

Azure Databricks

AI Discovery Day Workshop

Hope Foley
Data/AI Technical Specialist

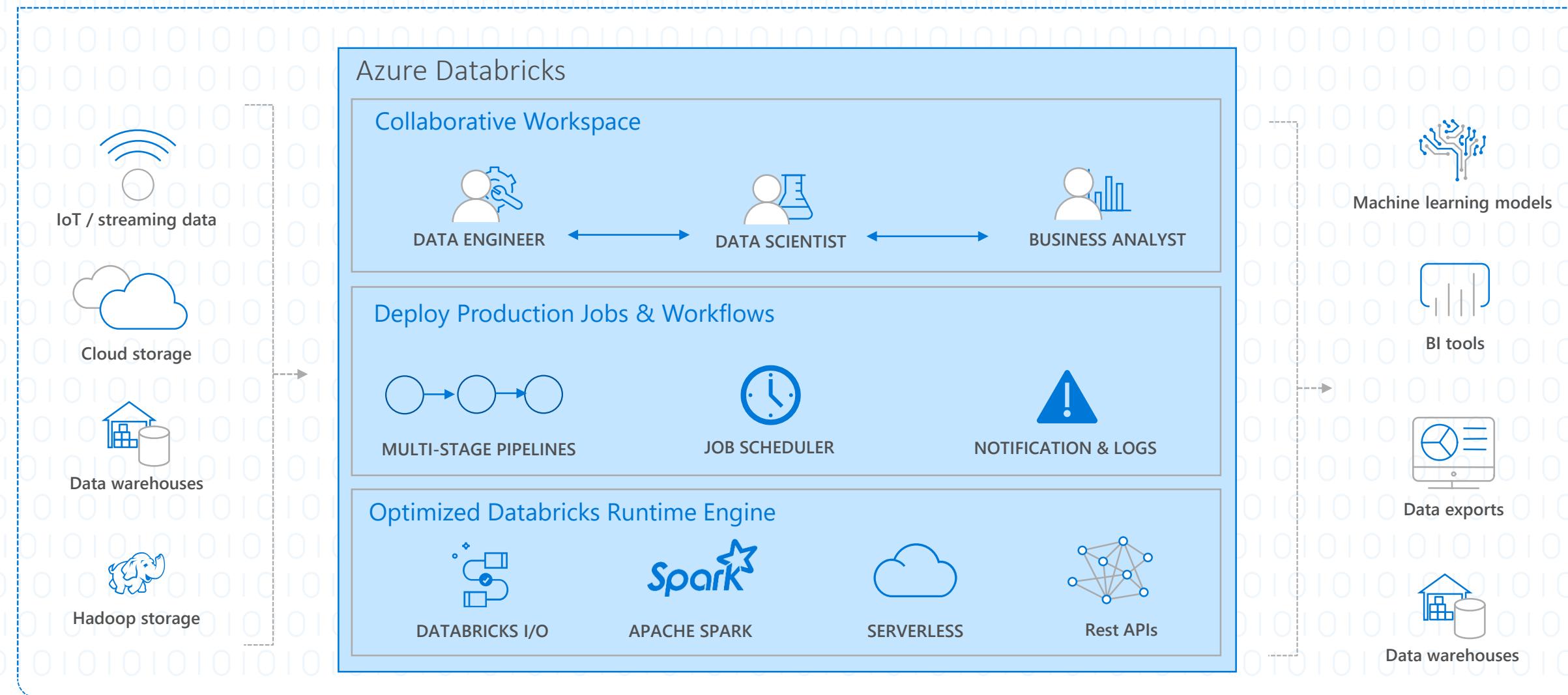
October 23, 2018



Topics to Cover

- Term review
- Azure Databricks – what is it?
- Cluster Creation
- Working with Databricks
 - Setup storage (security/AAD)
 - Working with SQL
 - Mix it up
- Where to go from here
 - Data Analytics, Stream Analytics, Machine Learning, Graph Processing

AZURE DATABRICKS



Enhance Productivity

Build on secure & trusted cloud

Scale without limits

What is...

Term	Definition
Apache Spark	Open source, in memory unified analytics engine for big data and machine learning from UC Berkley
Modules/Libraries	SparkSQL, Mllib, Streaming, GraphX
Hive	Datawarehouse project for Hadoop giving SQL like interface to query data
Databricks	Spark based analytics platform
Databricks for Data Engineering Workloads	Only run jobs
Databricks for Data Analytics Workloads	Workspaces/Interactive Clusters (auto)
Azure Databricks Premium Plan	RBAC access / JDBC&ODBC connectivity / audit logs / R Studio integration (adds \$0.15 per hour)
Standard Cluster	Default – Use Python, R, Scala, and SQL
High Concurrency Cluster	Multiple concurrently active users based – Use SQL, Python, and R (no Scala)

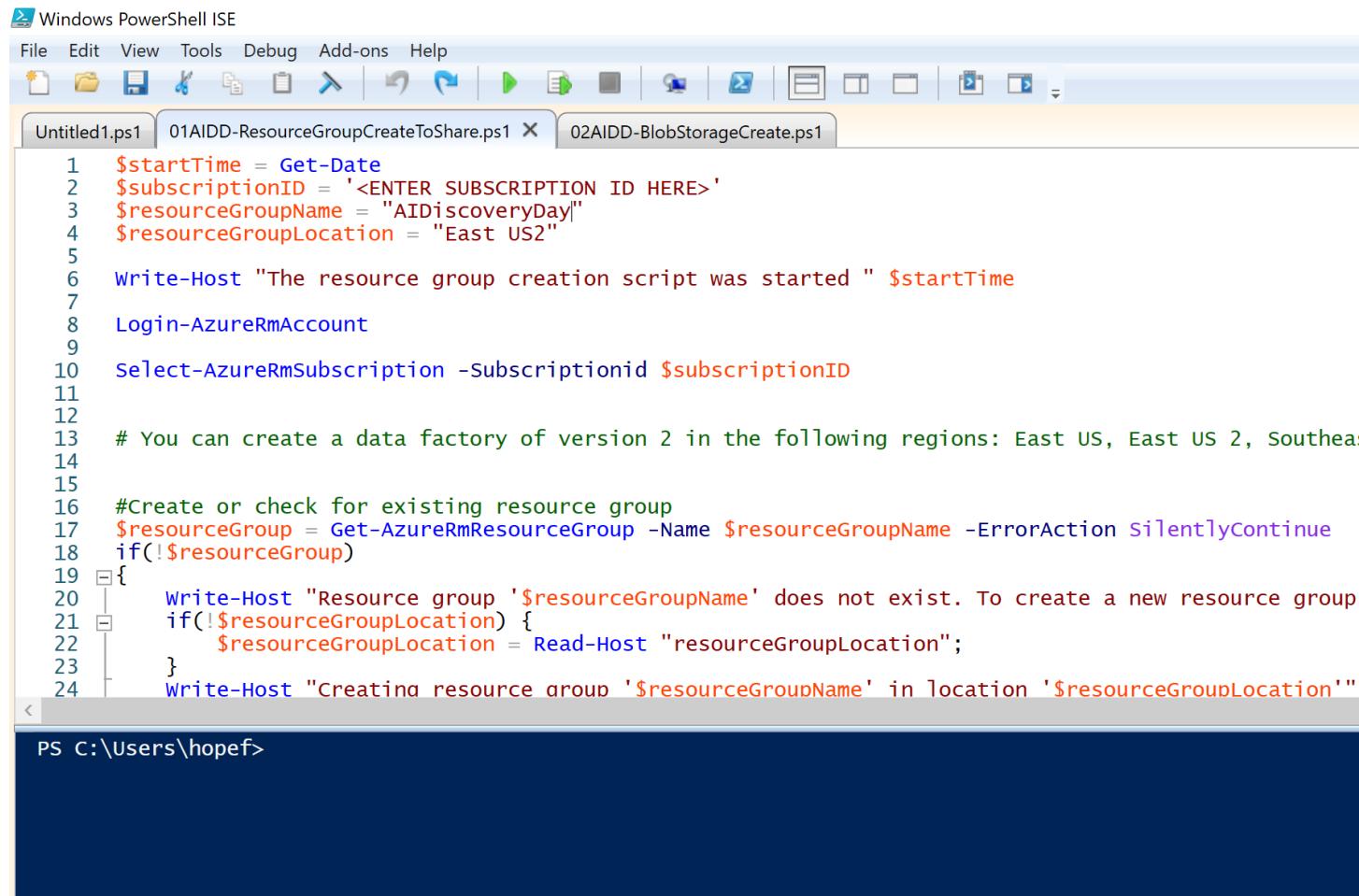
No more talking/all hands on now

Workshop

- Module 1 – PowerShell Create Environment
- Module 2 - Cluster Creation and Orientation
- Module 3 - Working in Notebook
 - Setup storage (security/AAD)
 - Working with SQL
 - Mix it up
- Module 4 - Visualization
 - Inside notebooks
 - Connect to Databricks from Power BI
- Bonus Module 5 (if time permits – Import Notebook/ML)
 - Import notebook
 - Explore ML model creation

Module 1 – PowerShell Create Environment

https://github.com/hfoley/AI-DD



Windows PowerShell ISE

File Edit View Tools Debug Add-ons Help

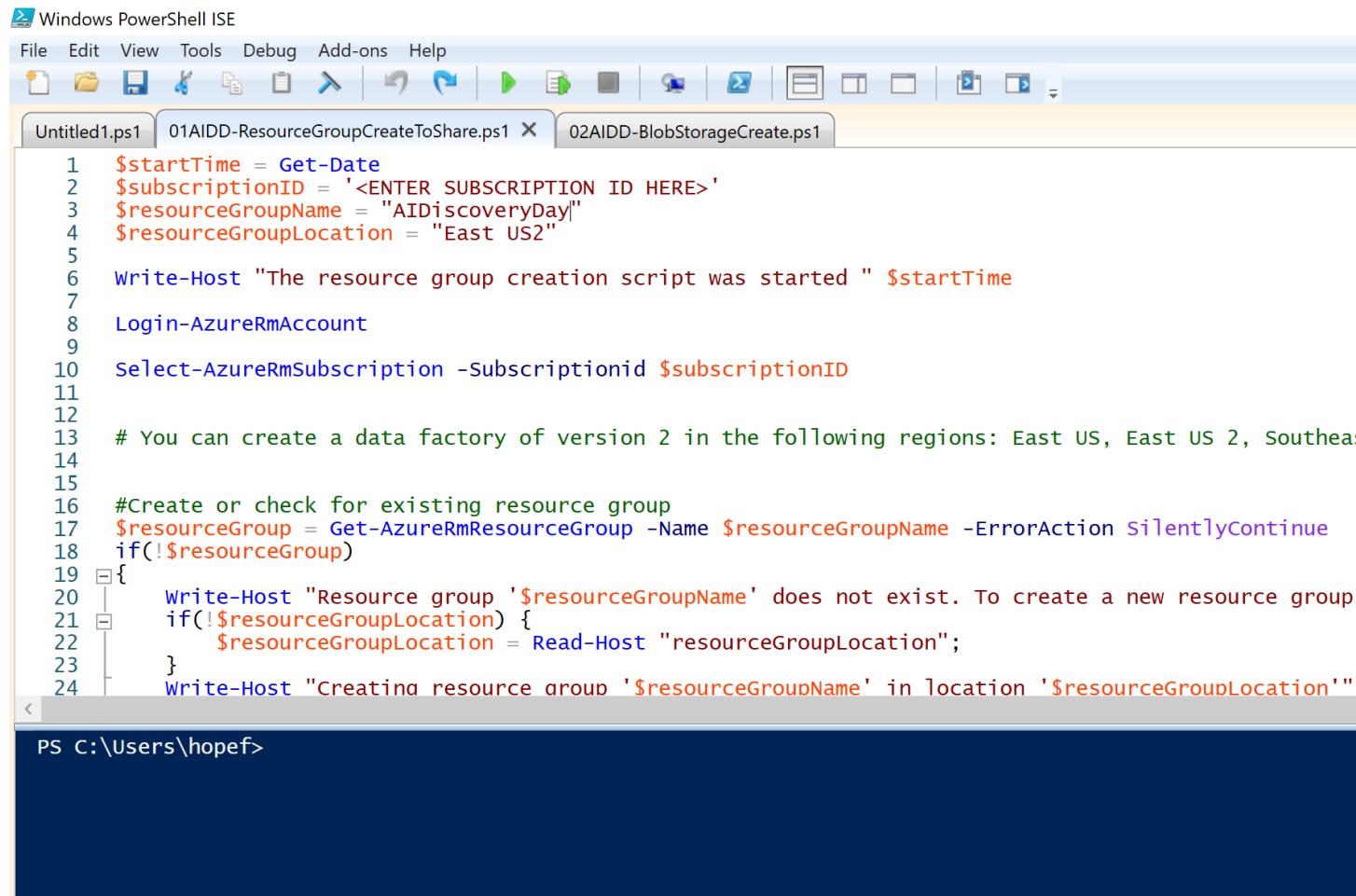
Untitled1.ps1 01AIDD-ResourceGroupCreateToShare.ps1 02AIDD-BlobStorageCreate.ps1

```
1 $startTime = Get-Date
2 $subscriptionID = '<ENTER SUBSCRIPTION ID HERE>'
3 $resourceGroupName = "AIDiscoveryDay"
4 $resourceGroupLocation = "East US2"
5
6 Write-Host "The resource group creation script was started " $startTime
7
8 Login-AzureRmAccount
9
10 Select-AzureRmSubscription -Subscriptionid $subscriptionID
11
12
13 # You can create a data factory of version 2 in the following regions: East US, East US 2, Southeast
14
15
16 #Create or check for existing resource group
17 $resourceGroup = Get-AzureRmResourceGroup -Name $resourceGroupName -ErrorAction SilentlyContinue
18 if(!$resourceGroup)
19 {
20     Write-Host "Resource group '$resourceGroupName' does not exist. To create a new resource group,
21     if(!$resourceGroupLocation) {
22         $resourceGroupLocation = Read-Host "resourceGroupLocation";
23     }
24     Write-Host "Creating resource group '$resourceGroupName' in location '$resourceGroupLocation':"
```

PS C:\Users\hopef>

Open PowerShell and open the files downloaded earlier

Open PowerShell ISE



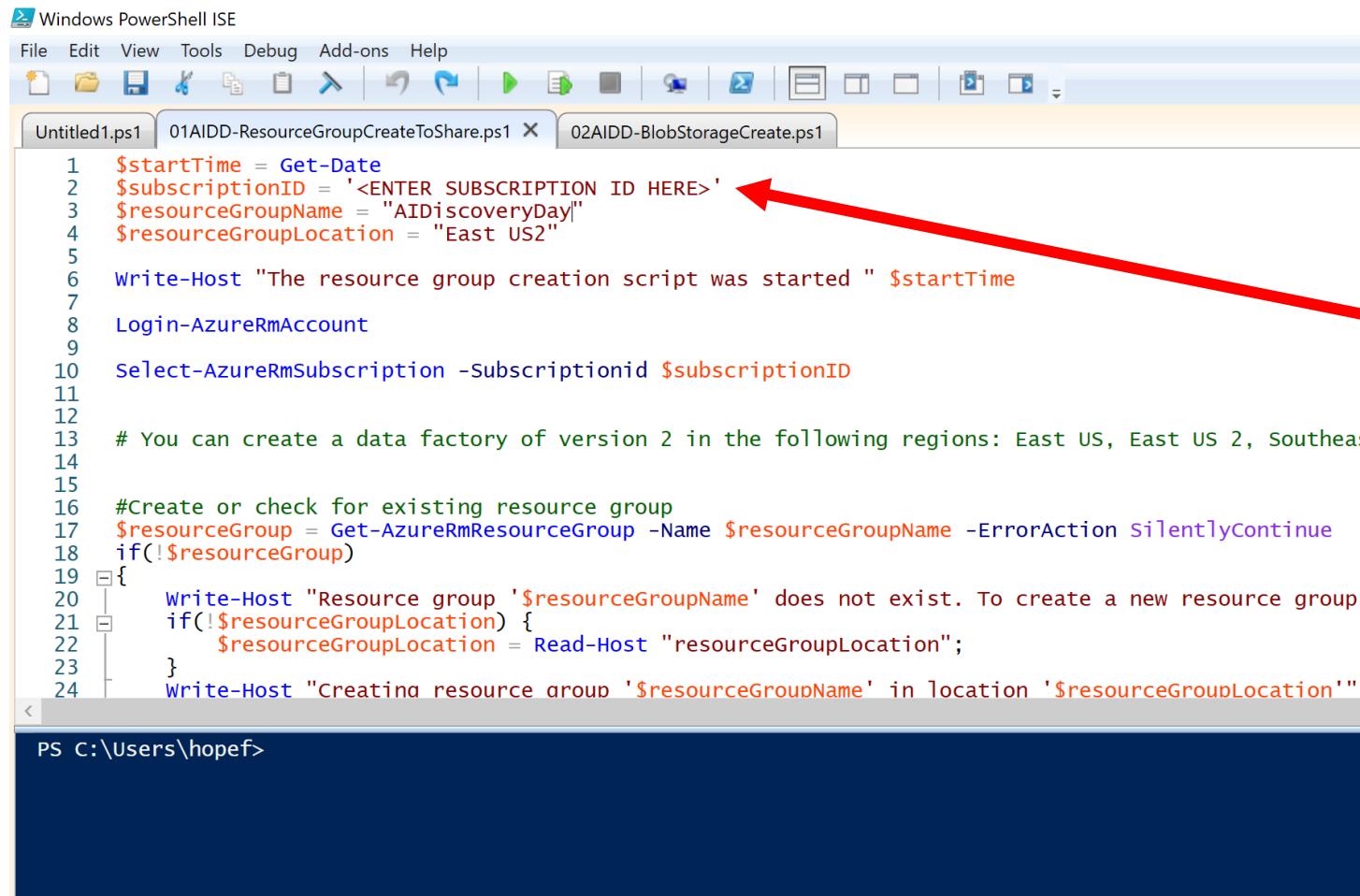
The image shows the Windows PowerShell ISE (Integrated Scripting Environment) interface. The title bar says "Windows PowerShell ISE". The menu bar includes File, Edit, View, Tools, Debug, Add-ons, and Help. The toolbar has icons for New, Open, Save, Cut, Copy, Paste, Find, Replace, and Run. There are two tabs open: "Untitled1.ps1" and "01AIDD-ResourceGroupCreateToShare.ps1". The "01AIDD-ResourceGroupCreateToShare.ps1" tab is active and contains the following PowerShell script:

```
1 $startTime = Get-Date
2 $subscriptionID = '<ENTER SUBSCRIPTION ID HERE>'
3 $resourceGroupName = "AIDDiscoveryDay"
4 $resourceGroupLocation = "East US2"
5
6 Write-Host "The resource group creation script was started " $startTime
7
8 Login-AzureRmAccount
9
10 Select-AzureRmSubscription -SubscriptionId $subscriptionID
11
12 # You can create a data factory of version 2 in the following regions: East US, East US 2, Southeast
13
14 #Create or check for existing resource group
15 $resourceGroup = Get-AzureRmResourceGroup -Name $resourceGroupName -ErrorAction SilentlyContinue
16 if(!$resourceGroup)
17 {
18     Write-Host "Resource group '$resourceGroupName' does not exist. To create a new resource group,
19     if(!$resourceGroupLocation) {
20         $resourceGroupLocation = Read-Host "resourceGroupLocation";
21     }
22     Write-Host "Creating resource group '$resourceGroupName' in location '$resourceGroupLocation':"
23 }
24
```

The status bar at the bottom shows "PS C:\Users\hopef>".

