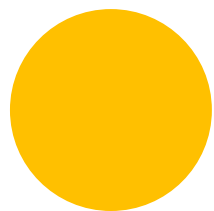
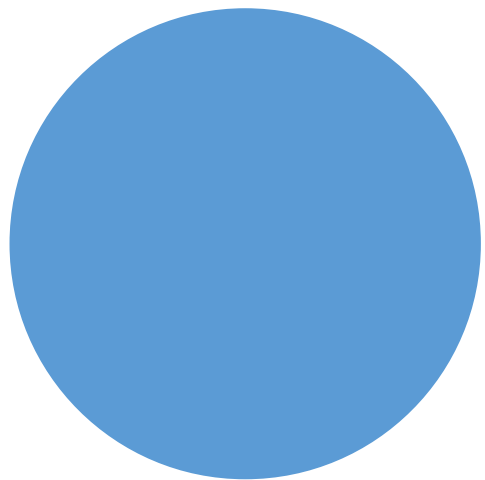


Workshop Cloud Experts

« Vos analyses de Machine Learning avec Azure Databricks »

Mercredi 20 mars 2019





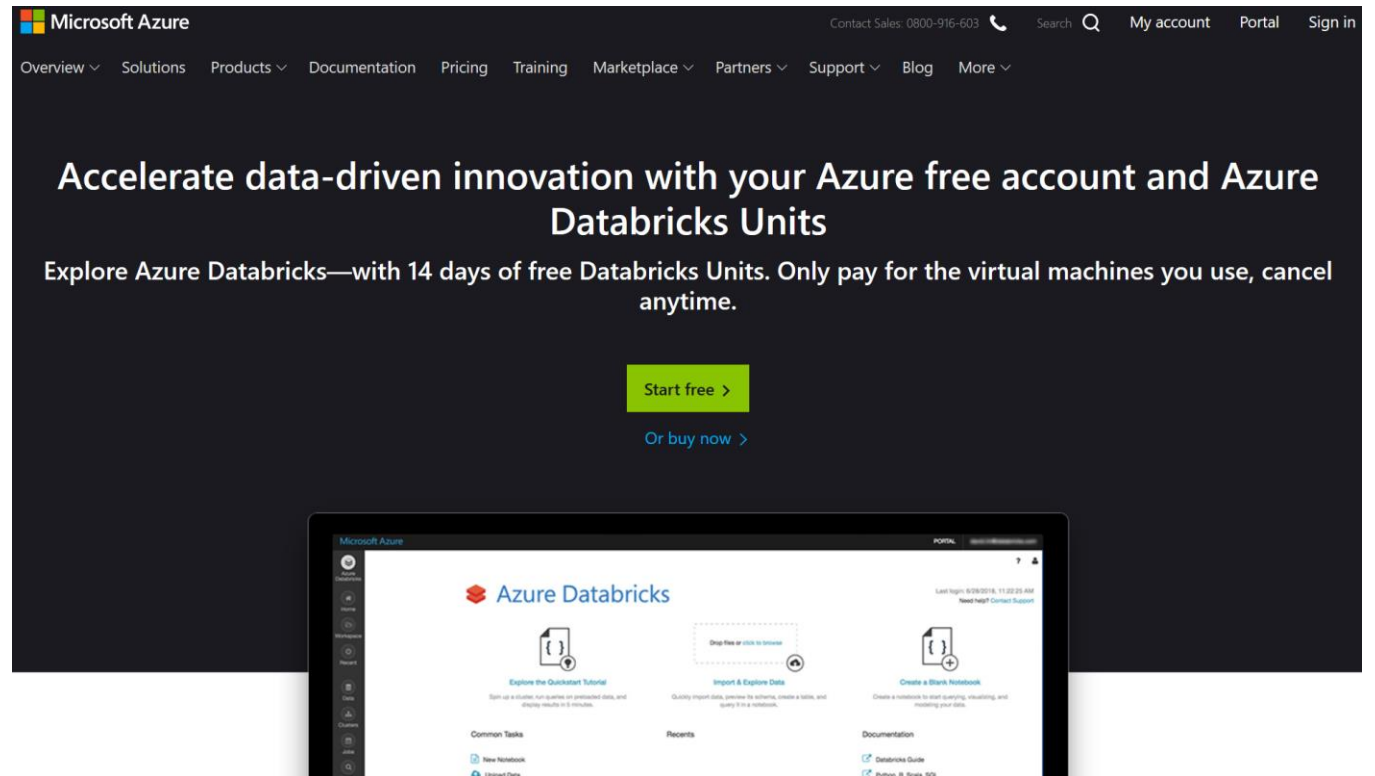
Création d'un compte Azure

Azure Databricks

[Already using Azure? try Azure Databricks now](#) or

create a [free Azure account to start using Azure Databricks](#)

