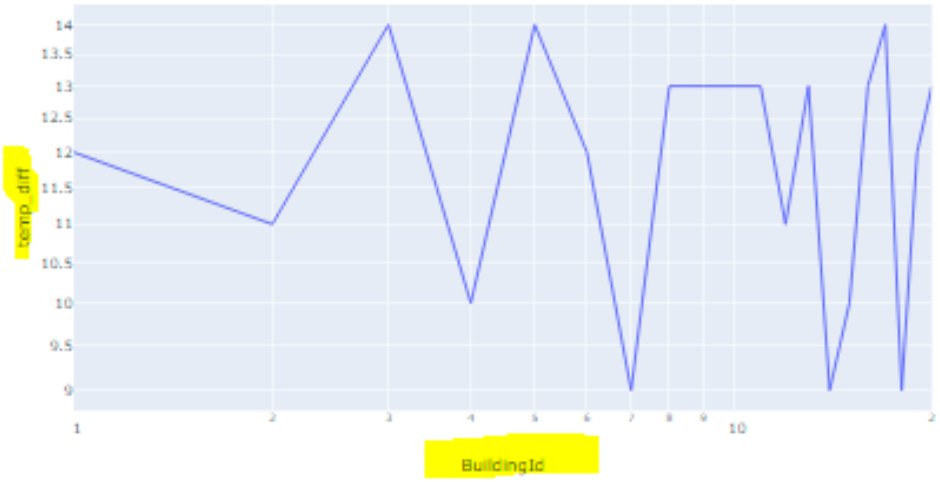
%%sql
SELECT * FROM hvac



%%sql

SELECT _C0 as date, (_C2 - _C3) AS temp_diff FROM hvac WHERE _C0 = \"6/1/13\"





4) References :

https://docs.microsoft.com/en-us/azure/hdinsight/spark/apache-spark-overview

https://docs.microsoft.com/en-us/azure/hdinsight/spark/apache-spark-jupyter-spark-sql-use-portal

https://docs.microsoft.com/en-us/azure/hdinsight/spark/apache-spark-use-with-data-lake-store

https://stackoverflow.com/questions/68508375/azure-hdinsight-sparksql-how-to-load-csv-file-from-github-in-dataframe

https://docs.microsoft.com/en-us/azure/hdinsight/spark/apache-spark-load-data-run-query

https://docs.microsoft.com/en-us/azure/hdinsight/spark/apache-spark-load-data-run-query

https://www.youtube.com/watch?v=pocW4ZHP_ng

5) Dowload Azure StoreExplorer

https://azure.microsoft.com/en-in/features/storage-explorer/#overview