1. Open PowerShell and open the files downloaded earlier
2. Switch to 01AIDD-ResourceGroupCreateToShare.ps1 if necessary

Edit PowerShell script for your subscription

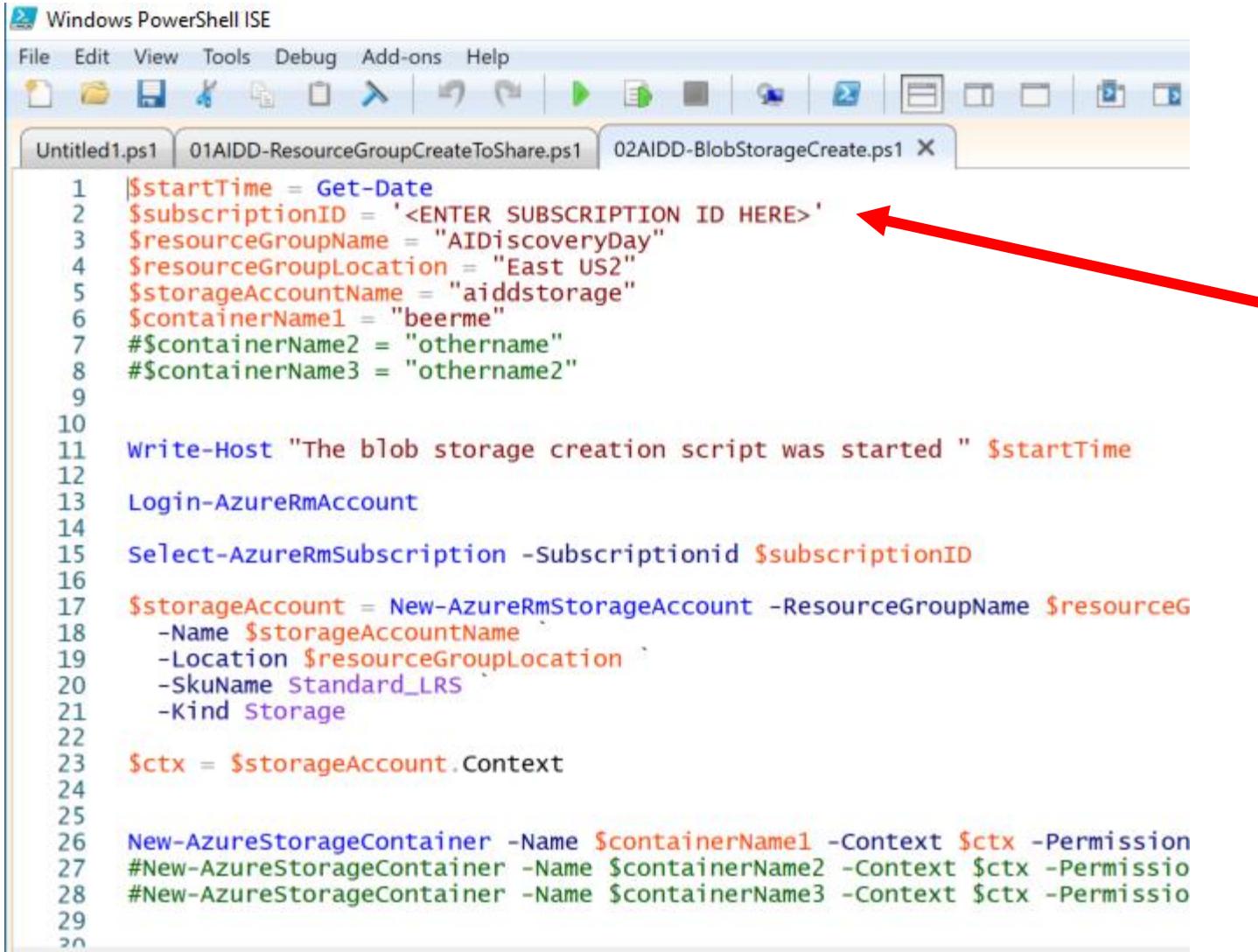


```
Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
Untitled1.ps1 01AIDD-ResourceGroupCreateToShare.ps1 02AIDD-BlobStorageCreate.ps1
1 $startTime = Get-Date
2 $SubscriptionID = '<ENTER SUBSCRIPTION ID HERE>' ←
3 $resourceGroupName = "AIDiscoveryDay"
4 $resourceGroupLocation = "East US2"
5
6 Write-Host "The resource group creation script was started " $startTime
7
8 Login-AzureRmAccount
9
10 Select-AzureRmSubscription -SubscriptionId $SubscriptionID
11
12
13 # You can create a data factory of version 2 in the following regions: East US, East US 2, Southeast
14
15
16 #Create or check for existing resource group
17 $resourceGroup = Get-AzureRmResourceGroup -Name $resourceGroupName -ErrorAction SilentlyContinue
18 if(!$resourceGroup)
19 {
20     Write-Host "Resource group '$resourceGroupName' does not exist. To create a new resource group,
21     if(!$resourceGroupLocation) {
22         $resourceGroupLocation = Read-Host "resourceGroupLocation";
23     }
24     Write-Host "Creating resource group '$resourceGroupName' in location '$resourceGroupLocation':"
```

PS C:\Users\hopef>

1. Enter your actual Subscription ID where indicated
2. Run the script
3. Authenticate to subscription when prompted
4. Validate in Azure Portal if possible, the creation of the AIDiscoveryDay resource group

Edit and run script 2



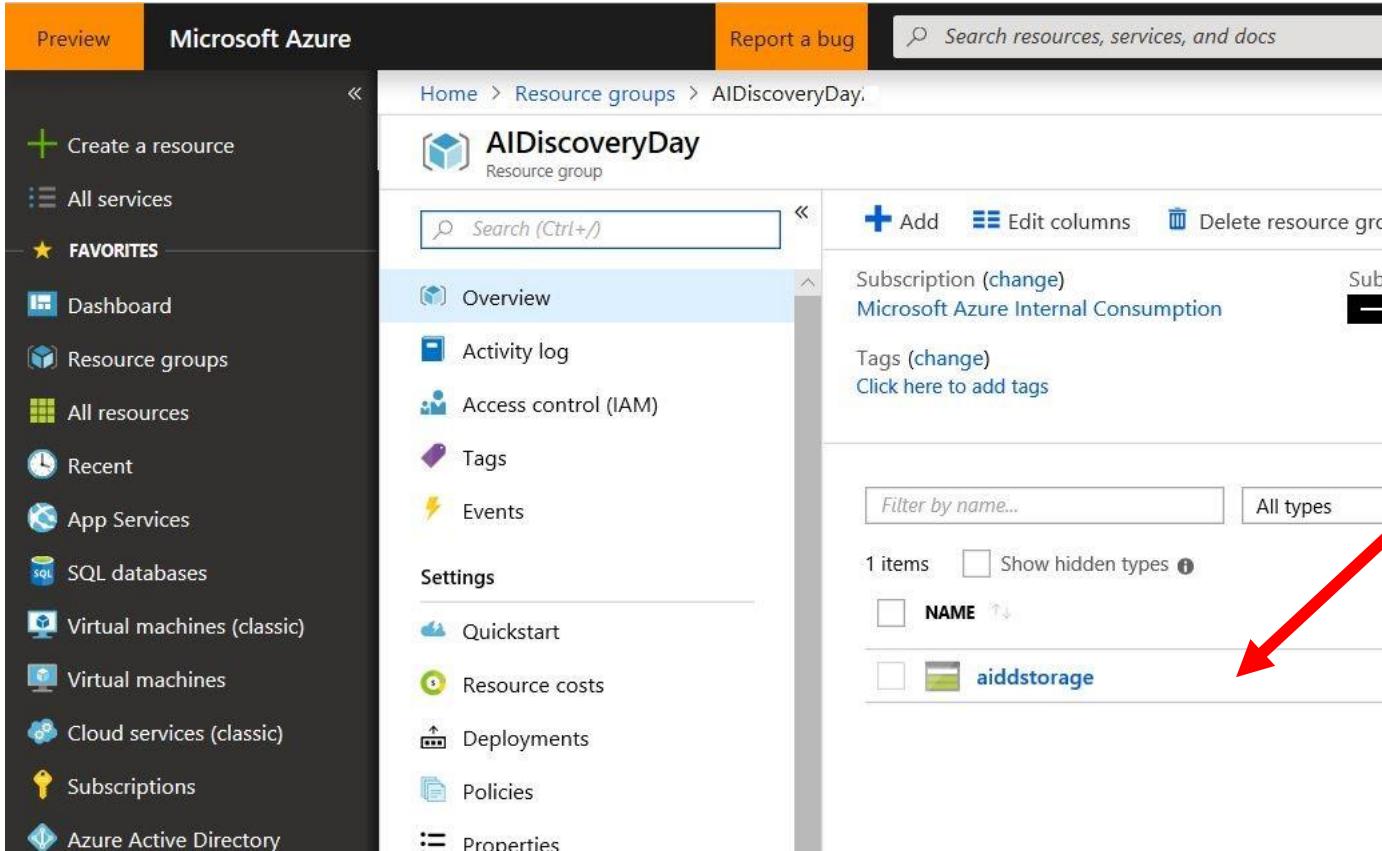
```
Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
Untitled1.ps1 01AIDD-ResourceGroupCreateToShare.ps1 02AIDD-BlobStorageCreate.ps1 X

1 $startTime = Get-Date
2 $subscriptionID = '<ENTER SUBSCRIPTION ID HERE>' ← Red arrow points here
3 $resourceGroupName = "AIDiscoveryDay"
4 $resourceGroupLocation = "East US2"
5 $storageAccountName = "aiddstorage"
6 $containerName1 = "beerme"
7 #$containerName2 = "othername"
8 #$containerName3 = "othername2"
9
10
11 Write-Host "The blob storage creation script was started " $startTime
12
13 Login-AzureRmAccount
14
15 Select-AzureRmSubscription -Subscriptionid $subscriptionID
16
17 $storageAccount = New-AzureRmStorageAccount -ResourceGroupName $resourceG
18   -Name $storageAccountName
19   -Location $resourceGroupLocation
20   -SkuName Standard_LRS
21   -Kind Storage
22
23 $ctx = $storageAccount.Context
24
25
26 New-AzureStorageContainer -Name $containerName1 -Context $ctx -Permission
27 #New-AzureStorageContainer -Name $containerName2 -Context $ctx -Permissio
28 #New-AzureStorageContainer -Name $containerName3 -Context $ctx -Permissio
29
30
```

1. Change to 02AIDD-BlobStorageCreate.ps1 in PowerShell.
2. Enter your actual Subscription ID where indicated
3. Run the script
4. Authenticate to subscription when prompted

*If running multiple times – change blob storage account name

Validate and upload data

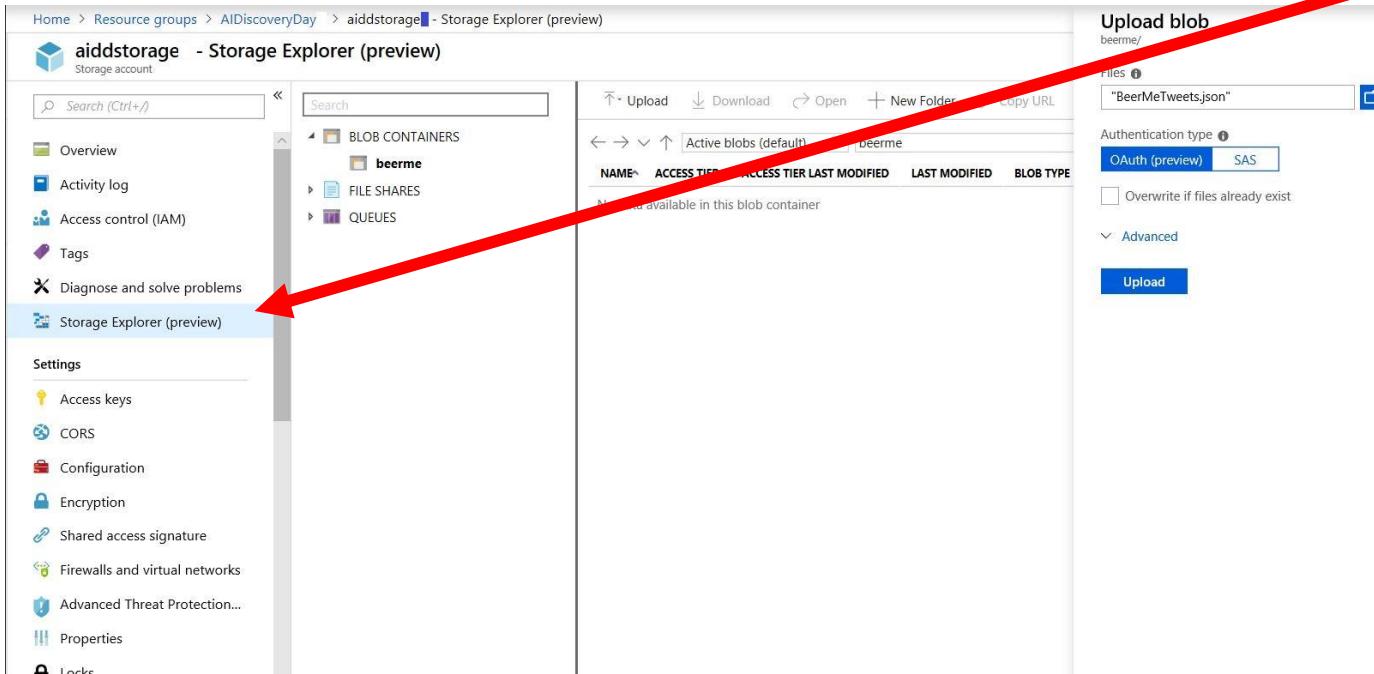


The screenshot shows the Microsoft Azure Portal interface. The top navigation bar includes 'Preview', 'Microsoft Azure', 'Report a bug', and a search bar. The left sidebar lists various service categories: 'Create a resource', 'All services', 'FAVORITES' (with 'Dashboard' selected), 'Resource groups', 'All resources', 'Recent', 'App Services', 'SQL databases', 'Virtual machines (classic)', 'Virtual machines', 'Cloud services (classic)', 'Subscriptions', and 'Azure Active Directory'. The main content area shows the 'Resource groups' page for the 'AIDiscoveryDay' group. The 'Overview' tab is active. The 'Subscription (change)' dropdown is set to 'Microsoft Azure Internal Consumption'. The 'Tags (change)' section has a link to 'Click here to add tags'. A red arrow points to the 'aiddstorage' blob storage account listed in the resource table, which includes columns for 'NAME' and 'TYPE'.

NAME	TYPE
aiddstorage	blob

1. Validate in Azure Portal if possible, the creation of the blob storage in previously created RG
2. Click on the storage account

Validate and upload data



1. Click Storage Explorer(preview)
2. Expand Blob Containers and select beerme
3. Select Upload
4. Select location for BeerMeTweets.json
5. Validate successful upload

View and capture storage information

Home > Resource groups > AIDiscoveryDay > aiddstorage - Access keys

aiddstorage - Access keys
Storage account

Search (Ctrl+)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Storage Explorer (preview)

Settings

Access keys

CORS

Configuration

Encryption

Shared access signature

Firewalls and virtual networks

Advanced Threat Protection...

Properties

Locks

Automation script

Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines. [Learn more](#)

Storage account name
aiddstorage

key1

Key
iCmcrlA6BzhJTJZkQVPM60M...

Connection string
itME1csAQoeHJgOmiVWtoPd...

key2

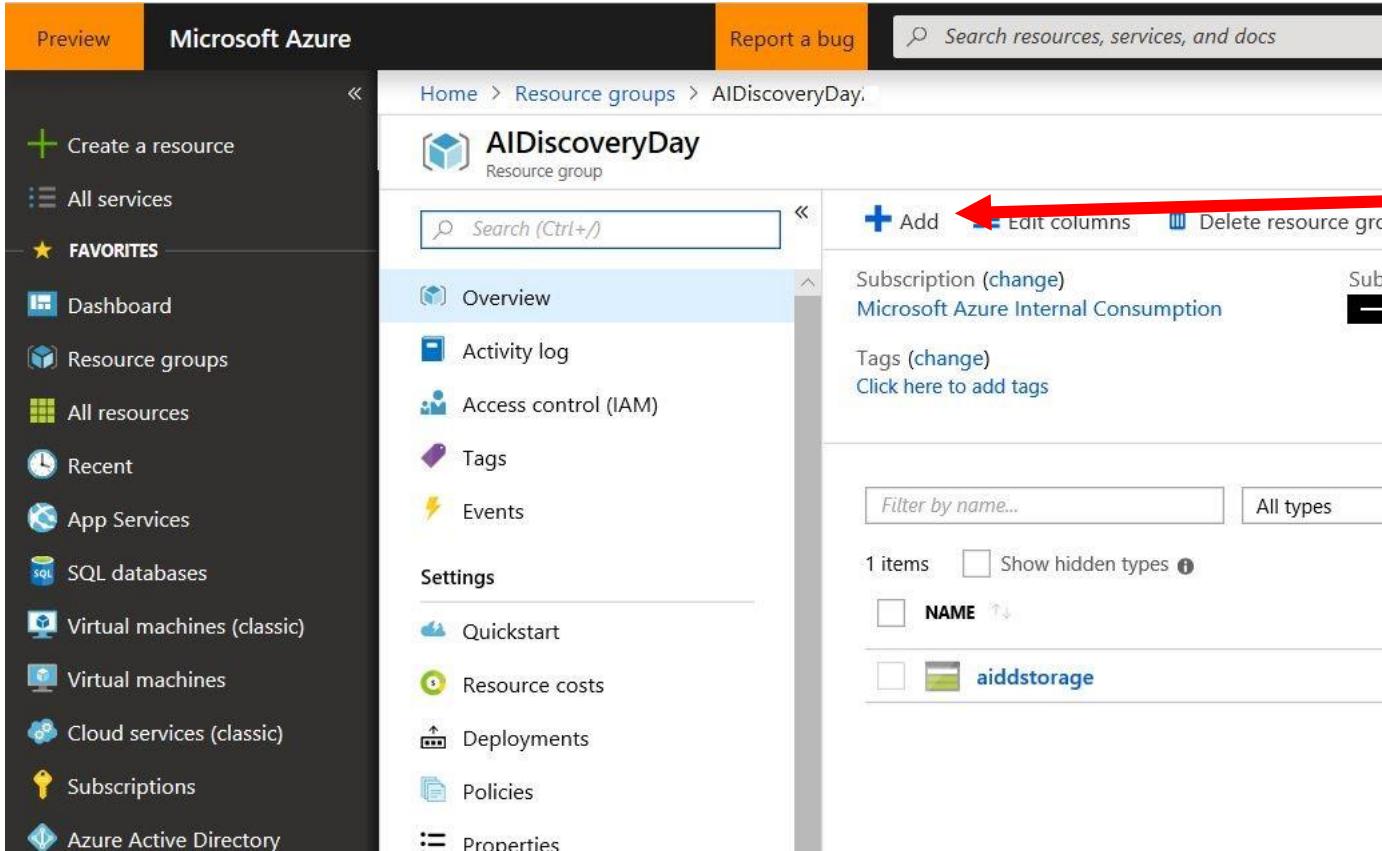
Key
itME1csAQoeHJgOmiVWtoPd...

Connection string
iCmcrlA6BzhJTJZkQVPM60M...

1. Click on the storage created
2. Click Access keys
3. Copy storage account name and save for use later
4. Copy and save one of the key values to use later

Module 2 - Cluster Creation and Orientation

Create Databricks Cluster



The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes 'Preview', 'Microsoft Azure', 'Report a bug', and a search bar. Below the navigation is a breadcrumb trail: Home > Resource groups > AIDiscoveryDay. The main content area is titled 'AIDiscoveryDay' and shows the 'Overview' tab selected. The 'Add' button in the top right corner is highlighted with a red arrow. The 'Add' button is part of a row of buttons: 'Add', 'Edit columns', and 'Delete resource group'. The 'Add' button is accompanied by a plus sign icon. The 'Edit columns' button has a gear icon, and the 'Delete resource group' button has a trash bin icon. The 'Overview' section displays 'Subscription (change) Microsoft Azure Internal Consumption' and a 'Tags (change)' section with a 'Click here to add tags' link. On the left side, a sidebar lists various Azure services: 'Create a resource', 'All services', 'FAVORITES' (with a star icon), 'Dashboard', 'Resource groups', 'All resources', 'Recent', 'App Services', 'SQL databases', 'Virtual machines (classic)', 'Virtual machines', 'Cloud services (classic)', 'Subscriptions', and 'Azure Active Directory'. The 'Resource groups' item is currently selected.