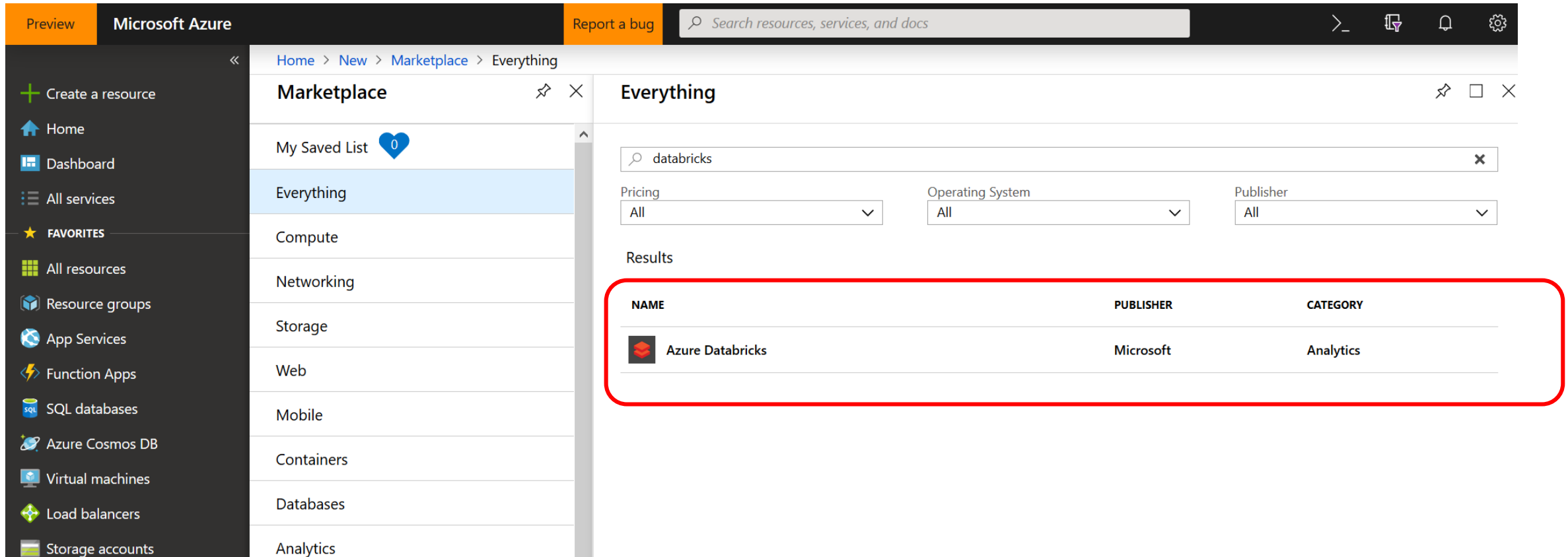
<https://azure.microsoft.com/en-us/free/services/databricks/>





Provisionnement service Azure Databricks

1. Création du service Azure Databricks



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 contains navigation links for 'Create a resource', 'Home', 'Dashboard', 'All services', and 'FAVORITES'. The main content area displays the 'Marketplace' section with a search for 'databricks'. The search results show a single entry for 'Azure Databricks' by Microsoft, categorized under 'Analytics'. The entry is highlighted with a red box.

Home > New > Marketplace > Everything

Marketplace

My Saved List 0

Everything

Compute

Networking

Storage

Web

Mobile

Containers

Databases

Analytics

Everything


Search: databricks

Pricing: All

Operating System: All

Publisher: All

Results

NAME	PUBLISHER	CATEGORY
 Azure Databricks	Microsoft	Analytics

1. Création du service Azure Databricks

The screenshot displays the Azure Databricks service overview page. The left-hand navigation pane includes the 'Azure Databricks' header, a search bar, and a list of settings such as 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Virtual Network Peerings', 'Locks', 'Automation script', and 'Support + troubleshooting'. The main content area features a 'Delete' button, a 'Managed Resource Group' section with fields for 'Resource group (change)', 'Subscription (change)', 'Subscription ID', 'URL', and 'Pricing Tier', and a large 'Launch Workspace' button. Below these are several quick-start links: 'Documentation', 'Getting Started', 'Import Data from File', 'Import Data from Azure Storage', 'Notebook', and 'Admin Guide'.