1. Select Add within the AIDiscoveryDay RG
2. Find and select Azure Databricks
3. Select Create

Specify Details for Databricks Cluster

Azure Databricks Service

* Workspace name

* Subscription

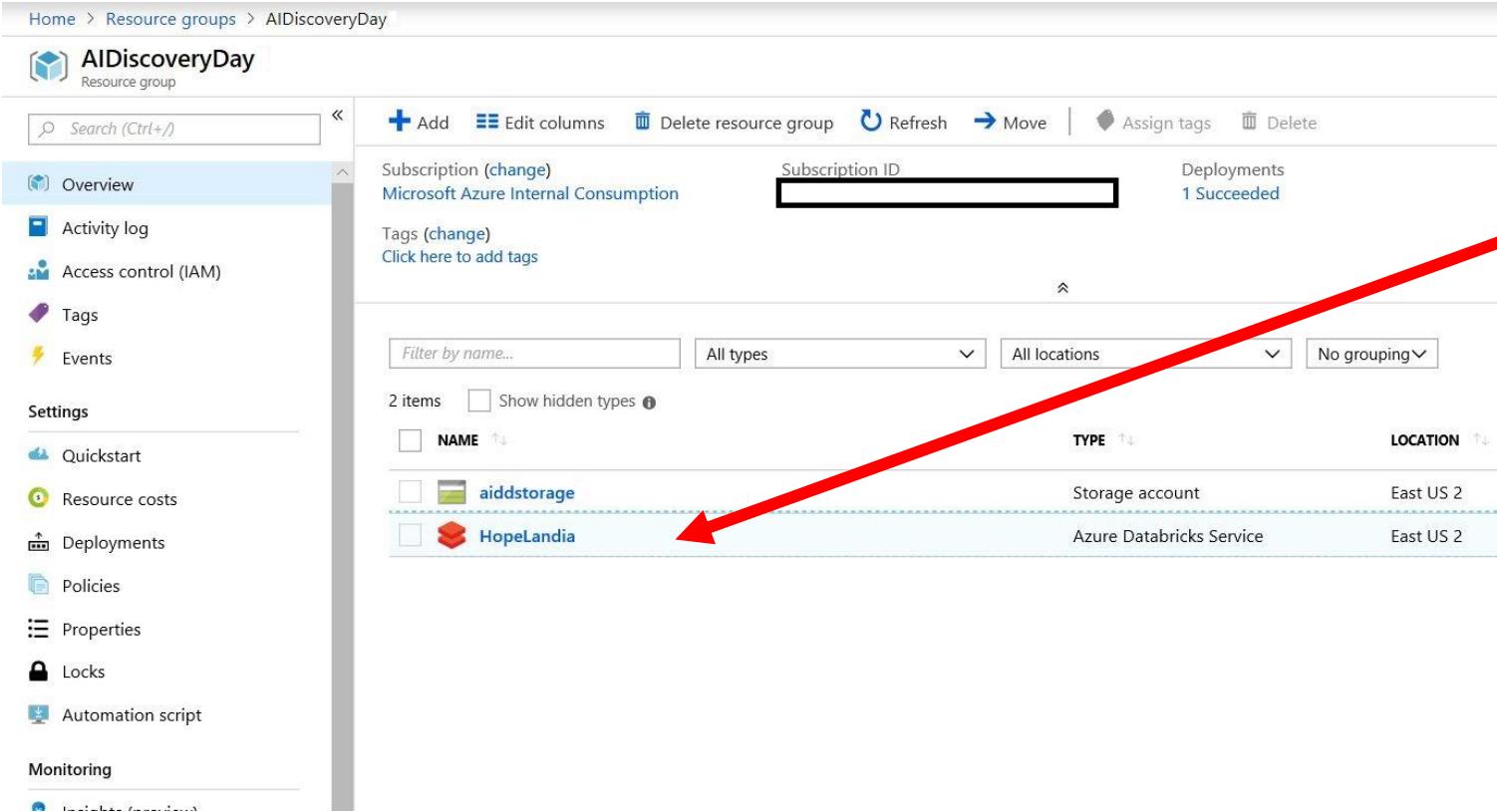
* Resource group ?
 Create new Use existing

* Location

* Pricing Tier (View full pricing details)

1. Specify name you wish and other details like so.
2. Click Create

Validate Databricks Cluster



Home > Resource groups > AIDiscoveryDay

AIDiscoveryDay
Resource group

Search (Ctrl+I)

+ Add Edit columns Delete resource group Refresh Move Assign tags Delete

Subscription (change) Microsoft Azure Internal Consumption Subscription ID [REDACTED] Deployments 1 Succeeded

Tags (change) Click here to add tags

Overview Activity log Access control (IAM) Tags Events

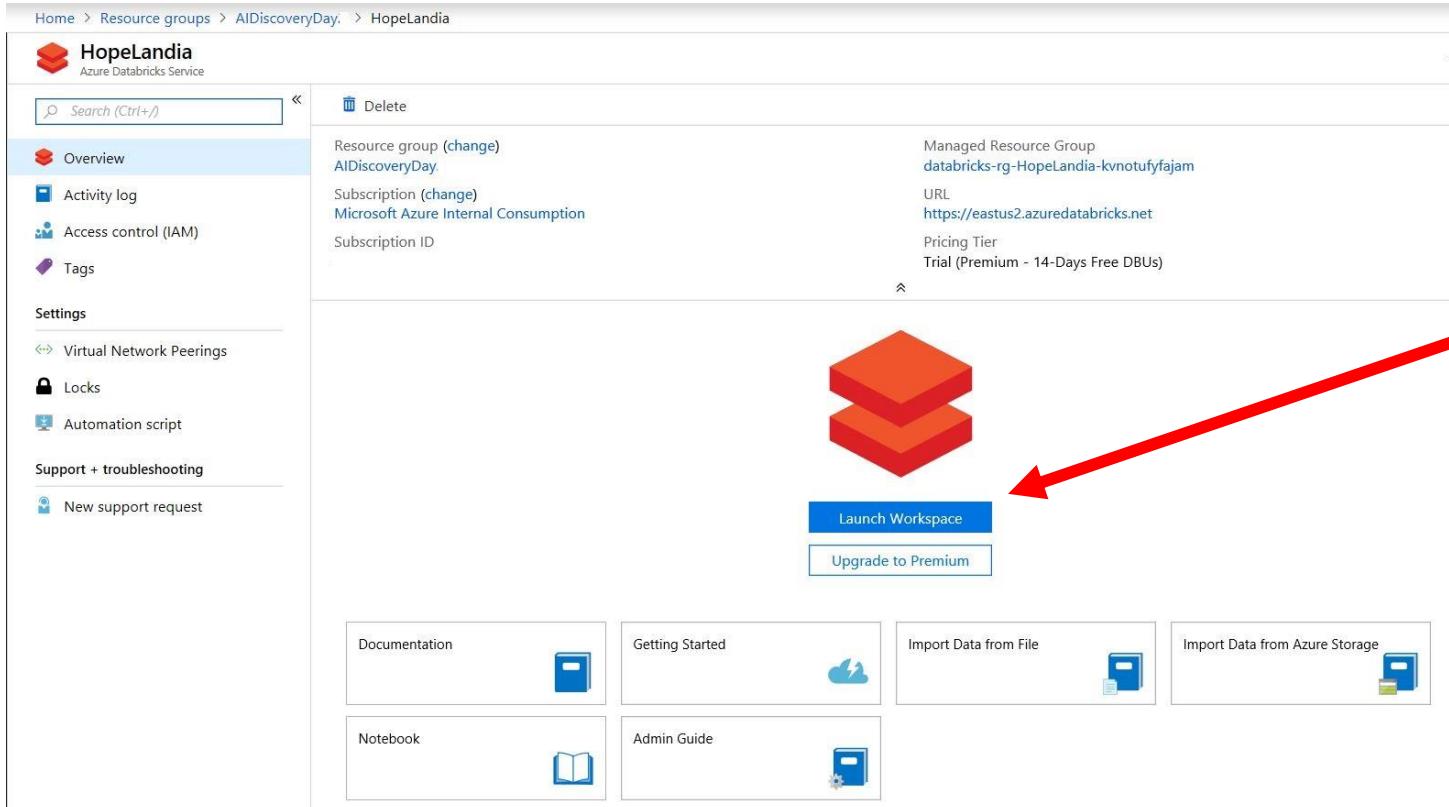
Filter by name... All types All locations No grouping

2 items Show hidden types

NAME	TYPE	LOCATION
addstorage	Storage account	East US 2
HopeLandia	Azure Databricks Service	East US 2

1. Go back to the resource group **AIDiscoveryDay** and validate Azure Databricks Service is listed
2. Click on the Databricks Service

Launch Databricks Workspace



The screenshot shows the Azure Databricks service overview page for the 'HopeLandia' cluster. The left sidebar includes links for Home, Resource groups, AIDiscoveryDay, and HopeLandia. The main content area displays cluster details: Resource group (change) AIDiscoveryDay, Subscription (change) Microsoft Azure Internal Consumption, Managed Resource Group databricks-rg-HopeLandia-kvnotufyfajam, URL <https://eastus2.azuredatabricks.net>, and Pricing Tier Trial (Premium - 14-Days Free DBUS). A large red arrow points to the 'Launch Workspace' button, which is highlighted in blue. Below the button are two other buttons: 'Upgrade to Premium' and 'Documentation' (with a book icon). Other links include 'Getting Started' (with a cloud icon), 'Import Data from File' (with a folder icon), 'Import Data from Azure Storage' (with a folder and cloud icon), 'Notebook' (with an open book icon), and 'Admin Guide' (with a book icon).

1. Look at the details of the cluster. Capture the URL
2. Click on Launch Workspace

View Workspace

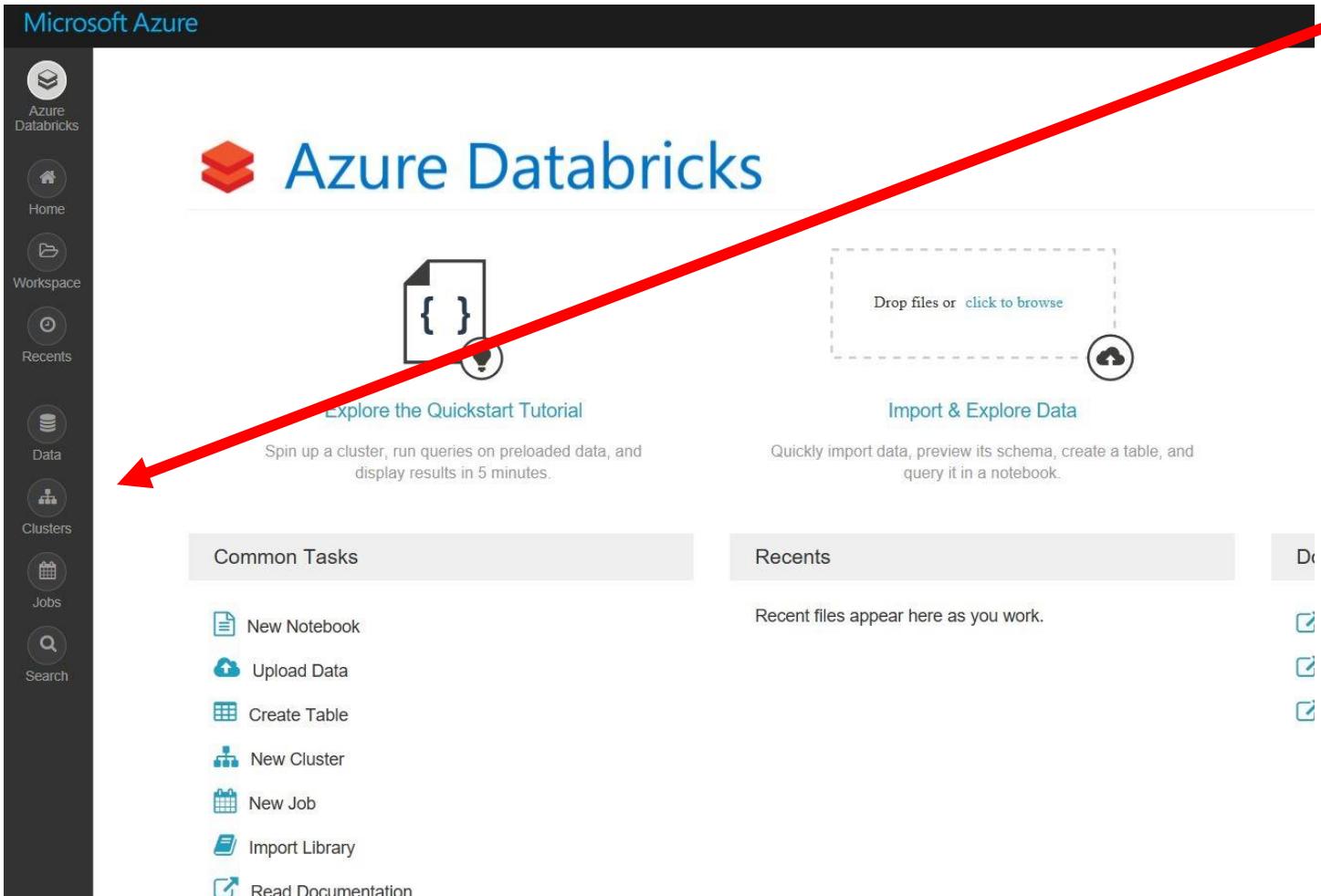
Microsoft Azure PORTAL

Free trial ends in 14 days. Upgrade to Premium in Azure Portal ?

Azure Databricks

1. Look at the details of the cluster. Capture the URL
2. Click on Launch Workspace

Setup Cluster



The screenshot shows the Azure Databricks home page. On the left, a vertical sidebar lists navigation items: Microsoft Azure, Azure Databricks (selected), Home, Workspace, Recents, Data, Clusters (selected), Jobs, and Search. The main content area features the 'Azure Databricks' logo. Below it, there are two main sections: 'Explore the Quickstart Tutorial' (with a 'New Notebook' icon) and 'Import & Explore Data' (with a 'Drop files or click to browse' area and a cloud icon). At the bottom, there are 'Common Tasks' (New Notebook, Upload Data, Create Table, New Cluster, New Job, Import Library, Read Documentation) and 'Recents' (Recent files list).

1. Click on Clusters
2. Click Create Cluster

Cluster Config

Microsoft Azure

Create Cluster

New Cluster |

2-8 Workers: 28.0-112.0 GB Memory, 8-32 Cores, 1.5-6 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU Cost \$0.55 per DBU

Cluster Name:

Cluster Mode: Standard High Concurrency

Optimized to run concurrent SQL, Python, and R workloads.
Does not support Scala. Previously known as Serverless.

Databricks Runtime Version:

Python Version:

Driver Type:

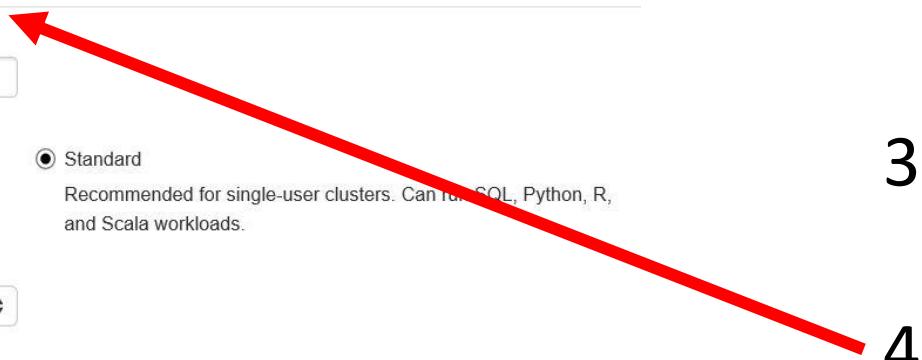
Worker Type: Min Workers: Max Workers: Enable autoscaling

Auto Termination: Terminate after minutes of inactivity

Spark Logging

Spark Config

Enter your Spark configuration options here. Provide only one key-value pair per line.
Example:



1. Select Name of choice
2. Select Standard cluster mode
3. Other options to left or similar.
4. Click Create Cluster

Cluster Creation

Microsoft Azure

Clusters

Free trial ends in 14 days. [Upgrade to Prem](#)

[+ Create Cluster](#)

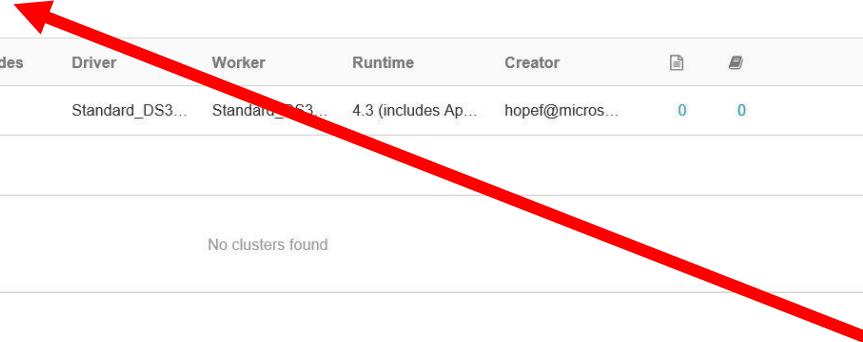
All Created by me Accessible by me

Interactive Clusters

Name	State	Nodes	Driver	Worker	Runtime	Creator	⋮	⋮
HopeLandia	Pending	0	Standard_DS3...	Standard_C3...	4.3 (includes Ap...	hopef@micros...	0	0

Job Clusters

No clusters found



Azure Databricks

Home

Workspace

Recents

Data

Clusters

1. Await for the creation

Cluster Creation Complete

Microsoft Azure

Clusters

Free trial ends in 14 days. [Upgrade](#)

All Created by me Accessible by me

Name	State	Nodes	Driver	Worker	Runtime	Creator	Actions
HopeLandia	Running	3	Standard_DS3...	Standard_DS3...	4.3 (includes Ap...	hopef@micros...	0 0

Interactive Clusters

Job Clusters

No clusters found

1. When completed will see cluster in a state of “Running”
2. Click on the cluster

Cluster Config/Admin

Microsoft Azure

Clusters / HopeLandia

HopeLandia      

Configuration [Notebooks \(0\)](#) [Libraries \(0\)](#) [Event Log](#) [Spark UI](#) [Driver Logs](#) [Spark Cluster UI - Master](#) 

Cluster Mode

High Concurrency
Optimized to run concurrent SQL, Python, and R workloads.
Does not support Scala. Previously known as Serverless.

Standard
Recommended for single-user clusters. Can run SQL, Python, R, and Scala workloads.

Databricks Runtime Version

4.3 (includes Apache Spark 2.3.1, Scala 2.11)

Python Version 

2

Driver Type

Standard_DS3_v2 14.0 GB Memory, 4 Cores, 0.75 DBU

Worker Type

Min Workers	Max Workers
2	3

Enable autoscaling 

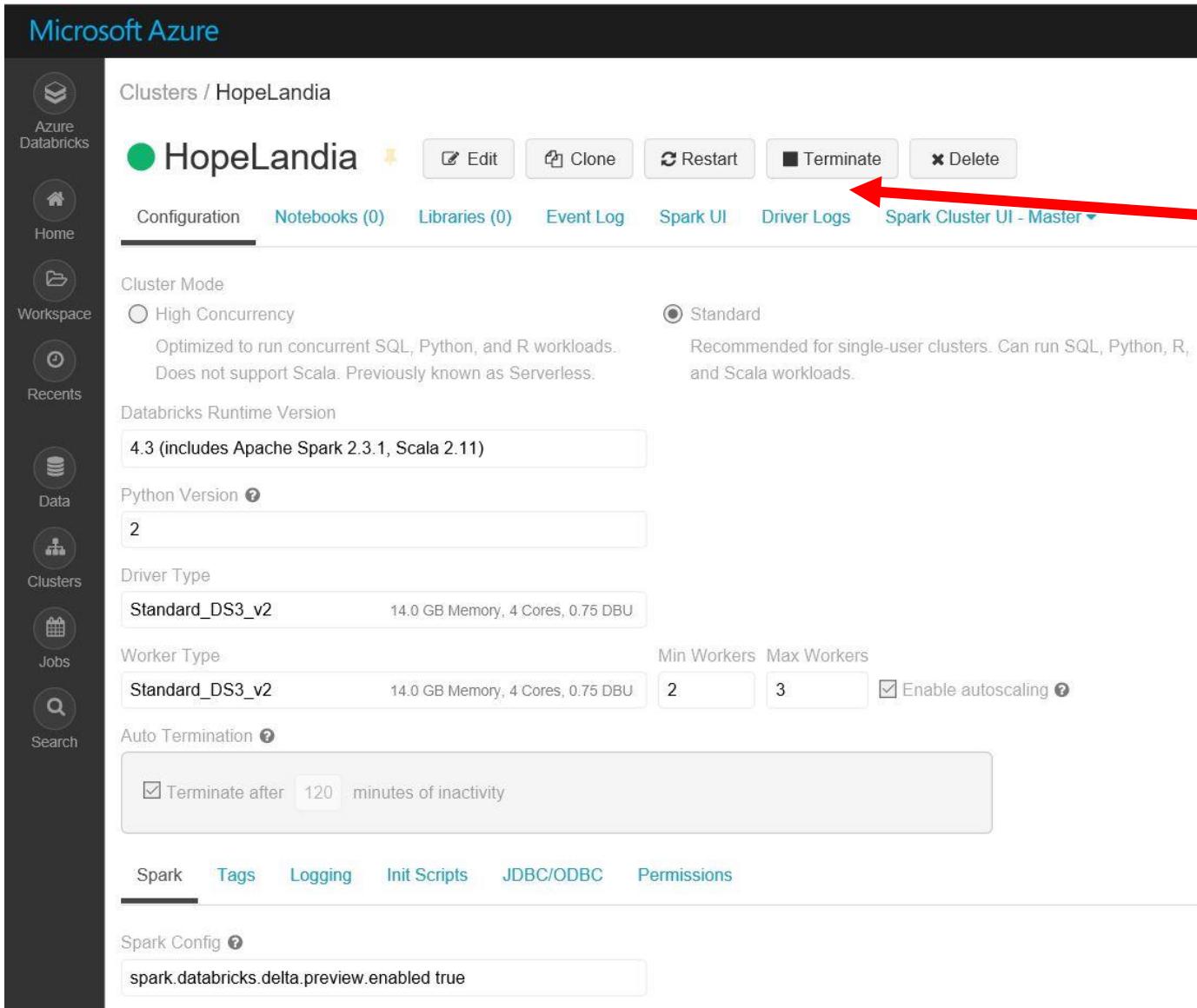
Auto Termination 

Terminate after 120 minutes of inactivity

Spark [Tags](#) [Logging](#) [Init Scripts](#) [JDBC/ODBC](#) [Permissions](#)

Spark Config 

spark.databricks.delta.preview.enabled true



1. Cluster administration is done here
2. Note the Terminate button

Add storage to cluster configuration

Microsoft Azure

Clusters / HopeLandia

HopeLandia     

Configuration Notebooks (0) Libraries (0) Event Log Spark UI Driver Logs Spark Cluster UI - Master

Cluster Mode

High Concurrency
Optimized to run concurrent SQL, Python, and R workloads.
Does not support Scala. Previously known as Serverless.