Dashboard > Azure Databricks > Databricks

Azure Databricks Microsoft

+ Add Edit columns More

Filter by name...

NAME ↑↓

Databricks

Databricks
Azure Databricks Service

Search (Ctrl+/)

Overview

Activity log

Access control (IAM)

Tags

Settings

Virtual Network Peerings

Locks

Automation script

Support + troubleshooting

New support request

Delete

Resource group (change)
AzureDB

Subscription (change)
Microsoft Azure Internal Consumption

Subscription ID

Managed Resource Group

URL
<https://westeurope.azuredatabricks.net>

Pricing Tier
premium

Launch Workspace

Documentation

Getting Started

Import Data from File

Import Data from Azure Storage

Notebook

Admin Guide

2. Création du Cluster Spark Databricks

Clusters / Cluster HC 5.2

Cluster HC 5.2



Edit

Clone

Restart

Terminate

Delete

Configuration

Notebooks (2)

Libraries (3)

Event Log

Spark UI

Driver Logs

Spark Cluster UI - Master

Cluster Mode

Standard

Databricks Runtime Version

5.2 (includes Apache Spark 2.4.0, Scala 2.11)

Python Version

3

Autopilot Options

☒ Enable autoscaling

☒ Terminate after 120 minutes of inactivity

Worker Type

Standard_DS3_v2

14.0 GB Memory, 4 Cores, 0.75 DBU

Min Workers

2

Max Workers

8

Driver Type

Standard_DS3_v2


14.0 GB Memory, 4 Cores, 0.75 DBU


Advanced Options


Bien vérifier que les options *Enable autoscaling* et *Terminate after X minutes of activity* soient activées.

Cela permet de mettre en place *l'autoscaling* entre les *workers* du cluster Spark et de désactiver automatiquement le cluster en cas d'inactivité spécifiée par un timeout.

2. Création du Cluster Spark Databricks

 Azure Databricks

 Home


 Workspace

Clusters

+ Create Cluster

▼ Interactive Clusters

AllCreated by meAccessible by meFilter

Name	State	Nodes	Driver	Worker	Runtime	Creator		
 Cluster HC 5.2	Running	3	Standard_DS3_v2	Standard_DS3_v2	5.2 (includes Apach...	seretkow@micros...	2	3

3. Importation des librairies Azure ML service dans Databricks

Microsoft Azure

Azure Databricks

Home

Workspace

Recents

Data

Clusters

Jobs

Search

Create Library

Library Source

Upload DBFS **PyPI** Maven CRAN

Repository ?

Optional

Package








azureml-sdk[databricks]

Create Cancel



4. Association des librairies Azure ML service au cluster Databricks




Microsoft Azure

Clusters / Cluster HC 5.2

Cluster HC 5.2    Edit  Clone  Restart  Terminate  Delete

[Configuration](#) [Notebooks \(2\)](#) [Libraries \(3\)](#) [Event Log](#) [Spark UI](#) [Driver Logs](#) [Spark Cluster UI - Master ▼](#)