Standard
Recommended for single-user clusters. Can run SQL, Python, R, and Scala workloads.

Databricks Runtime Version

4.3 (includes Apache Spark 2.3.1, Scala 2.11)

Python Version

2

Driver Type

Standard_DS3_v2 14.0 GB Memory, 4 Cores, 0.75 DBU

Worker Type

Standard_DS3_v2 14.0 GB Memory, 4 Cores, 0.75 DBU

Min Workers Max Workers

2 3 Enable autoscaling

Auto Termination

Terminate after 120 minutes of inactivity

Spark Tags Logging Init Scripts JDBC/ODBC Permissions

Spark Config

```
spark.databricks.delta.preview.enabled true
```

Azure Databricks

Home

Workspace

Recents

Data

Clusters

Jobs

Search

1. Click Edit
2. Scroll to Spark Config
3. We are going to add storage account key at the cluster level to use. The format will use is:
`spark.hadoop.fs.azure.account.key.<storageAccount>.blob.core.windows.net <key>`

Add storage key adding to Spark Config

Microsoft Azure

Clusters / HopeLandia

HopeLandia

Confirm and Restart

2-3 Workers: 28.0-42.0 GB Memory, 8-12 Cores, 1.5-2.25 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU Cost \$0.55 per DBU

Driver Type: Standard_DS3_v2 (14.0 GB Memory, 4 Cores, 0.75 DBU)

Worker Type: Standard_DS3_v2 (14.0 GB Memory, 4 Cores, 0.75 DBU)

Min Workers: 2, Max Workers: 3, Enable autoscaling

Auto Termination: Terminate after 120 minutes of inactivity

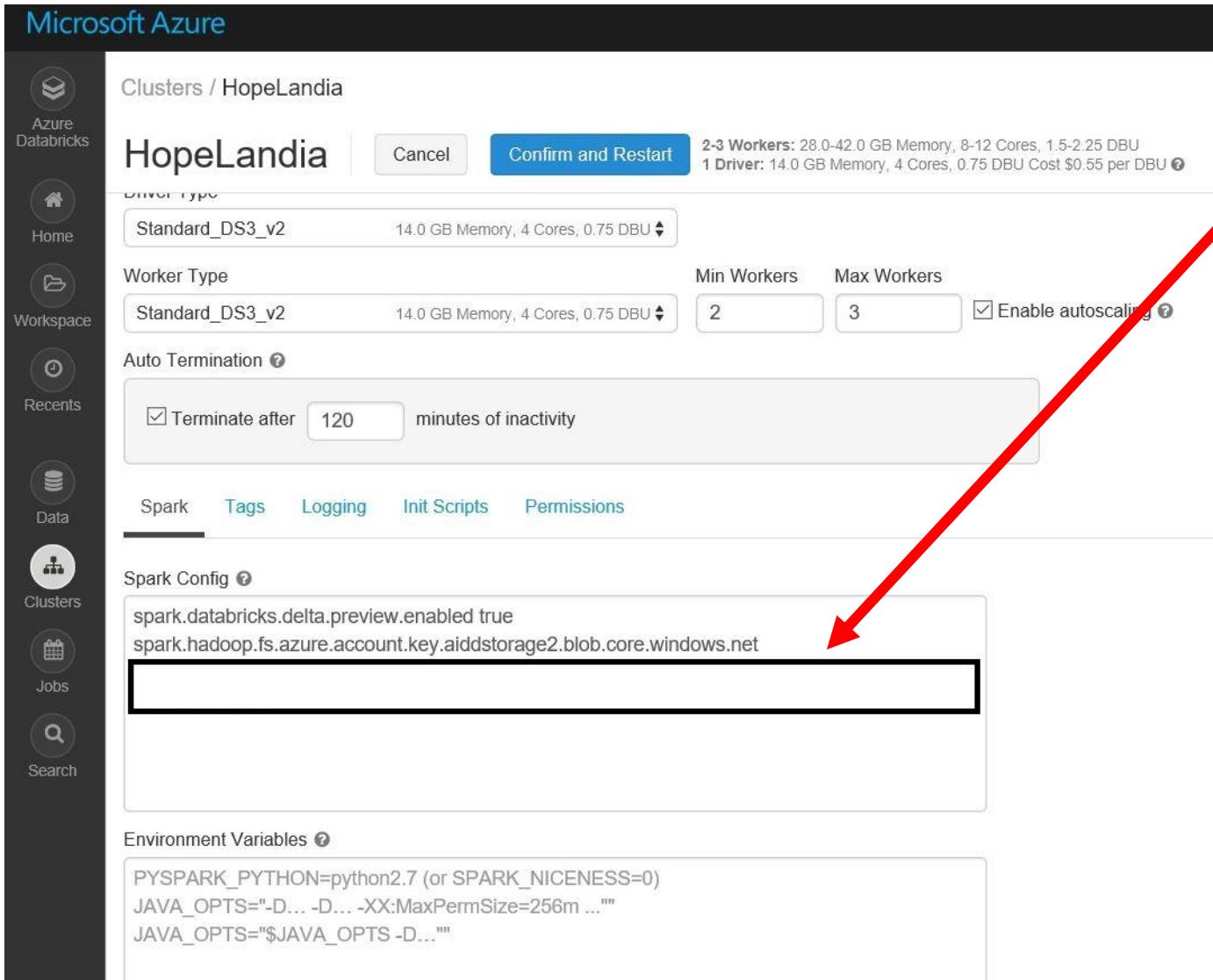
Spark Tags Logging Init Scripts Permissions

Spark Config

```
spark.databricks.delta.preview.enabled true
spark.hadoop.fs.azure.account.key.addstorage2.blob.core.windows.net
```

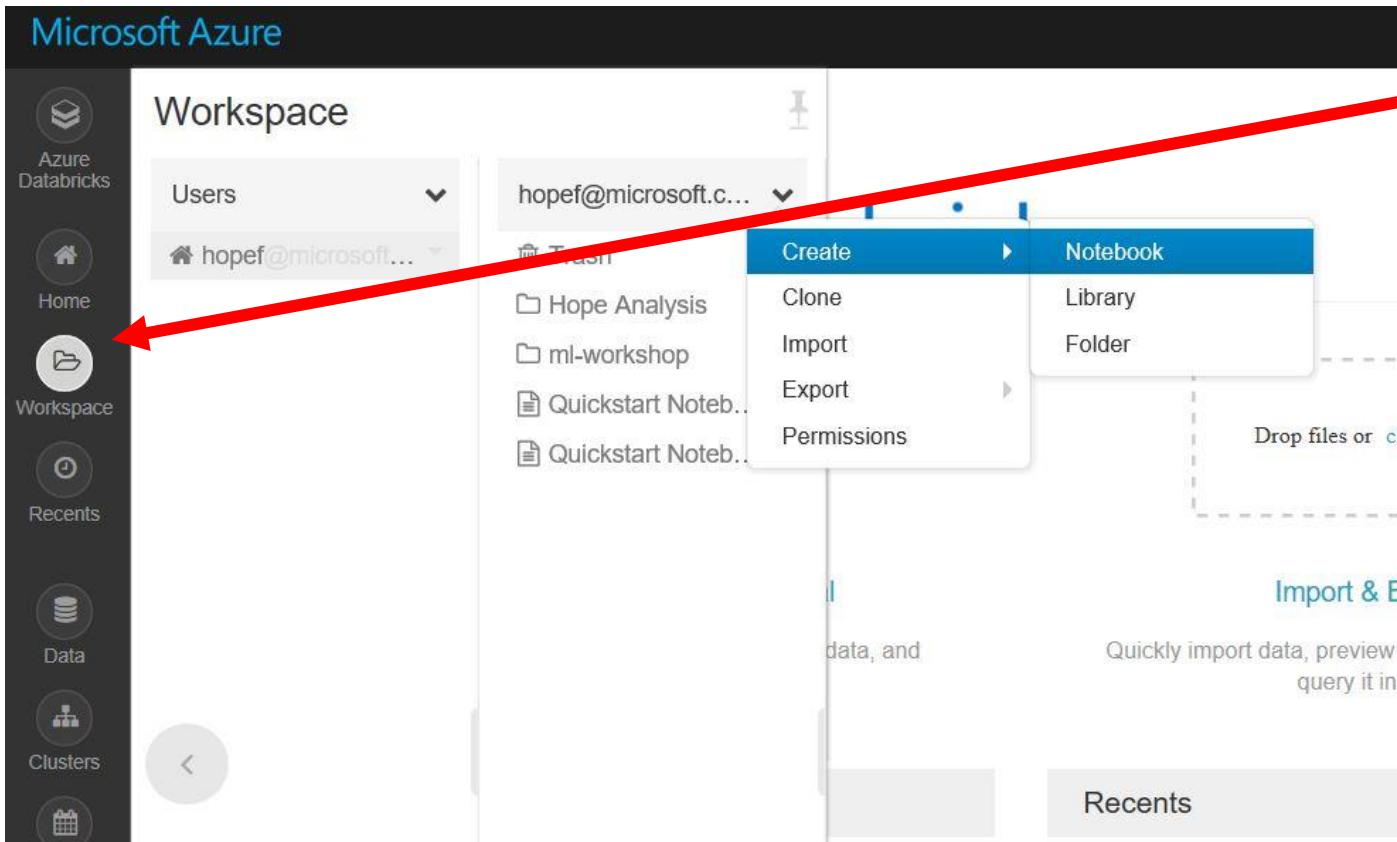
Environment Variables

```
PYSPARK_PYTHON=python2.7 (or SPARK_NICENESS=0)
JAVA_OPTS="-D... -D... -XX:MaxPermSize=256m ..."
JAVA_OPTS="$JAVA_OPTS -D..."
```



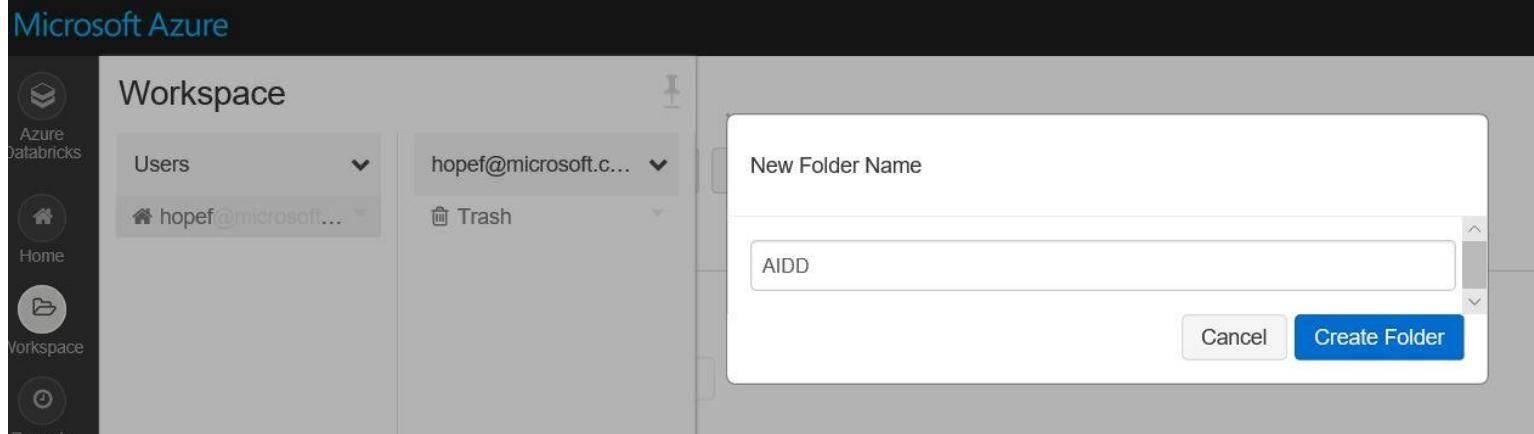
1. Add that value to the Spark Config
2. Click Confirm and Restart
3. Click Confirm

Create Workspace Pieces



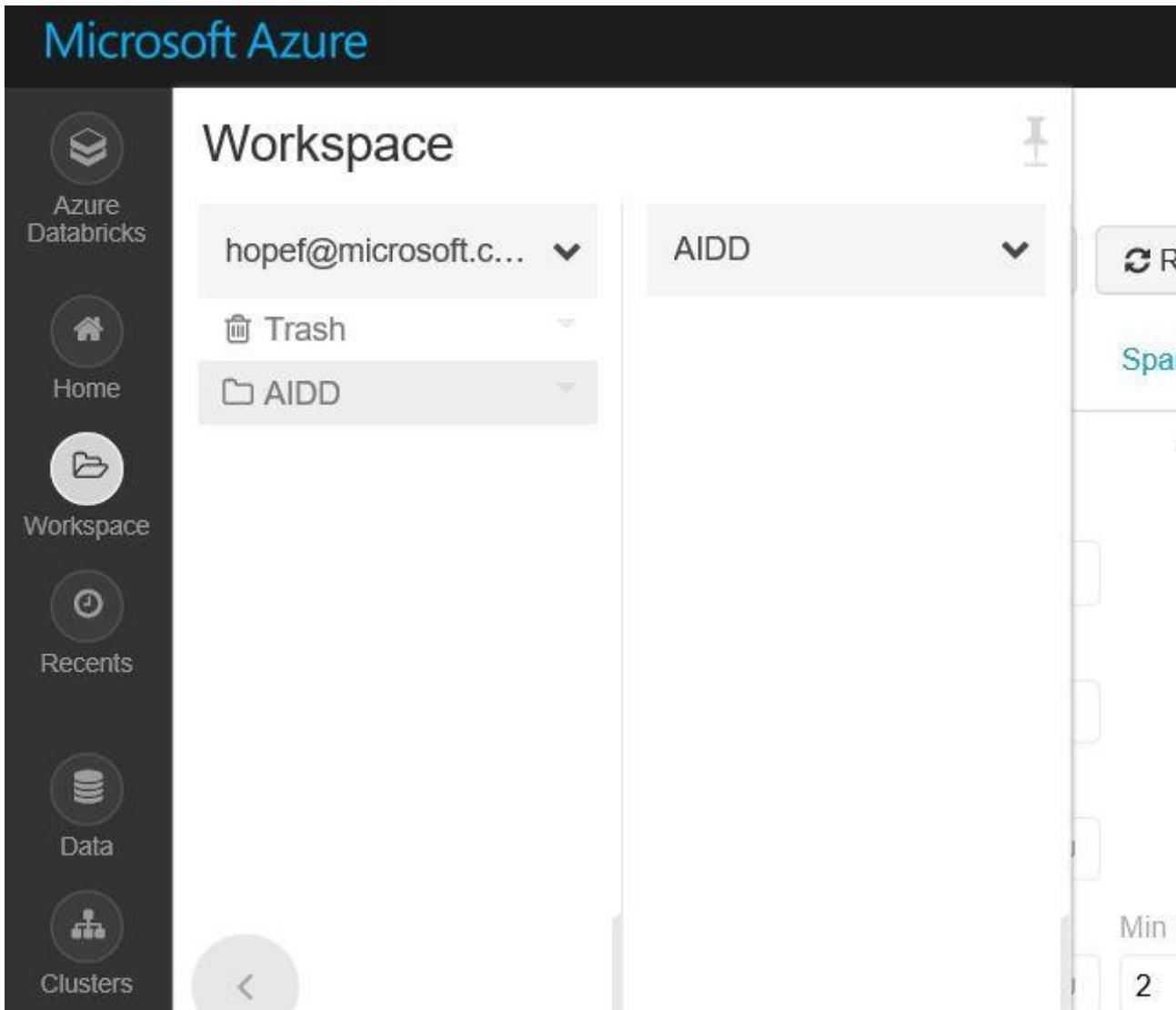
1. Click Workspace
2. Expand Users
3. Click on your user
4. Click Down Arrow
5. Choose Create

Create Folder



1. Select Folder
2. Specify Folder Name
“AIDD”
3. Select Create Folder

Look at Folder



1. Select the AIDD folder
2. Nothing to see in there just yet

Import Notebook

Microsoft Azure

Workspace

hopef@microsoft.c... ▾

Trash

📁 AIDD

AIDD

Restart

Terminate

Create

Clone

Rename

Move

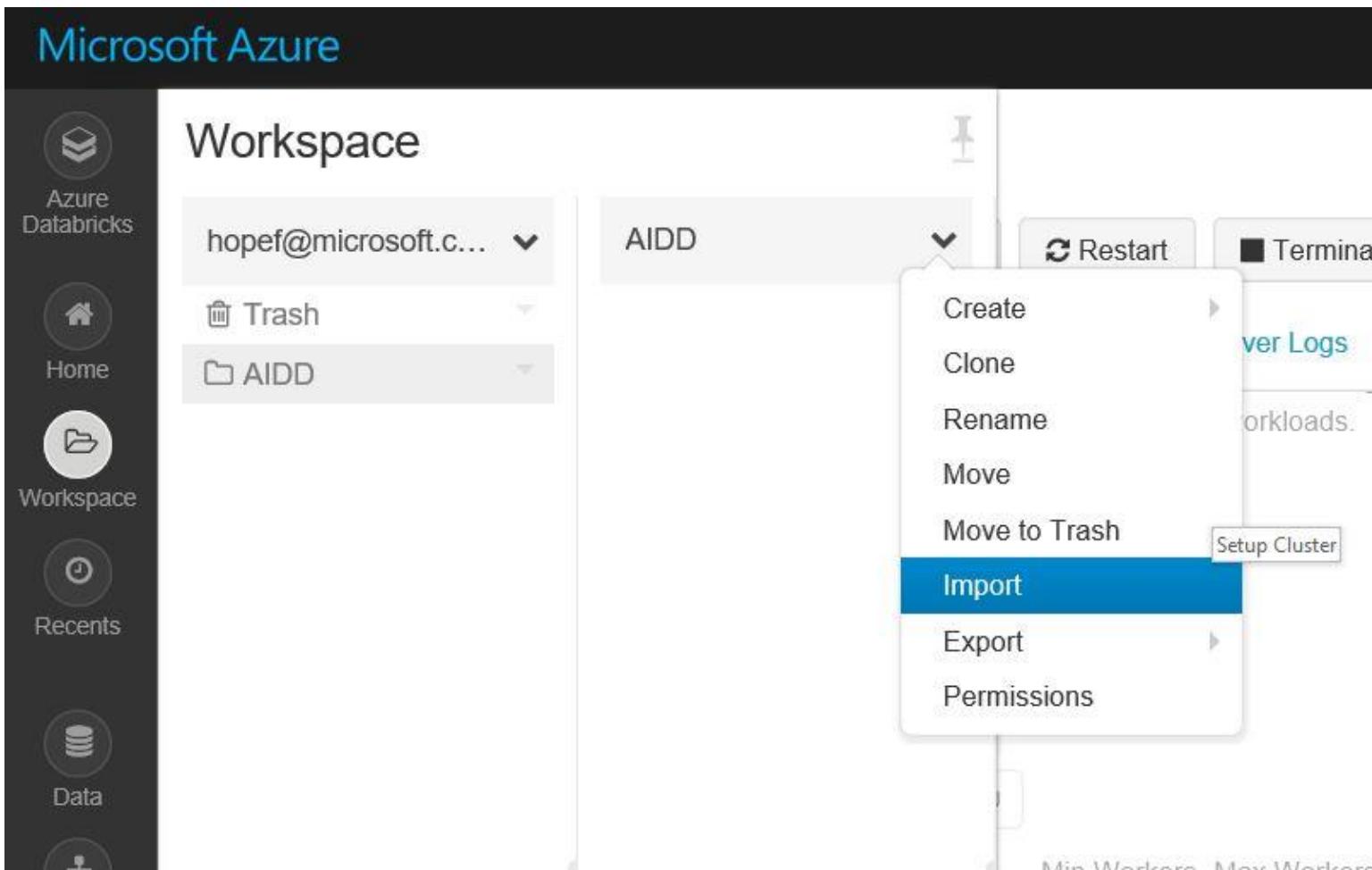
Move to Trash

Import

Export

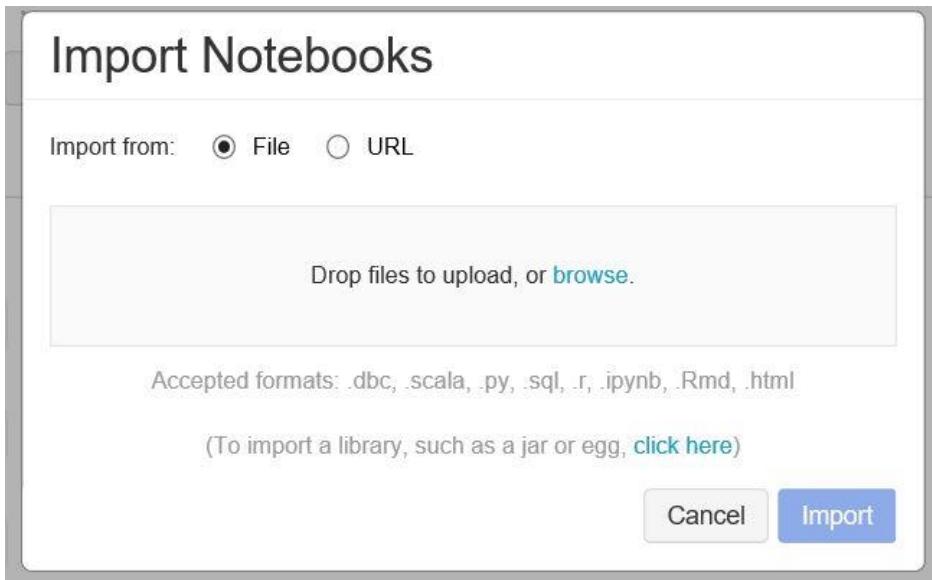
Permissions

Setup Cluster

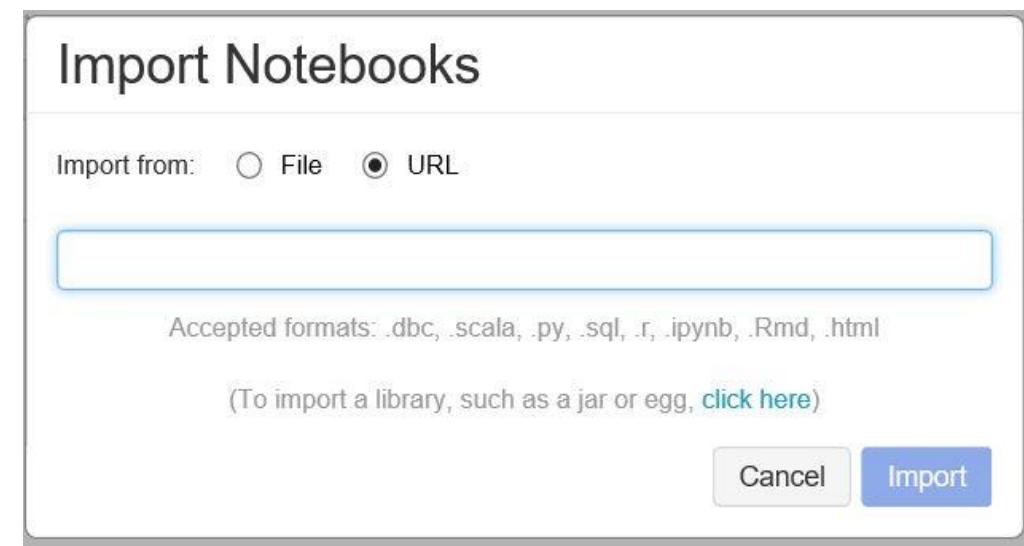
A screenshot of the Microsoft Azure Databricks workspace. The left sidebar shows navigation links: Azure Databricks, Home, Workspace, Recents, and Data. The main workspace shows a folder named 'AIDD' in the 'Workspace' section. A context menu is open over the 'AIDD' folder, listing options: Create, Clone, Rename, Move, Move to Trash, Import (which is highlighted in blue), Export, and Permissions. A sub-menu for 'Import' is also visible, containing 'Setup Cluster'. The background shows some notebook-related text: 'ver Logs', 'orkloads.', 'Min Workload', and 'Max Workload'.

1. Click the down arrow on the folder level of AIDD
2. Click Import

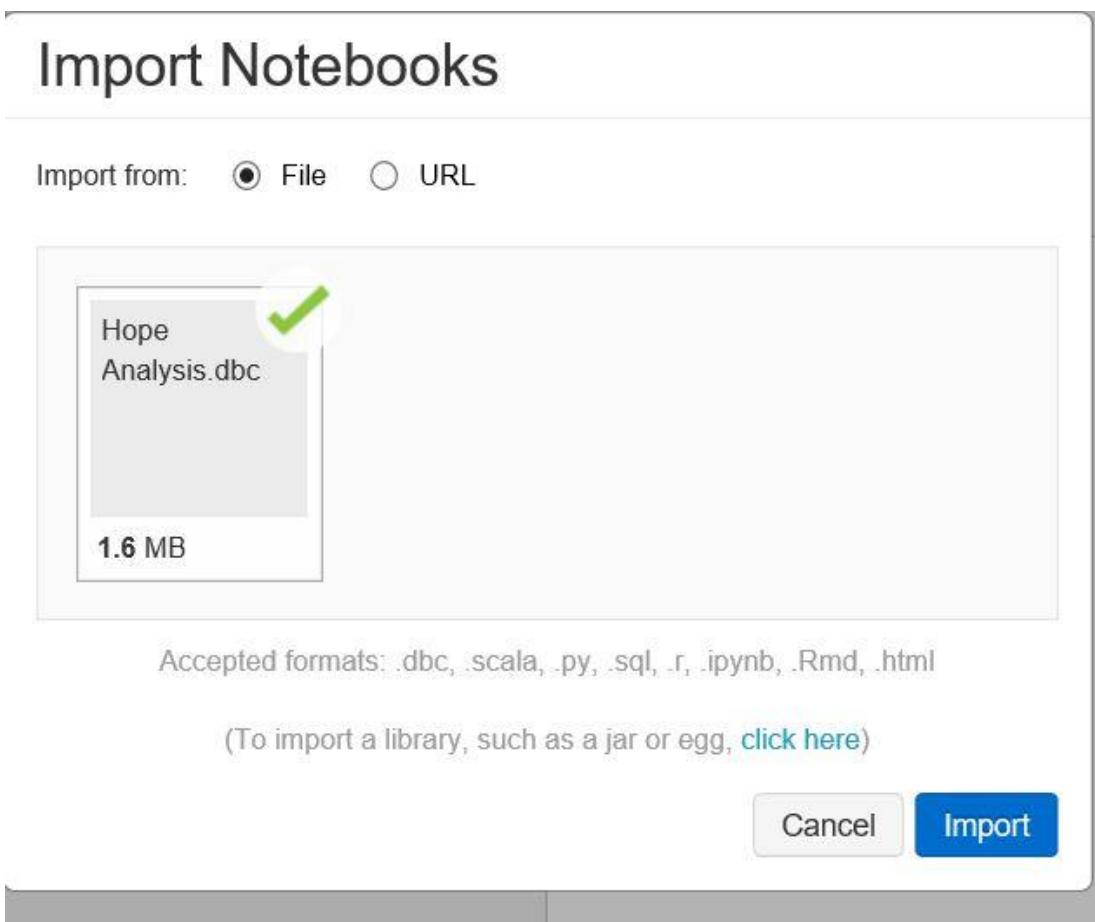
Select Notebook to Import



1. Instructor left leeway here for getting files to you



Import HopeAnalysis



1. Select HopeAnalysisdbc
2. Select Import

Validate Notebook Imported

Microsoft Azure

Azure Databricks

Home

Workspace

Recents

Data

Workspace

AIDD

Hope Analysis

- Beer Me - Setup...
- BeerMe - Mix It Up
- BeerMe - Python
- BeerMe - Workin...

1. Verify notebooks imported properly

Module 3 - Working in Notebook

Setting up Storage

Begin working in notebooks

Beer Me - Setup Storage (Scala)

Attached: HopeLandia ▾ File ▾ View: Code ▾ Permissions Run All Clear ▾

Cmd 1

```
1 %md # Setting Up The Storage to Cluster Using Blob Storage
```

Cmd 2

1. Define credentials

We defined storage account credentials at the cluster level to work with the database UI of Azure databricks Here

Cmd 3

```
//Here's what comments look like in Scala //We need to first go setup storage ac ...
```

Show cell

Cmd 4

```
1 //Make sure before moving on to the next cell that
2 //you have copied the BeerMeTweets.json file to the blob storage
```

Cmd 5

2. Work with blob directly

Cmd 6

```
1 val tweets = sqlContext.read.format("json")
2 .option("header", "true")
```

1. Select the “Beer Me – Setup Storage” notebook
2. Click on Show cell
3. Double click the first cell
4. Note the markdown language and %md (magic command)
5. Hit Shift + Enter while that command cell selected

Define storage in notebook

Beer Me - Setup Storage (Scala)

Attached: HopeLandia ▾ File ▾ View: Code ▾ Permissions Run All Clear ▾

Cmd 1

```
1 %md # Setting Up The Storage to Cluster Using Blob Storage
```

Cmd 2

1. Define credentials

We defined storage account credentials at the cluster level to work with the database UI of Azure databricks Here

Cmd 3

```
//Here's what comments look like in Scala //We need to first go setup storage ac ...
```

Show cell



Cmd 4

```
1 //Make sure before moving on to the next cell that
2 //you have copied the BeerMeTweets.json file to the blob storage
```

Cmd 5

2. Work with blob directly

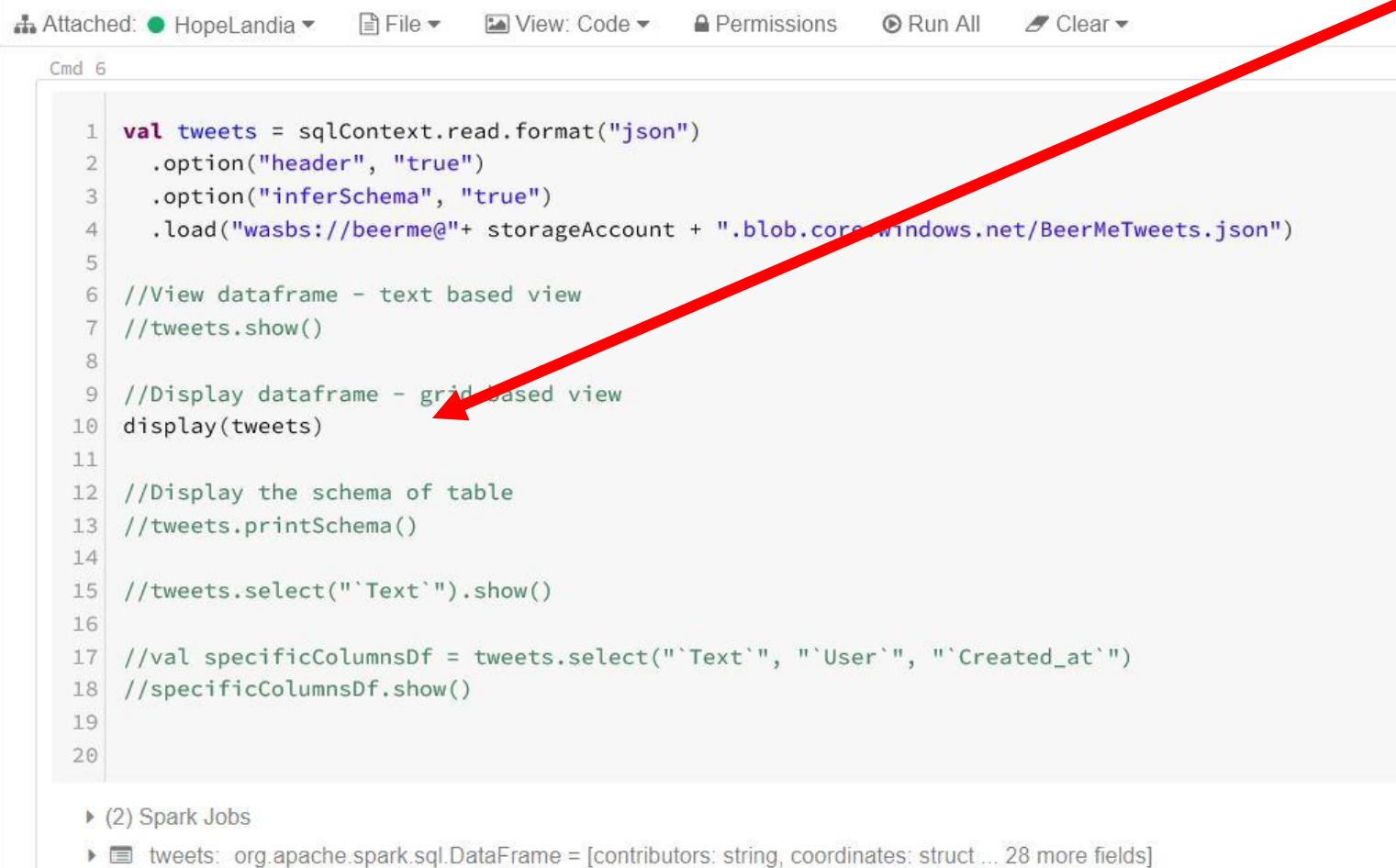
Cmd 6

```
1 val tweets = sqlContext.read.format("json")
2 .option("header", "true")
```

1. Click Show cell to expand command cell 3
2. Change the values for storage account to value for blob storage account noted earlier
3. Hit Shift + Enter while that command cell selected

Work with the data in blob storage directly

Beer Me - Setup Storage (Scala)



```
Attached: HopeLandia File View: Code Permissions Run All Clear
Cmd 6
1 val tweets = sqlContext.read.format("json")
2   .option("header", "true")
3   .option("inferSchema", "true")
4   .load("wasbs://beermefiles@storageAccount.blob.core.windows.net/BeerMeTweets.json")
5
6 //View dataframe - text based view
7 //tweets.show()
8
9 //Display dataframe - grid based view
10 display(tweets)
11
12 //Display the schema of table
13 //tweets.printSchema()
14
15 //tweets.select("`Text`").show()
16
17 //val specificColumnsDf = tweets.select(`Text`, `User`, `Created_at`)
18 //specificColumnsDf.show()
19
20
```

▶ (2) Spark Jobs

▶ tweets: org.apache.spark.sql.DataFrame = [contributors: string, coordinates: struct ... 28 more fields]

1. Scroll further down in the notebook to Cmd 6
2. Uncomment out the various functions
3. Hit Shift + Enter while that command cell selected
4. Look at various results back from dataframes

Create Mount to Blob Storage

Beer Me - Setup Storage (Scala)

Attached: HopeLandia File View: Code Permissions Run All Clear

Free trial ends in 14 days

Cmd 8

```
1 def mountStorageContainer(storageAccount: String, storageAccountKey: String, storageContainer: String, blobMountPoint: String)
2 {
3   val mountStatus = dbutils.fs.mount(
4     source = "wasbs://" + containername + "@" + storageAccount + ".blob.core.windows.net/",
5     mountPoint = blobMountPoint,
6     extraConfigs = Map("fs.azure.account.key." + storageAccount + ".blob.core.windows.net" -> storageAccountKey))
7
8   println("Status of mount of container " + containername + " is: " + mountStatus)
9 }
```

mountStorageContainer: (storageAccount: String, storageAccountKey: String, storageContainer: String, blobMountPoint: String)Unit

Command took 0.23 seconds -- by hopef@microsoft.com at 10/21/2018 5:22:14 PM on HopeLandia

Cmd 9

```
1 mountStorageContainer(storageAccount,storageAccountKey,"beerme","/mnt/data/beerme")
```

▶ (1) Spark Jobs

Status of mount of container beerme is: true

Command took 22.34 seconds -- by hopef@microsoft.com at 10/21/2018 5:22:37 PM on HopeLandia

Cmd 10

```
1 display(dbutils.fs.ls("/mnt/data"))
```

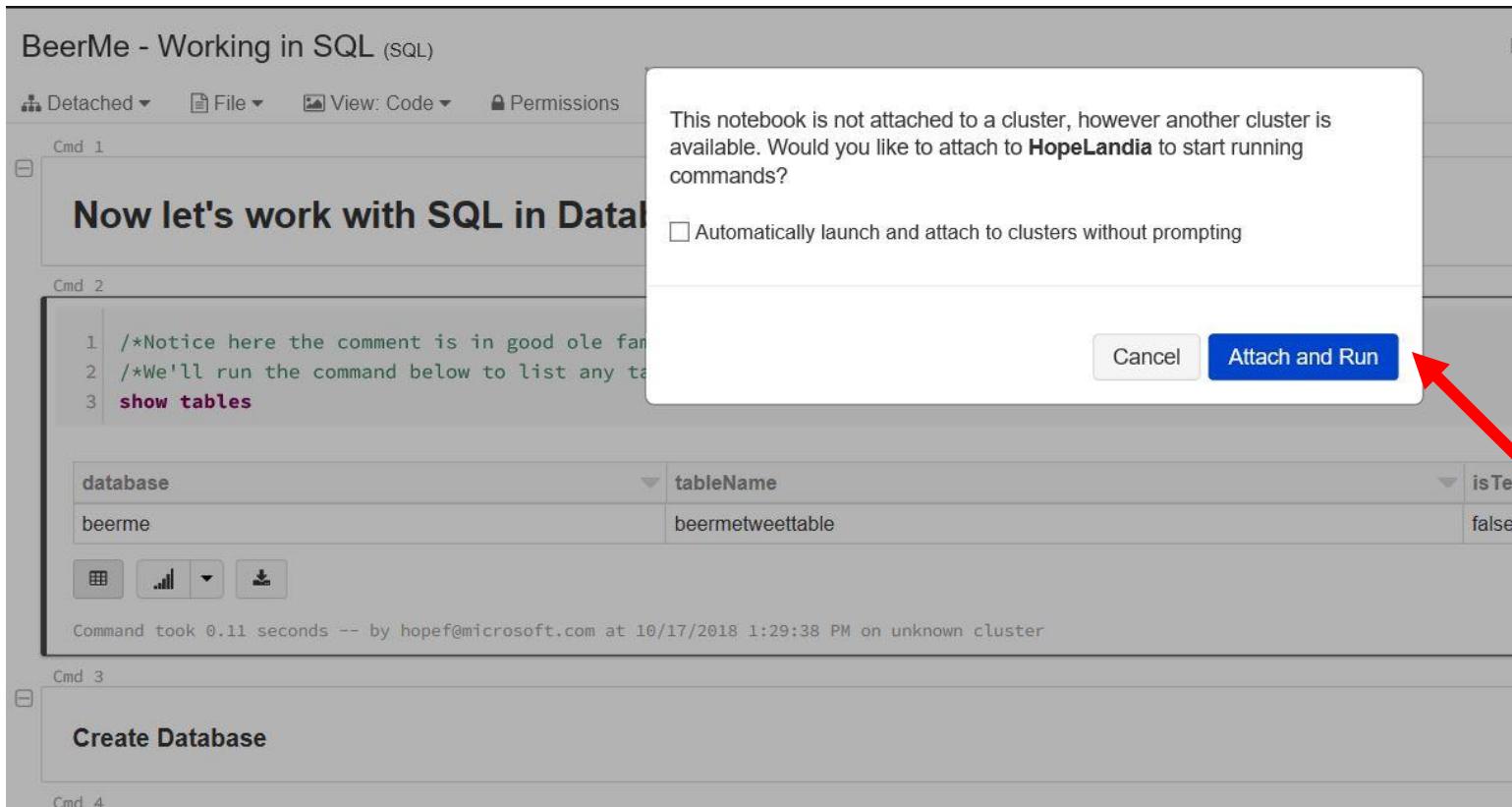
path	name
dbfs:/mnt/data/beerme/	beerme/

1. Scroll further down in the notebook to Cmd 8
2. Hit Shift + Enter while that command cell selected
3. Select Cmd 9 and hit Shift + Enter
4. Verify status of mount = true
5. Run Cmd 10 and note the paths (add /beerme)

Module 3 - Working in Notebook

Working with SQL

Change to BeerMe – Working in SQL



BeerMe - Working in SQL (SQL)

Detached File View: Code Permissions

Cmd 1

Now let's work with SQL in Data

Cmd 2

```
1 /*Notice here the comment is in good ole fashioned sql
2 /*We'll run the command below to list any tables in the database
3 show tables
```

database tableName isTemp

beerme	beermetweetable	false
--------	-----------------	-------

Command took 0.11 seconds -- by hopef@microsoft.com at 10/17/2018 1:29:38 PM on unknown cluster

Cmd 3

Create Database

Cmd 4

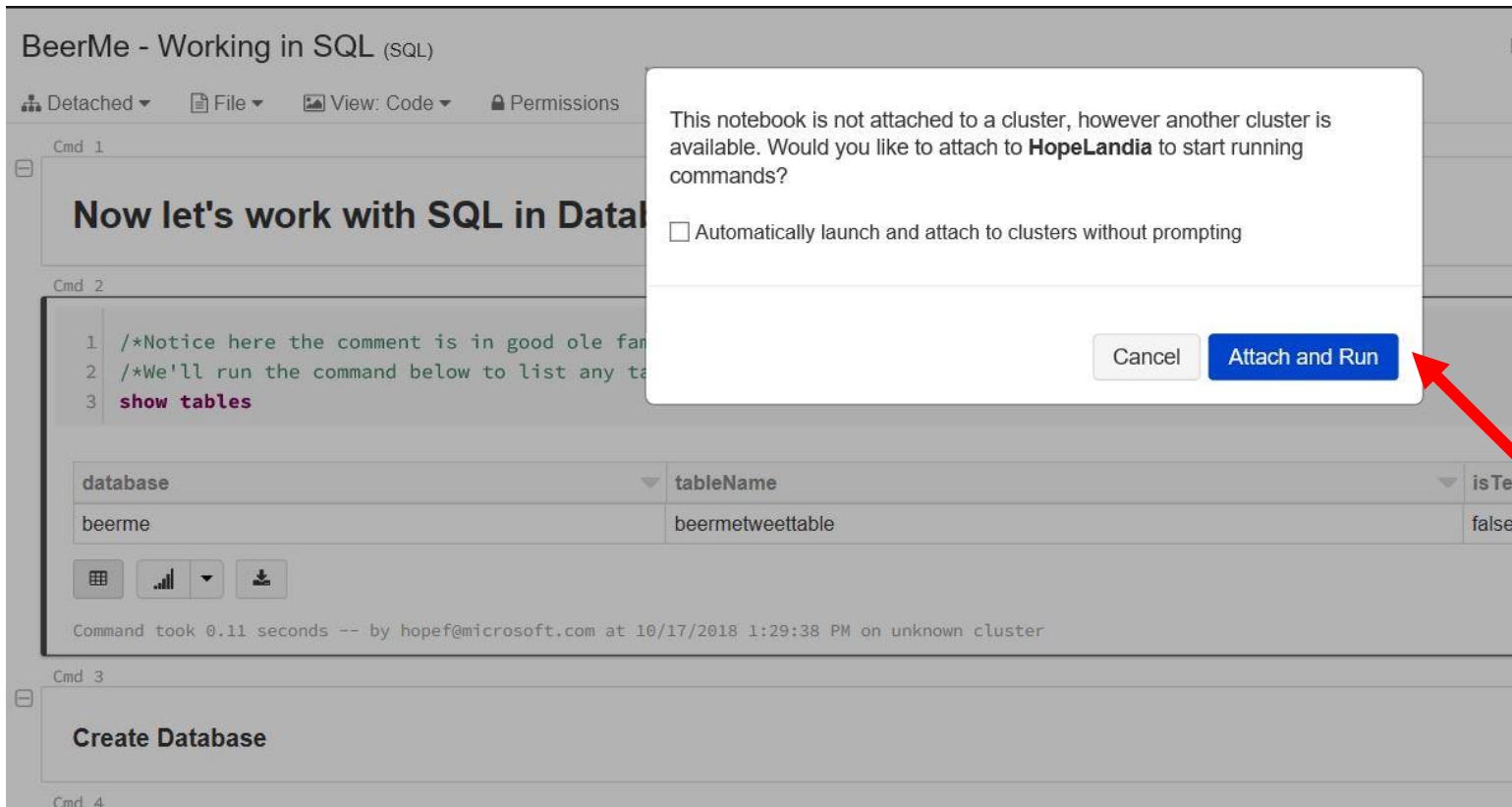
This notebook is not attached to a cluster, however another cluster is available. Would you like to attach to **HopeLandia** to start running commands?