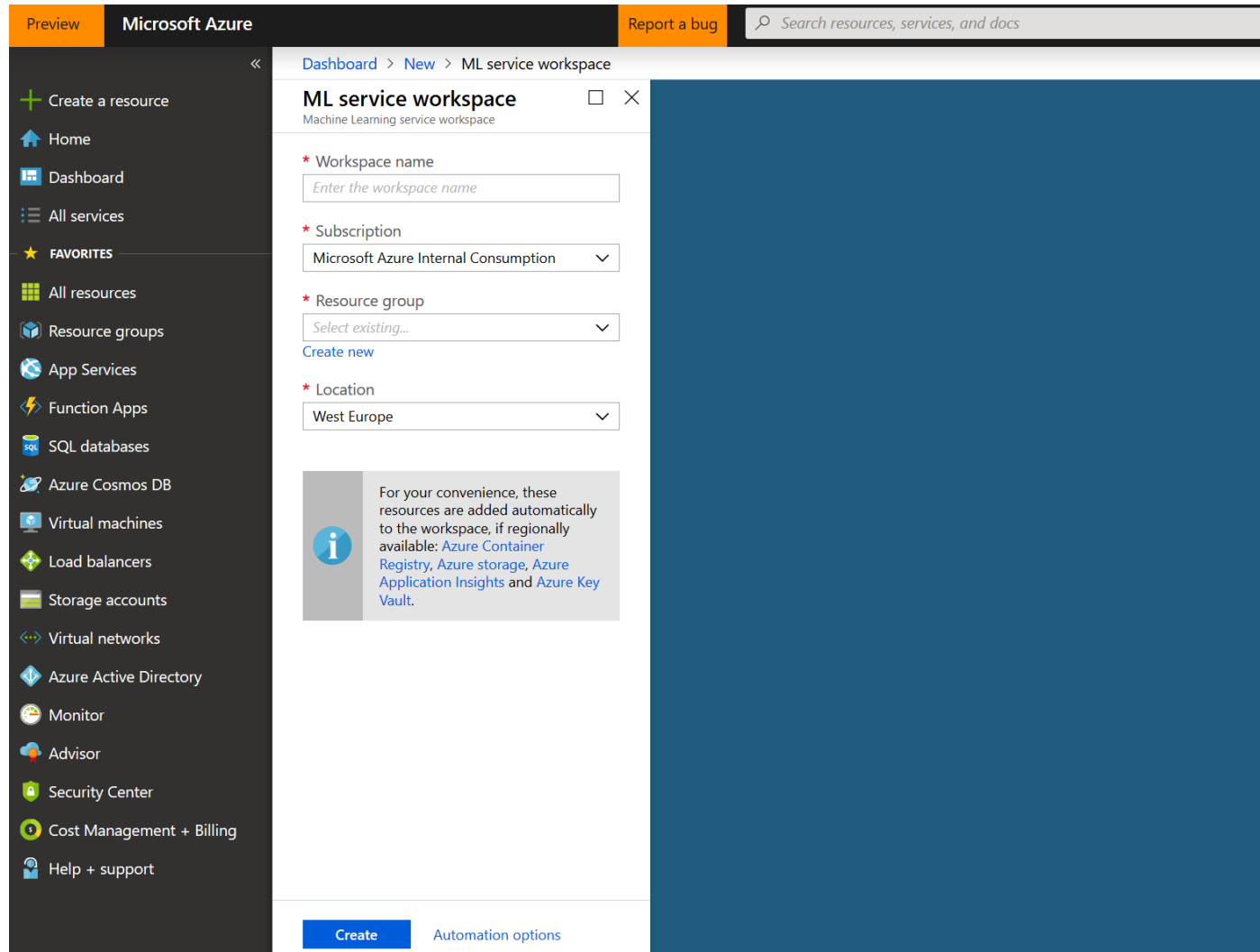
 Uninstall  Install New

<input type="checkbox"/>	Name	Type	Status	Source
<input type="checkbox"/>	azureml-sdk[automl_databricks]	PyPI	 Installed	
<input type="checkbox"/>	azureml-sdk[databricks]	PyPI	 Installed	
<input type="checkbox"/>	tqdm	PyPI	 Installed	



Provisionnement Azure ML service

5. Provisionnement du service Azure ML service



The screenshot shows the Azure portal interface for creating a new ML service workspace. The left sidebar contains navigation links such as 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES', 'All resources', 'Resource groups', 'App Services', 'Function Apps', 'SQL databases', 'Azure Cosmos DB', 'Virtual machines', 'Load balancers', 'Storage accounts', 'Virtual networks', 'Azure Active Directory', 'Monitor', 'Advisor', 'Security Center', 'Cost Management + Billing', and 'Help + support'. The main content area is titled 'ML service workspace' and includes the following fields:

- Workspace name:** A text input field with the placeholder 'Enter the workspace name'.
- Subscription:** A dropdown menu currently showing 'Microsoft Azure Internal Consumption'.
- Resource group:** A dropdown menu with 'Select existing...' and a link to 'Create new'.
- Location:** A dropdown menu currently showing 'West Europe'.

Below these fields, an information box states: 'For your convenience, these resources are added automatically to the workspace, if regionally available: [Azure Container Registry](#), [Azure storage](#), [Azure Application Insights](#) and [Azure Key Vault](#).' At the bottom, there is a blue 'Create' button and a link for 'Automation options'.

Une fois le service provisionné il sera nécessaire de retenir le nom du workspace, la localisation, le ressource groupe ainsi que l'ID du service.

Ces valeurs seront nécessaires pour assurer une connexion au workspace Azure ML service depuis un notebook Azure Databricks.