Automatically launch and attach to clusters without prompting

Cancel Attach and Run

1. Navigate to the notebook “BeerME – Working in SQL”
2. Select Cmd 2 and hit Shift + Enter
3. Note that it is not attached. Select Attach & Run

Attach a notebook to cluster



The screenshot shows a Jupyter Notebook interface titled "BeerMe - Working in SQL (SQL)". The notebook has a single cell containing the following SQL command:

```
1 /*Notice here the comment is in good ole fashioned sql
2 /*We'll run the command below to list any tables in the database
3 show tables
```

Below the cell, a table displays the results of the `show tables` command:

database	tableName	isTemp
beerme	beermetweetable	false

At the bottom of the cell, the message "Command took 0.11 seconds -- by hopef@microsoft.com at 10/17/2018 1:29:38 PM on unknown cluster" is shown.

A modal dialog box is overlaid on the interface, asking: "This notebook is not attached to a cluster, however another cluster is available. Would you like to attach to **HopeLandia** to start running commands?". It contains a checkbox for "Automatically launch and attach to clusters without prompting" and two buttons: "Cancel" and "Attach and Run". A red arrow points to the "Attach and Run" button.

1. Note that it is not attached. Select Attach & Run
2. No tables yet to see

Create Database

BeerMe - Working in SQL (SQL)

Attached: HopeLandia ▾ File ▾ View: Code ▾ Permissions Run All Clear

Cmd 1

Now let's work with SQL in Databricks

Cmd 2

```
1 /*Notice here the comment is in good ole familiar SQL syntax */
2 /*We'll run the command below to list any tables in your cluster*/
3 show tables
```

OK

Command took 0.10 seconds -- by hopef@microsoft.com at 10/21/2018 5:33:49 PM on HopeLandia

Cmd 3

Create Database

Cmd 4

```
1 /*
2 CREATE DATABASE BeerMe
3
4 */
5 SHOW DATABASES
```

databaseName

beerme

default

grid icon, chart icon, download icon

1. Move to Cmd 4
2. Move commands in and out of the comments to create the database “BeerMe”.
3. Run show databases command to validate creation.

Create View

BeerMe - Working in SQL (SQL)

Attached: HopeLandia File View: Code Permissions Run All Clear

databaseName
beerme
default

Command took 0.03 seconds -- by hopef@microsoft.com at 10/21/2018 5:35:58 PM on HopeLandia

Cmd 5

```
1 /* Let's create a temp view
2 CREATE TEMPORARY VIEW BeerMeTweetsRaw
3 USING JSON
4 OPTIONS (
5   path "/mnt/data/beerme/BeerMeTweets.json"
6 )
7
8 select * from BeerMeTweetsRaw
9
10 show tables
11
12 DROP VIEW BeerMeTweetsRaw
13 */
14
15 select * from BeerMeTweetsRaw
16
```

▶ (1) Spark Jobs

contributors	coordinates	created_at	entities
null	null	Wed Oct 17 14:33:27 +0000 2018	▶ {"hashtags": [{"indices": [0, 12], "text": "picoftheday"}, {"indices": [102, 106], "text": "hbo"}], "media": null, "symbols": [], "urls": [{"id": 1423540536, "id_str": "1423540536", "indices": [16, 26], "nar": null}]} ▶ {"indices": [16, 26], "text": "picoftheday"} ▶ {"indices": [102, 106], "text": "hbo"} ▶ {"id": 1423540536, "id_str": "1423540536", "indices": [16, 26], "nar": null}

1. Move to Cmd 5
2. Move commands in and out of the comments to create the temporary view “BeerMeTweetsRaw”.
3. Run show databases command to validate creation.

Create Table

BeerMe - Working in SQL (SQL)

Attached: ● HopeLandia ▾ File ▾ View: Code ▾ Permissions Run All Clear ▾
Command took 1.47 seconds -- by noper@microsofc.com at 10/21/2018 5:42:24 PM on HopeLandia

Cmd 6

```
1 /* Ok now lets create a persisted table */
2 /*
3 CREATE TABLE beerme.BeerMeTweetTable
4 USING JSON
5 OPTIONS (path "/mnt/data/beerme/BeerMeTweets.json", header "true", inferSchema "true")
6
7 select * from beermetweettable
8
9
10 show tables
11 */
12 select * from beerme.beermetweettable
```

▶ (1) Spark Jobs

contributors	coordinates	created_at	entities
null	null	Wed Oct 17 14:33:27 +0000 2018	▶ {"hashtags": [{"indices": [0, 12], "text": "picoftheday"}, {"indices": [30, 46], "indices": [102, 106], "text": "hbo"}], "media": null, "symbols": [], "urls": [{"display_url": "https://t.co/1423540536", "id": 1423540536, "id_str": "1423540536", "indices": [16, 26], "name": "Rile"}]}
null	null	Wed Oct 17 14:25:30 +0000 2018	▶ {"hashtags": [], "media": null, "symbols": [], "urls": [{"display_url": "twitter.com", "indices": [0, 23], "id": 102915111111111111, "id_str": "102915111111111111", "name": "Twitter"}]}

Showing the first 528 rows.

1. Move to Cmd 6
2. Move commands in and out of the comments to create the table “beermetweettable”.
3. Run commands in cell to validate creation.
4. Run command in Cmd 7 to further validate

Begin Getting Data Useful

BeerMe - Working in SQL (SQL)

Free trial ends in 14 days. Upgrade to P

space/Users/hopef@microsoft.com/AIDD/Hopeysis/BeerMe - Working in SQL

View: Code Permissions Run All Clear

database tableName isTemporary

database	tableName	isTemporary
beerme	beermetweetable	false
	beermetweetsraw	true

Error running command 8. Go Clear all notifications | Settings

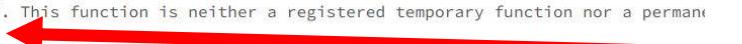
Command took 0.06 seconds -- by hopef@microsoft.com at 10/21/2018 6:00:26 PM on HopeLandia

Cmd 8

```
1 SELECT created_at, id_str, text,
2 JSON_VALUE(user, '$.followers_count') AS Followers
3 FROM beerme.beermetweetable
4
```

✉ Error in SQL statement: AnalysisException: Undefined function: 'JSON_VALUE'. This function is neither a registered temporary function nor a permanent function in database 'beerme'.; line 2 pos 2

Command took 0.20 seconds -- by hopef@microsoft.com at 10/21/2018 6:03:02 PM on HopeLandia



1. Move to Cmd 8
2. Run Shift + Enter to attempt to break down the field User which is json format
3. Note the error

Select what we need

BeerMe - Working in SQL (SQL)

Free trial ends in 14 days. [Upgrade to Premium](#) in Azure Portal

space/Users/hopef@microsoft.com/AIDD/Hope

View: Code ▾ Permissions Run All Clear

Error running command 8. Go to command

Clear all notifications | Settings

```
2 JSON_VALUE(user, '$.followers_count') AS Followers
3 FROM beerme.beermetweetable
4
```

Error in SQL statement: AnalysisException: Undefined function: 'JSON_VALUE'. This function is neither a registered temporary function nor a permanent function registered in the database 'beerme'.; line 2 pos 2

Command took 0.20 seconds -- by hopef@microsoft.com at 10/21/2018 6:03:02 PM on HopeLandia

Cmd 9

```
1 select id_str, text, created_at, user.screen_name, user.followers_count
2 FROM beerme.beermetweetable
```

(1) Spark Jobs

id_str	text	created_at	screen_name	followers_count
1052568274217684992	#picoftheday by @laxbabe13 is #motherofdragons from @breweryommegang I #ommegang #gameofthrones #beer #hbo... https://t.co/OAfXYn4fDb	Wed Oct 17 14:33:27 +0000 2018	ShorePointDist	2428
1052566275610816513	No I'm not drinking at this time of day! Doing some analysis and need data and I ALWAYS have pics of beer. Last wee... https://t.co/Pdua3lNqNk	Wed Oct 17 14:25:30 +0000 2018	hope_foley	2080
1052563435483852800	RT @Essau1969: #TopBrewsTues The Shape of #Hops to Come #Imperial #IPA #Beeroclock 😊#BeerMe 😊X2 #jwagsapprovedABV 👉@BoroughBrew #DrinkLocal...	Wed Oct 17 14:14:13 +0000 2018	Maria41240084	1069
1052558801419821056	#newbeeralert from @alaskanbrewing Cranberry Tart #beer #beergeek #beerlovers #beernerdy #beerme #beerbeerbeer... https://t.co/8IQgVnJxfQ	Wed Oct 17 13:55:48 +0000 2018	gp_beerandwine	140
1052558542262153216	#newbeeralert from @lagunitasbeer Mozango - Mosaic hopped ale with Mango #beer #beergeek #beerlovers #beernerdy... https://t.co/jn42gDd0oa	Wed Oct 17 13:54:46 +0000 2018	gp_beerandwine	140

grid chart download

1. Move to Cmd 9
2. Run Shift + Enter
3. Note the expanded results of user utilizing “user.column”

Module 3 - Working in Notebook

Mix It Up

Change to BeerMe – Mix It Up

BeerMe - Working in SQL (SQL)

Detached File View: Code Permissions

Cmd 1

Now let's work with SQL in Data

Cmd 2

```
1 /*Notice here the comment is in good ole fashioned sql
2 /*We'll run the command below to list any tables in the database
3 show tables
```

database tableName isTemp

database	tableName	isTemp
beerme	beermetweetable	false

Command took 0.11 seconds -- by hopef@microsoft.com at 10/17/2018 1:29:38 PM on unknown cluster

Cmd 3

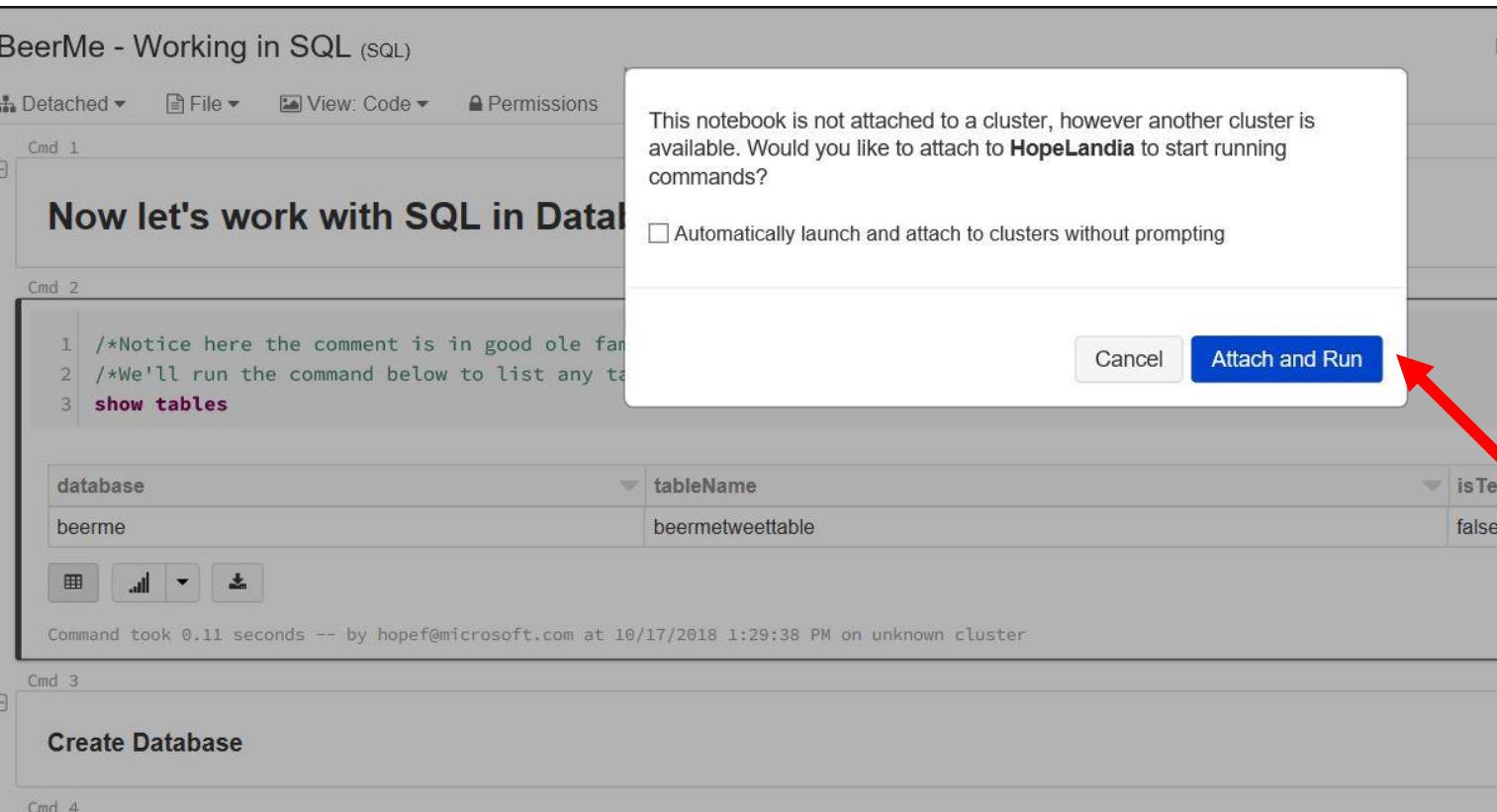
Create Database

Cmd 4

This notebook is not attached to a cluster, however another cluster is available. Would you like to attach to **HopeLandia** to start running commands?

Automatically launch and attach to clusters without prompting

Cancel **Attach and Run**



1. Navigate to the notebook “BeerME – Mix It Up”
2. Note the notebook type is SQL again
3. Select Cmd 2
4. Hit Shift + Enter to run cell

Module 4 - Visualizations

Inside the notebook

Scala Results Again

BeerMe - Mix It Up (SQL)

Attached: HopeLandia ▾ File ▾ View: Code ▾ Permissions ▾ Run All ▾ Clear ▾

Cmd 3

```
1 %scala
2 val tweetssql = spark.read.json("/mnt/data/beerme/BeerMeTweets.json")
3 val specificColumnsDf = tweetssql.select(`Text`, `User`, `Created_at`)
4 display(specificColumnsDf)
5
```

▶ (2) Spark Jobs

▶ tweetssql: org.apache.spark.sql.DataFrame = [contributors: string, coordinates: struct ... 28 more fields]

▶ specificColumnsDf: org.apache.spark.sql.DataFrame = [Text: string, User: struct ... 1 more fields]

Text	User
#picoftheday by @laxbabe13 is #motherofdragons from @breweryommegang ! #ommegang #gameofthrones #beer #hbo... https://t.co/OAfXYn4fDb	▶ {"contributors_enabled":false,"created_at":"Thu Nov 24 02:54:33 +0000 2011","default_profile":false,"de Distributing","notifications":null,"profile_background_color":"C0DEED","profile_background_image_url":"t Foley","notifications":null,"profile_background_color":"C0DEED","profile_background_image_url":"http://e https://t.co/Pdua3INqnk
No I'm not drinking at this time of day! Doing some analysis and need data and I ALWAYS have pics of beer. Last wee... https://t.co/Pdua3INqnk	▶ {"contributors_enabled":false,"created_at":"Wed Feb 20 15:20:57 +0000 2010","default_profile":true,"de Foley","notifications":null,"profile_background_color":"C0DEED","profile_background_image_url":"http://e https://t.co/Pdua3INqnk

Command took 1.59 seconds -- by hopef@microsoft.com at 10/21/2018 6:23:26 PM on HopeLandia

1. Select Cmd 3
2. Hit Shift + Enter to run cell
3. Note the results in grid
4. Select to change results to bar chart
5. Hit plot options

Python Viz

BeerMe - Mix It Up (SQL)

Free trial ends in 14 days. [Upgrade to Premium](#) in /

Attached: ● HopeLandia ▾

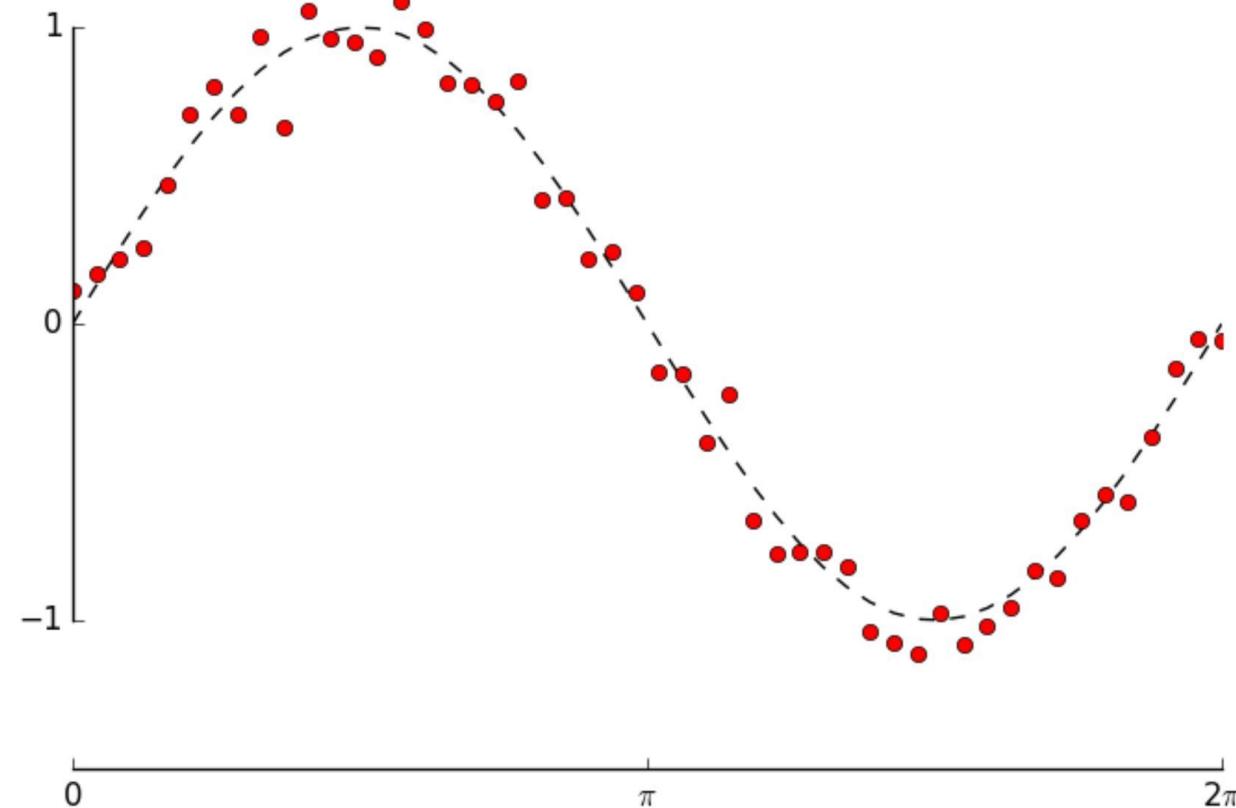
File ▾

View: Code ▾

Permissions

Run All

Clear ▾



1. Select Cmd 4
2. Hit Shift + Enter to run cell
3. Note the visualization in python

R Viz

BeerMe - Mix It Up (SQL)

Free trial ends in 14 days. [Upgrade to Premium](#) in Azure Portal

Attached:  HopeLandia

 File

 View: Code

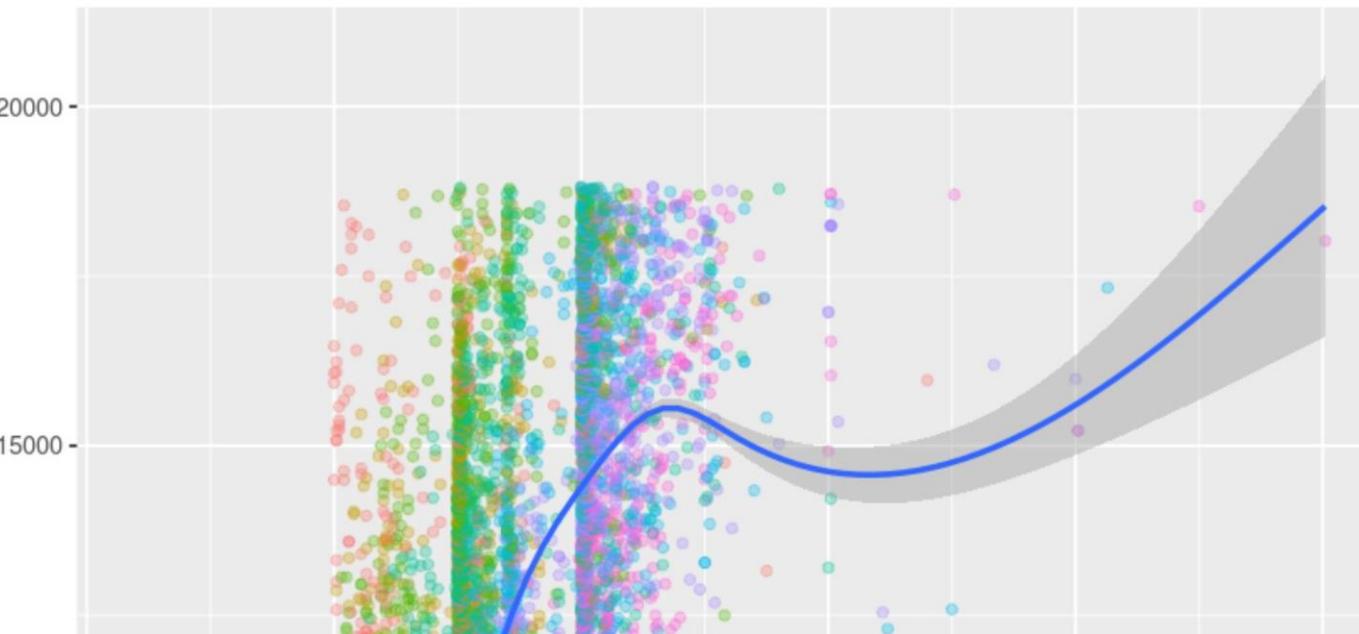
 Permissions

 Run All

 Clear

```
1 %r
2 library(ggplot2)
3 ggplot(diamonds, aes(carat, price, color = color, group = 1)) + geom_point(alpha = 0.3) + stat_smooth()
4
```

1. Select Cmd 5
2. Hit Shift + Enter to run cell
3. Note the visualization in R



Other Magic Commands

BeerMe - Mix It Up (SQL) Free trial ends in 14 days. [Upgrade](#)

Attached: HopeLandia File View: Code Permissions Run All Clear

Cmd 6

```
1 %sh wget -P /tmp
https://raw.githubusercontent.com/Azure/usql/master/Examples/Samples/Data/json/radio
--2018-10-18 21:21:23-- https://raw.githubusercontent.com/Azure/usql/master/Examples/San
ll_radio_json.json
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.32.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.32.133|:443...
HTTP request sent, awaiting response... 200 OK
Length: 8476 (8.3K) [text/plain]
Saving to: '/tmp/small_radio_json.json'

OK .....
100% 116M=0s

2018-10-18 21:21:23 (116 MB/s) - '/tmp/small_radio_json.json' saved [8476/8476]

Command took 0.18 seconds -- by hopef@microsoft.com at 10/18/2018 5:21:23 PM on unknown cluster
```

Cmd 7

```
1 %fs ls /mnt/data/beerme/
```

path	name
dbfs:/mnt/data/beerme/BeerMeTweets.json	BeerMeTweets.json

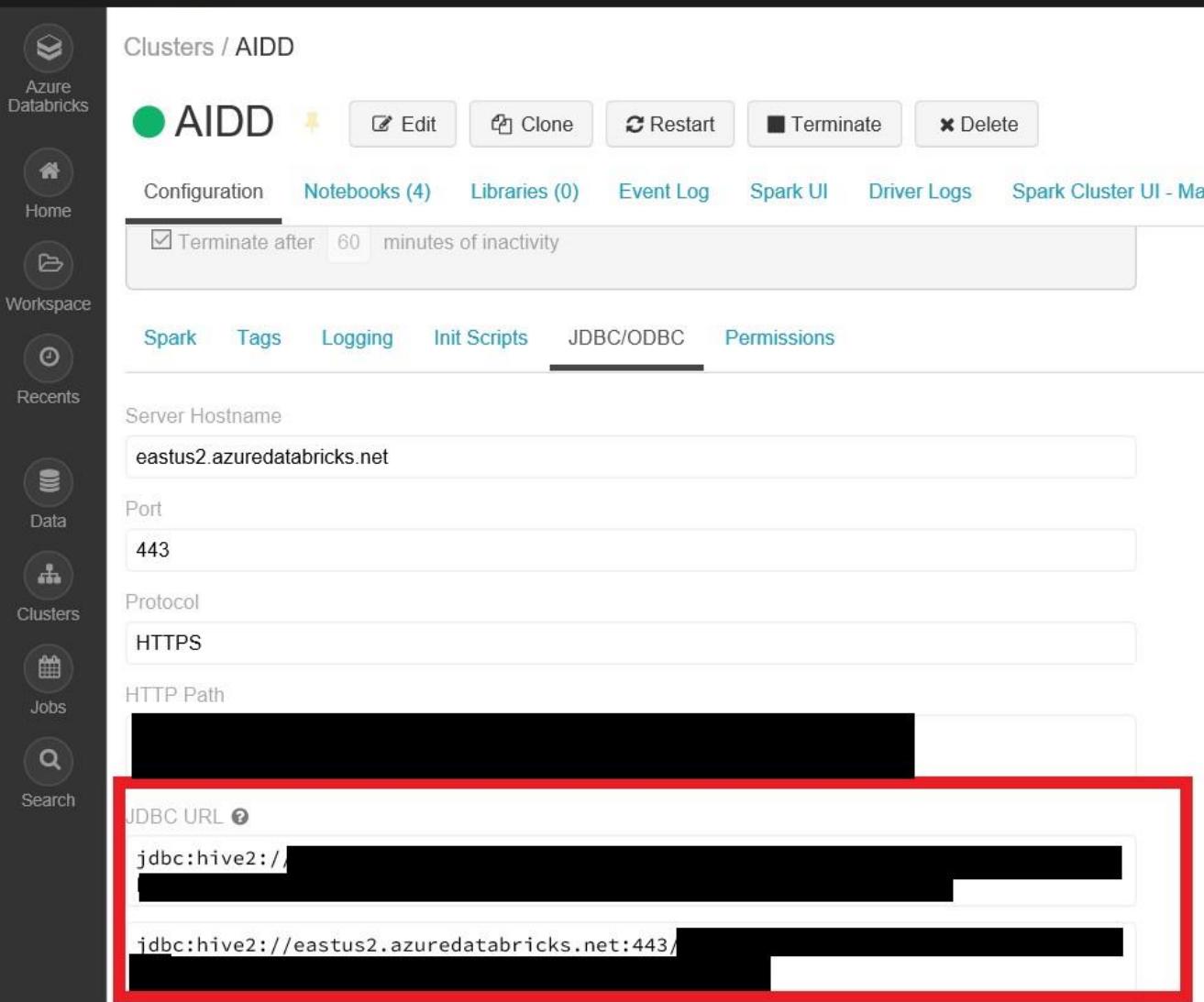
1. Explore the rest of the notebook and other magic commands

Module 4 - Visualizations

Connect Power BI

Get JDBC URL

Microsoft Azure



Clusters / AIDD

AIDD

Edit Clone Restart Terminate Delete

Configuration Notebooks (4) Libraries (0) Event Log Spark UI Driver Logs Spark Cluster UI - Mas

Terminate after 60 minutes of inactivity

Spark Tags Logging Init Scripts JDBC/ODBC Permissions

Server Hostname: eastus2.azuredatabricks.net

Port: 443

Protocol: HTTPS

HTTP Path: [REDACTED]

JDBC URL: `jdbc:hive2://[REDACTED]`
`idbc:hive2://eastus2.azuredatabricks.net:443/[REDACTED]`

1. Grab the JDBC url

Change the connection string

So we'll start out with something like this:

```
jdbc:hive2://eastus2.azuredatabricks.net:443/default;transportMode=http;ssl=true;httpPath=sql/protocolv1/o/5441075421675247/boost
```

Change the connection string

Remove and replace jdbc:hive2 portion with https

jdbc:hive2://eastus2.azuredatabricks.net:443/default;transportMode=http;ssl=true;httpPath=sql/protocolv1/o/5441075421675247/boost

Change the connection string

Remove and replace jdbc:hive2 portion with https

https://eastus2.azuredatabricks.net:443/default;transportMode=http;ssl=true;httpPath=sql/protocolv1/o/5441075421675247/boost

Change the connection string

Remove the portion below (everything between port # and sql)

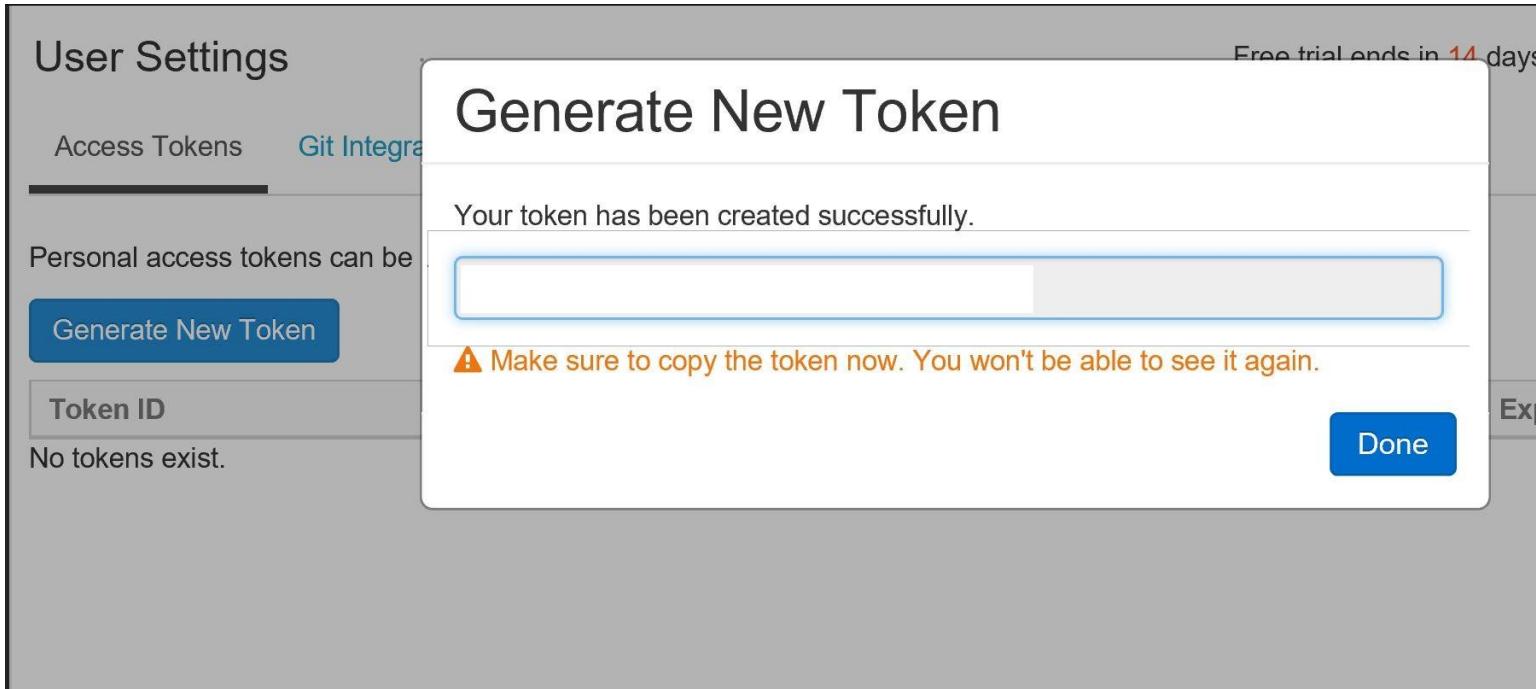
https://eastus2.azuredatabricks.net:443/**default;transportMode**
=http;ssl=true;httpPath=sql/protocolv1/o/5441075421675247
/boost

Change the connection string

You should have something like this now

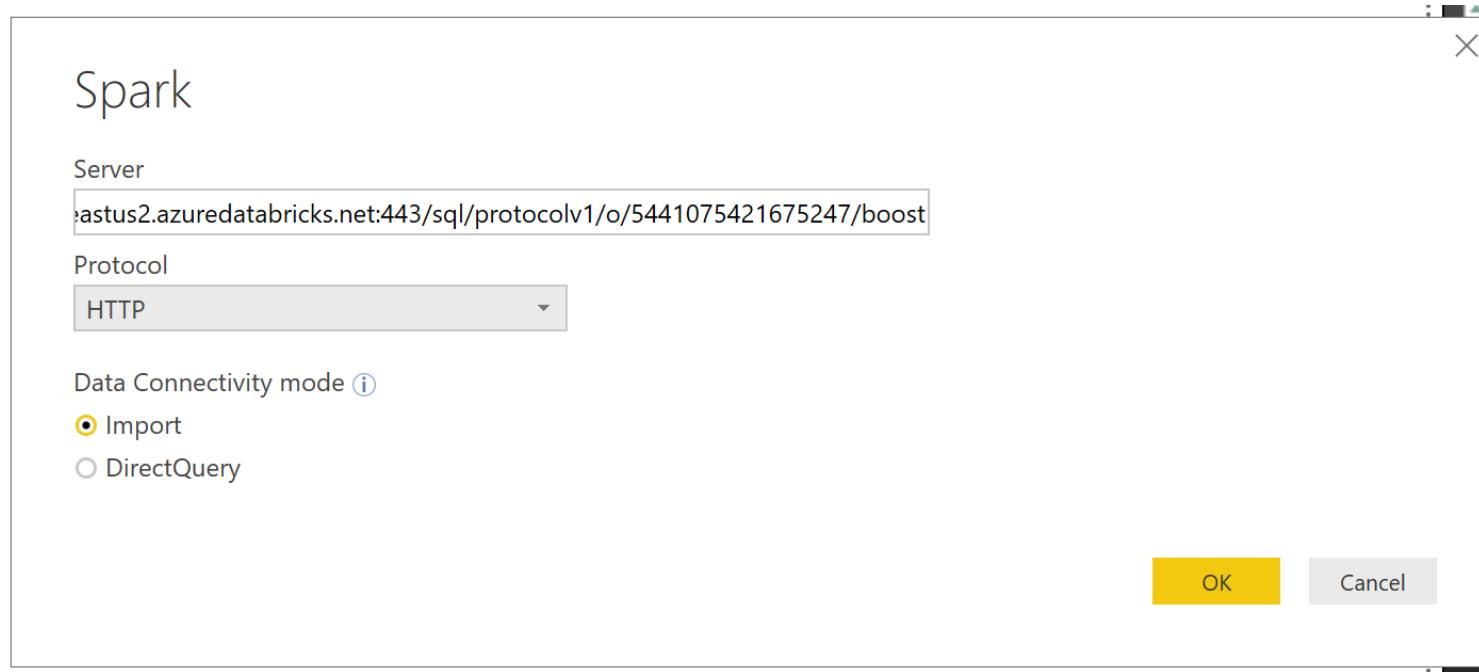
`https://eastus2.azuredatabricks.net:443/sql/protocolv1/o/544107
5421675247/boost`

Get access token



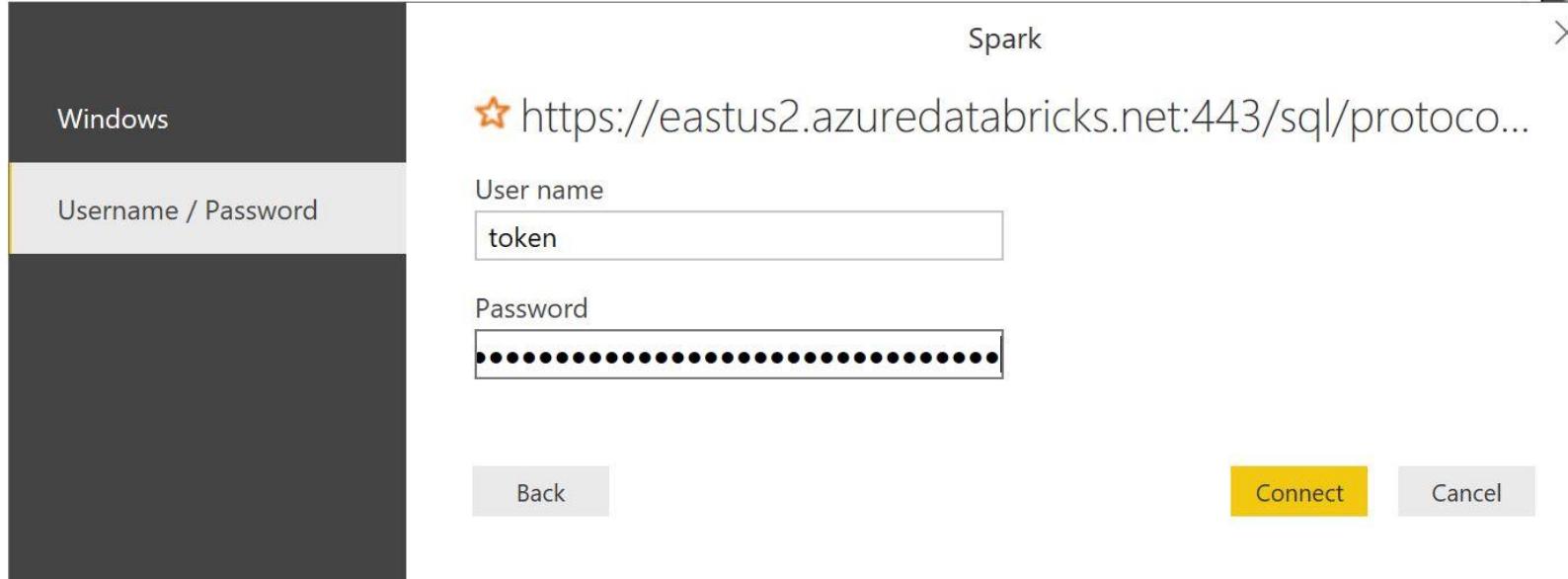
1. Click user profile in top right
2. Select User Settings
3. Click Generate New Token
4. Enter comment and lifetime
5. Click Generate
6. Capture the access token