Exercices du lab



Exercices

Notebooks & données

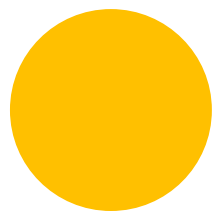
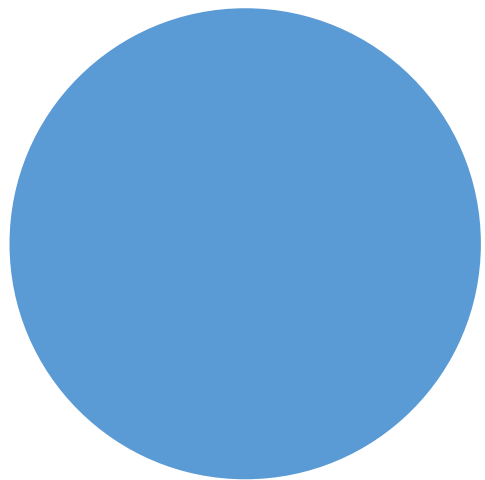
Les données sont dans Data.zip

Le Fichier .DBC est une archive Databricks qui contient tous les notebooks.

The screenshot shows the GitHub repository page for 'retkowsky / Cloud_Workshop_AzureDatabricks'. The repository has 2 commits, 1 branch, 0 releases, and 1 contributor. The 'Code' tab is selected, showing a list of files. The files include 14 HTML notebooks, a .dbc file, a README.md, and an index.html. Each file has an 'Add files via upload' button and a timestamp indicating when it was last committed.

File Name	Action	Time
01. Installation et Configuration.html	Add files via upload	6 minutes ago
02. Introduction Apache Spark avec Databricks.html	Add files via upload	6 minutes ago
03. Data Exploration.html	Add files via upload	6 minutes ago
04. Clustering.html	Add files via upload	6 minutes ago
05. Regression.html	Add files via upload	6 minutes ago
06. Classification.html	Add files via upload	6 minutes ago
07. Classification Evaluation.html	Add files via upload	6 minutes ago
08. Cross Validation.html	Add files via upload	6 minutes ago
09. Parameter Tuning.html	Add files via upload	6 minutes ago
10. AutoML avec Azure ML service.html	Add files via upload	6 minutes ago
11. Déploiement avec AKS.html	Add files via upload	6 minutes ago
12. Azure Databricks et MLFlow - Introduction.html	Add files via upload	6 minutes ago
13. Azure Databricks et MLFlow - Scikit Learn.html	Add files via upload	6 minutes ago
14. Azure Databricks et MLFlow - Batch.html	Add files via upload	6 minutes ago
Cloud Workshop.dbc	Add files via upload	6 minutes ago
README.md	Initial commit	7 minutes ago
index.html	Add files via upload	6 minutes ago

https://github.com/retkowsky/Cloud_Workshop_AzureDatabricks



**Chargement des
données**

Chargement des données

1. Télécharger le fichier Data.zip
2. Dézipper ce fichier sur votre poste de travail pour extraire les différents fichiers .CSV
3. Charger chaque fichier .CSV dans DBFS (Databricks File System) à l'aide de l'assistant suivant et cliquer ensuite sur *Create Table in Notebook* pour visualiser le code Python correspondant.

Microsoft Azure

Create New Table

Data source ?

Upload File DBFS Other Data Sources

Upload to DBFS ?

/FileStore/tables/ (optional) Select

File ?

customers.csv ✓

0.7 MB
Remove file

✓ File uploaded to /FileStore/tables/customers.csv

Create Table with UI

Create Table in Notebook ?

6. Chargement des données

Remplacer les valeurs initiales **false** par **true** du code Python pour inférer automatiquement les propriétés des variables et récupérer le nom de chaque variable en première ligne.

Un *dataframe* est ainsi créé.

Microsoft Azure

2019-03-11 - DBFS Example (5) (Python)

Detached File View: Code Permissions Run All Clear

Cmd 1

Overview

This notebook will show you how to create and query a table or DataFrame that you uploaded to DBFS. DBFS is a Data Lake File System that assumes that you have a file already inside of DBFS that you would like to read from.

This notebook is written in **Python** so the default cell type is Python. However, you can use different languages by using the language selector in the top right corner.

Cmd 2

```
# File location and type
file_location = "/FileStore/tables/flights.csv"
file_type = "csv"

# CSV options
infer_schema = "false"
first_row_is_header = "false"
delimiter = ","

# The applied options are for CSV files. For other file types, these will be ignored.
df = spark.read.format(file_type) \
    .option("inferSchema", infer_schema) \
    .option("header", first_row_is_header) \
    .option("sep", delimiter) \
    .load(file_location)

display(df)
```