Open Power BI Desktop



1. Open Power BI Desktop
2. Choose Get Data
3. Choose Spark
4. Plug in connection string and choose options shown

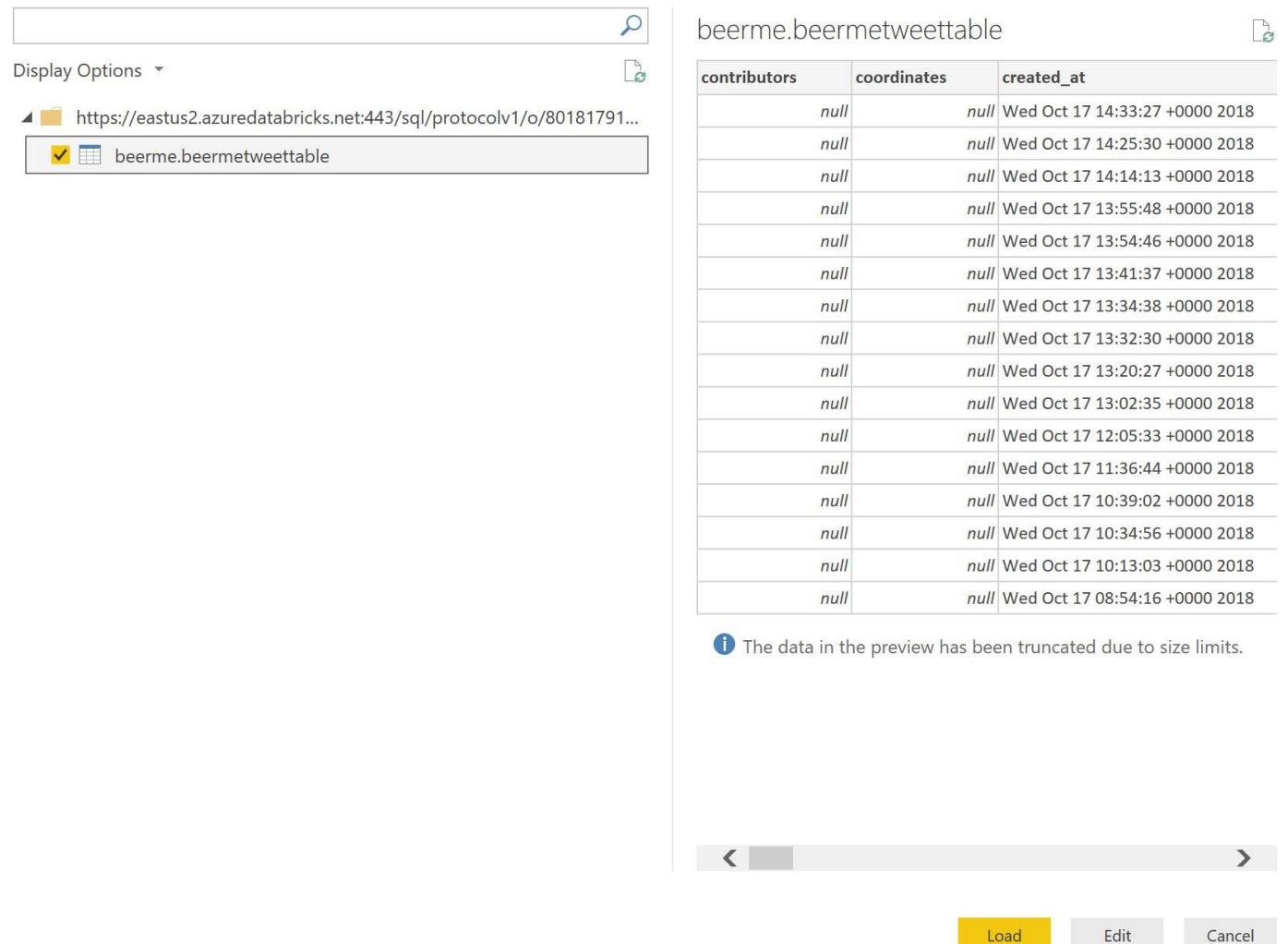
Open Power BI Desktop



1. Supply user name "token"
2. Plug in the access token in the password field
3. Click Connect

Open Power BI Desktop

Navigator



The screenshot shows the Power BI Desktop interface with the 'Navigator' tab selected. On the left, the 'Navigator' pane displays a list of data sources, with 'beerme.beermetweetable' selected. The main area shows a preview of the 'beerme.beermetweetable' table, which has three columns: 'contributors', 'coordinates', and 'created_at'. All rows in the preview show 'null' for 'contributors' and 'coordinates', and various dates in 2018 for 'created_at'. A note at the bottom indicates that the data is truncated due to size limits. At the bottom of the preview area are 'Load', 'Edit', and 'Cancel' buttons.

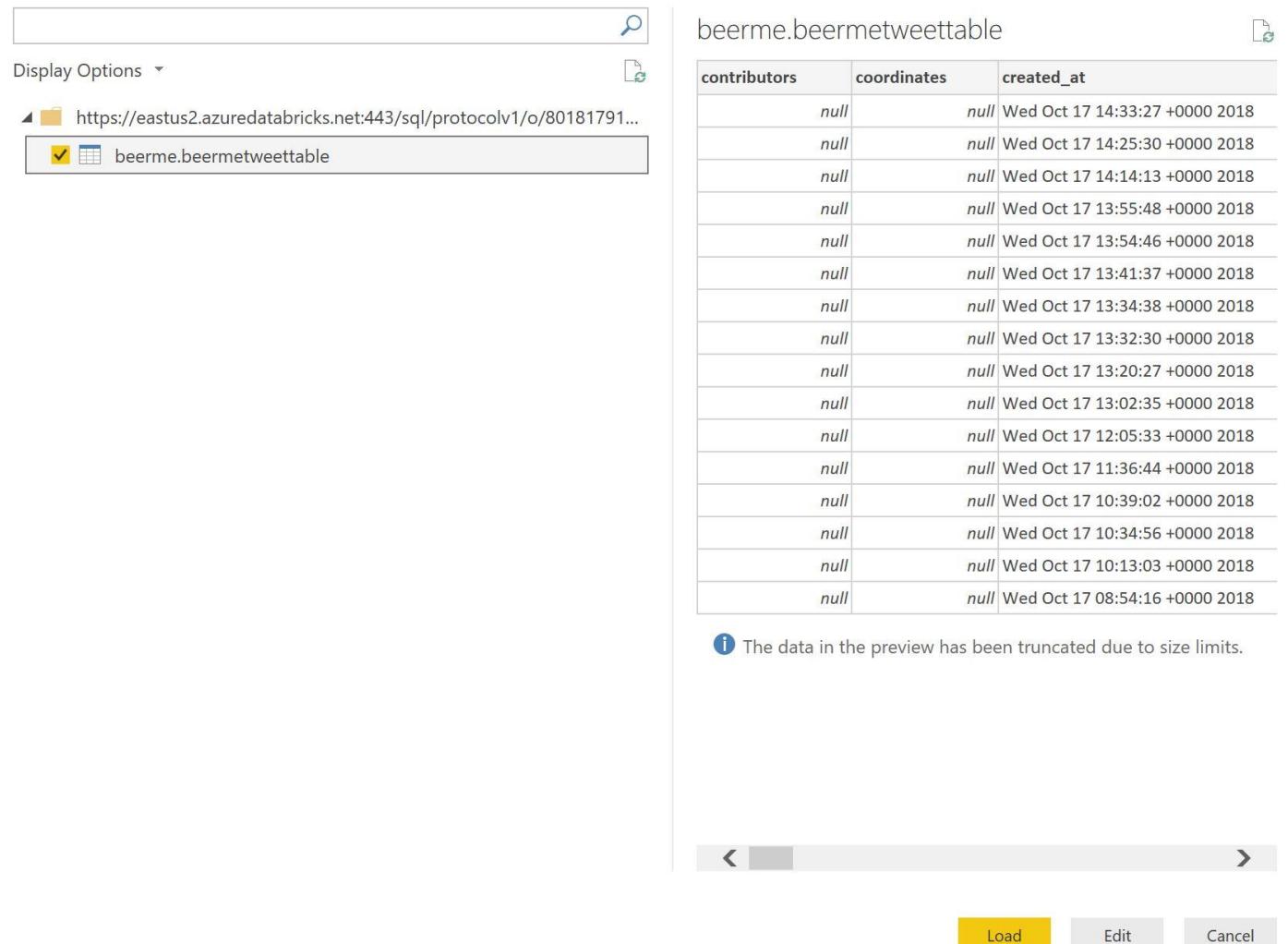
contributors	coordinates	created_at
null	null	Wed Oct 17 14:33:27 +0000 2018
null	null	Wed Oct 17 14:25:30 +0000 2018
null	null	Wed Oct 17 14:14:13 +0000 2018
null	null	Wed Oct 17 13:55:48 +0000 2018
null	null	Wed Oct 17 13:54:46 +0000 2018
null	null	Wed Oct 17 13:41:37 +0000 2018
null	null	Wed Oct 17 13:34:38 +0000 2018
null	null	Wed Oct 17 13:32:30 +0000 2018
null	null	Wed Oct 17 13:20:27 +0000 2018
null	null	Wed Oct 17 13:02:35 +0000 2018
null	null	Wed Oct 17 12:05:33 +0000 2018
null	null	Wed Oct 17 11:36:44 +0000 2018
null	null	Wed Oct 17 10:39:02 +0000 2018
null	null	Wed Oct 17 10:34:56 +0000 2018
null	null	Wed Oct 17 10:13:03 +0000 2018
null	null	Wed Oct 17 08:54:16 +0000 2018

i The data in the preview has been truncated due to size limits.

1. Select the beermetweetable
2. Select Load

Open Power BI Desktop

Navigator



The screenshot shows the Power BI Desktop Navigator interface. On the left, there's a tree view of data sources. The 'beerme.beermetweetable' table is selected, indicated by a checked checkbox. The main area displays a preview of the table with columns: 'contributors', 'coordinates', and 'created_at'. All previewed rows in the 'contributors' and 'coordinates' columns show 'null'. The 'created_at' column shows dates from October 17, 2018, at various times. A note at the bottom of the preview area states: 'The data in the preview has been truncated due to size limits.' At the bottom of the preview area, there are navigation arrows and buttons for 'Load', 'Edit', and 'Cancel'.

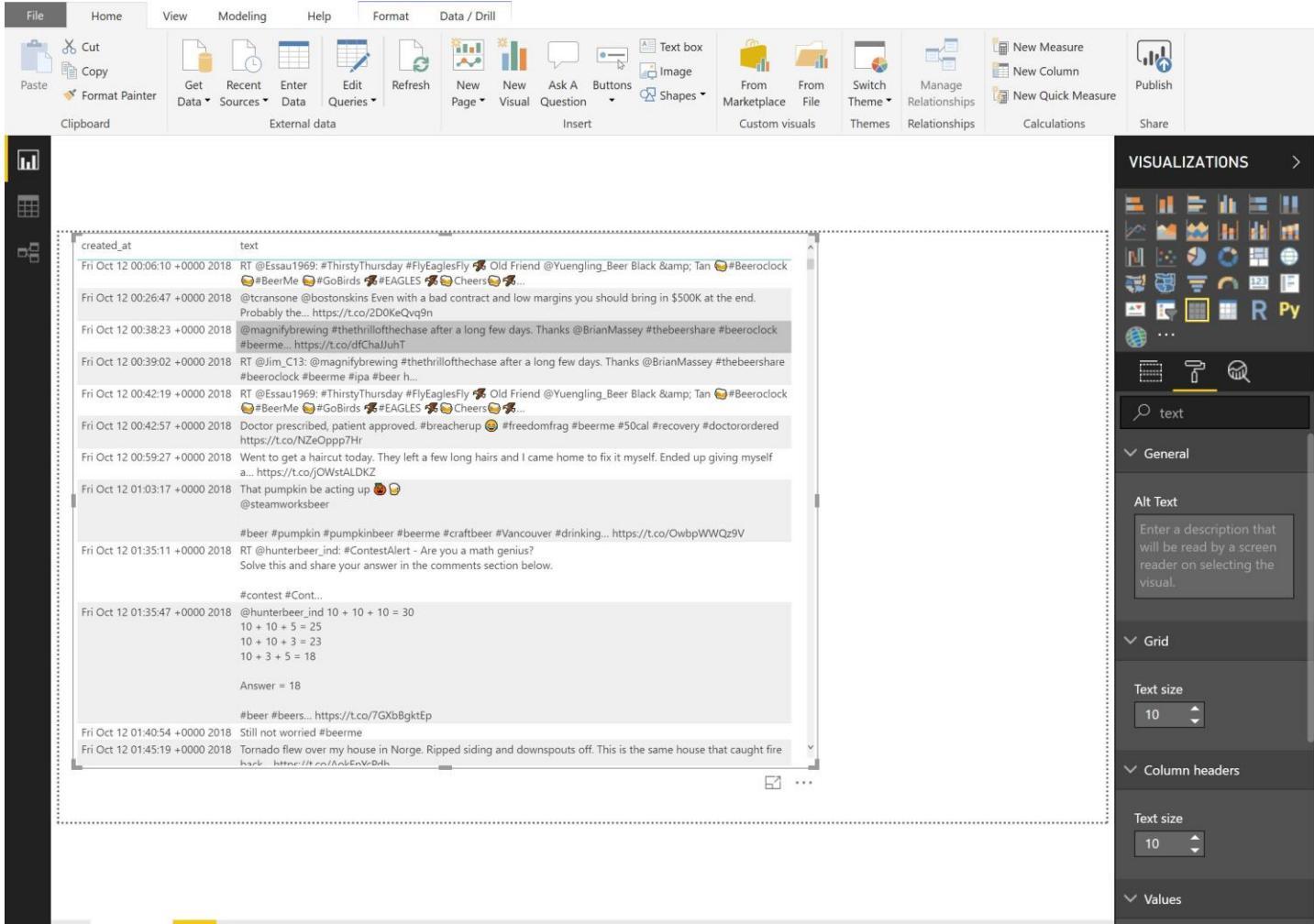
contributors	coordinates	created_at
null	null	Wed Oct 17 14:33:27 +0000 2018
null	null	Wed Oct 17 14:25:30 +0000 2018
null	null	Wed Oct 17 14:14:13 +0000 2018
null	null	Wed Oct 17 13:55:48 +0000 2018
null	null	Wed Oct 17 13:54:46 +0000 2018
null	null	Wed Oct 17 13:41:37 +0000 2018
null	null	Wed Oct 17 13:34:38 +0000 2018
null	null	Wed Oct 17 13:32:30 +0000 2018
null	null	Wed Oct 17 13:20:27 +0000 2018
null	null	Wed Oct 17 13:02:35 +0000 2018
null	null	Wed Oct 17 12:05:33 +0000 2018
null	null	Wed Oct 17 11:36:44 +0000 2018
null	null	Wed Oct 17 10:39:02 +0000 2018
null	null	Wed Oct 17 10:34:56 +0000 2018
null	null	Wed Oct 17 10:13:03 +0000 2018
null	null	Wed Oct 17 08:54:16 +0000 2018

The data in the preview has been truncated due to size limits.

Load Edit Cancel

1. Select the beermetweetable
2. Select Load

Open Power BI Desktop



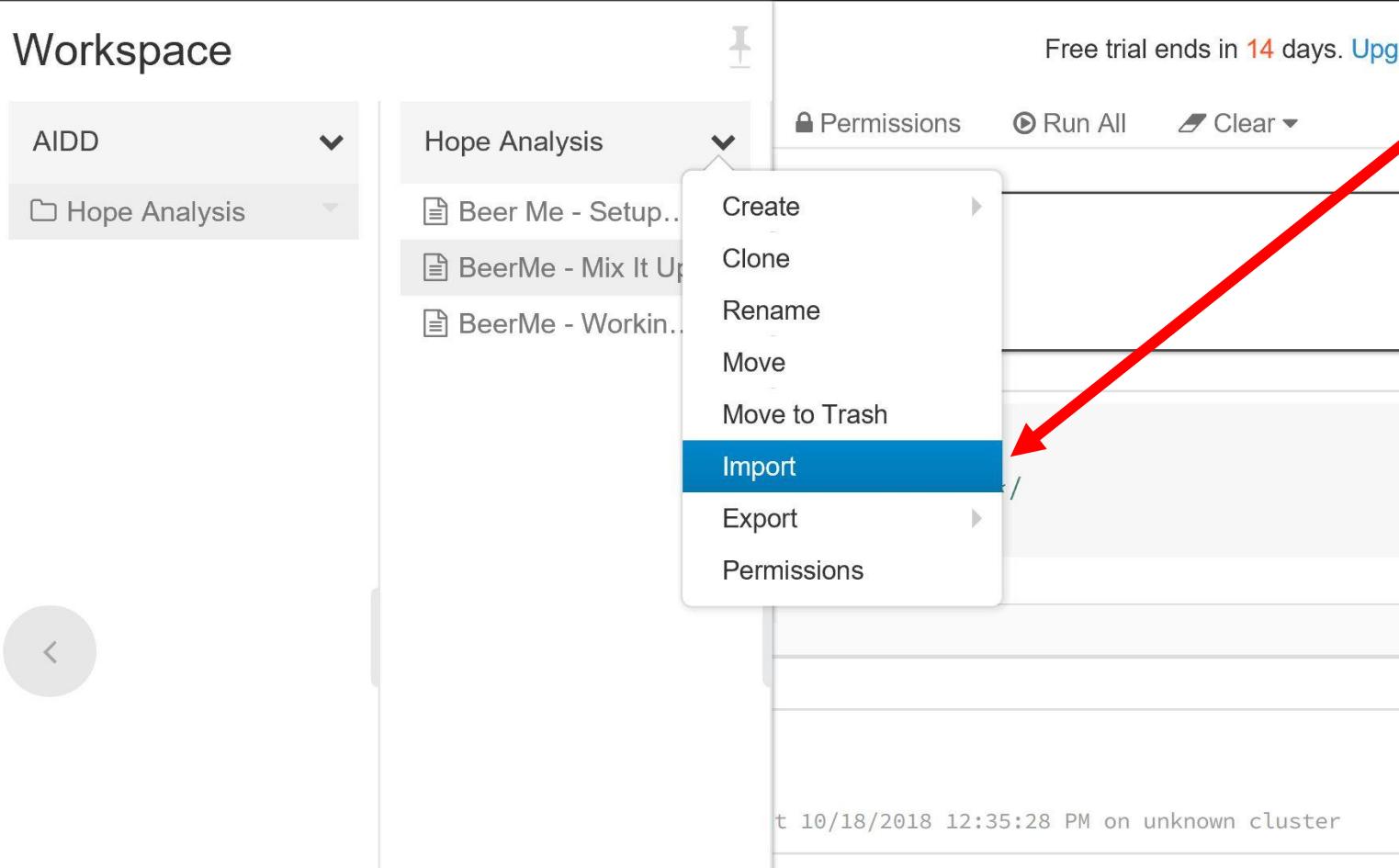
1. Begin creating visuals in the report

Module 5 - Bonus

Import Notebook/ML

Import ML Notebook

Workspace



A screenshot of the Databricks workspace interface. On the left, there is a sidebar with a dropdown menu set to 'AIDD' and a 'Hope Analysis' folder listed under 'Hope Analysis'. The main workspace shows a 'Hope Analysis' folder expanded, containing sub-notebooks like 'Beer Me - Setup...', 'BeerMe - Mix It Up...', and 'BeerMe - Workin...'. A context menu is open over the 'Hope Analysis' folder, with the 'Import' option highlighted in blue and a red arrow pointing to it. The menu also includes options like 'Create', 'Clone', 'Rename', 'Move', 'Move to Trash', 'Export', and 'Permissions'. At the top of the workspace, there is a message about a free trial ending in 14 days, and buttons for 'Permissions', 'Run All', and 'Clear'.

1. Select the down arrow next to Hope Analysis folder
2. Select Import
3. Select URL
4. Paste url below:
[https://github.com/JackXueIndia/Databricks-TensorFlow-LSTM-Predictive-Maintenance/blob/master/LSTM-AERO ENG TEST Databricks TF.ipynb](https://github.com/JackXueIndia/Databricks-TensorFlow-LSTM-Predictive-Maintenance/blob/master/LSTM-AERO%20ENG%20TEST%20Databricks%20TF.ipynb)
5. Click Import
6. Review contents

Wrap up

Q&A

Stuff didn't have time to dive into...

Read up on these things next

1. Jobs
2. Libraries
3. Databricks IO
4. SparkML
5. Streaming
6. GraphX
7. CLI
8. Databricks Utilities
9. Databricks REST API
- 10.

References

1. Quickstart to Setup Databricks to ADLS Gen 2 -
<https://docs.microsoft.com/en-us/azure/storage/data-lake-storage/quickstart-create-databricks-account#create-storage-account-file-system>
2. Tutorial: Extract, transform, and load data using Azure Databricks -
<https://docs.microsoft.com/en-us/azure/storage/data-lake-storage/handle-data-using-databricks#upload-data-to-the-storage-account>
3. Generate access token -
<https://docs.databricks.com/api/latest/authentication.html#token-management>
4. Connect Power BI to Databricks Cluster - <https://docs.databricks.com/user-guide/bi/power-bi.html#connect-power-bi-desktop-to-a-databricks-cluster>



© 2017 Microsoft Corporation. All rights reserved. Microsoft, Windows, and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries. The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation. **MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.**