Product: Databricks Lakehouse Platform

Product coverage: Databricks Lakehouse Platform

Technical depth: Concepts + Reference Architecture

Competency: Introductory 200 level session 20 mins slide and 10 mins demo

Talk title: **Destination Lakehouse: All your data, analytics and AI on one platform**

Talk abstract: the data lakehouse is the future for modern data teams seeking to innovate with a data architecture that simplifies data workloads, eases collaboration, and maintains the flexibility and openness to stay agile as a company scales. The Databricks Lakehouse Platform realizes this idea by unifying analytics, data engineering, machine learning, and streaming workloads across clouds on one simple, open data platform. In this session, learn how the Databricks Lakehouse Platform can meet your needs for every data and analytics workload, with examples of real-customer applications, reference architectures, and demos to showcase how you can create modern data solutions of your own.

Proposed speakers: Bharath Gowda (PM), Erika Ehrli (PMM) | Demo: RichardT (OPM)

DATA+AI
SUMMIT 2022

This information is provided to outline Databricks' general product direction and is for informational purposes only. Customers who purchase Databricks services should make their purchase decisions relying solely upon services, features, and functions that are currently available. Unreleased features or functionality described in forward-looking statements are subject to change at Databricks discretion and may not be delivered as planned or at all.

A new generation of data & AI applications

Trillions of rows of data processed from thousands of sources

Financial Services

Credit card **real-time** analysis to detect **fraudulent transactions**

Retail

Real-time customer choices and personalization

Healthcare

COVID-19 data by city to manage **real-time response**

Manufacturing

Predictive maintenance to lower operational and equipment cost

Energy

Monitoring for smart energy pricing models

Gaming

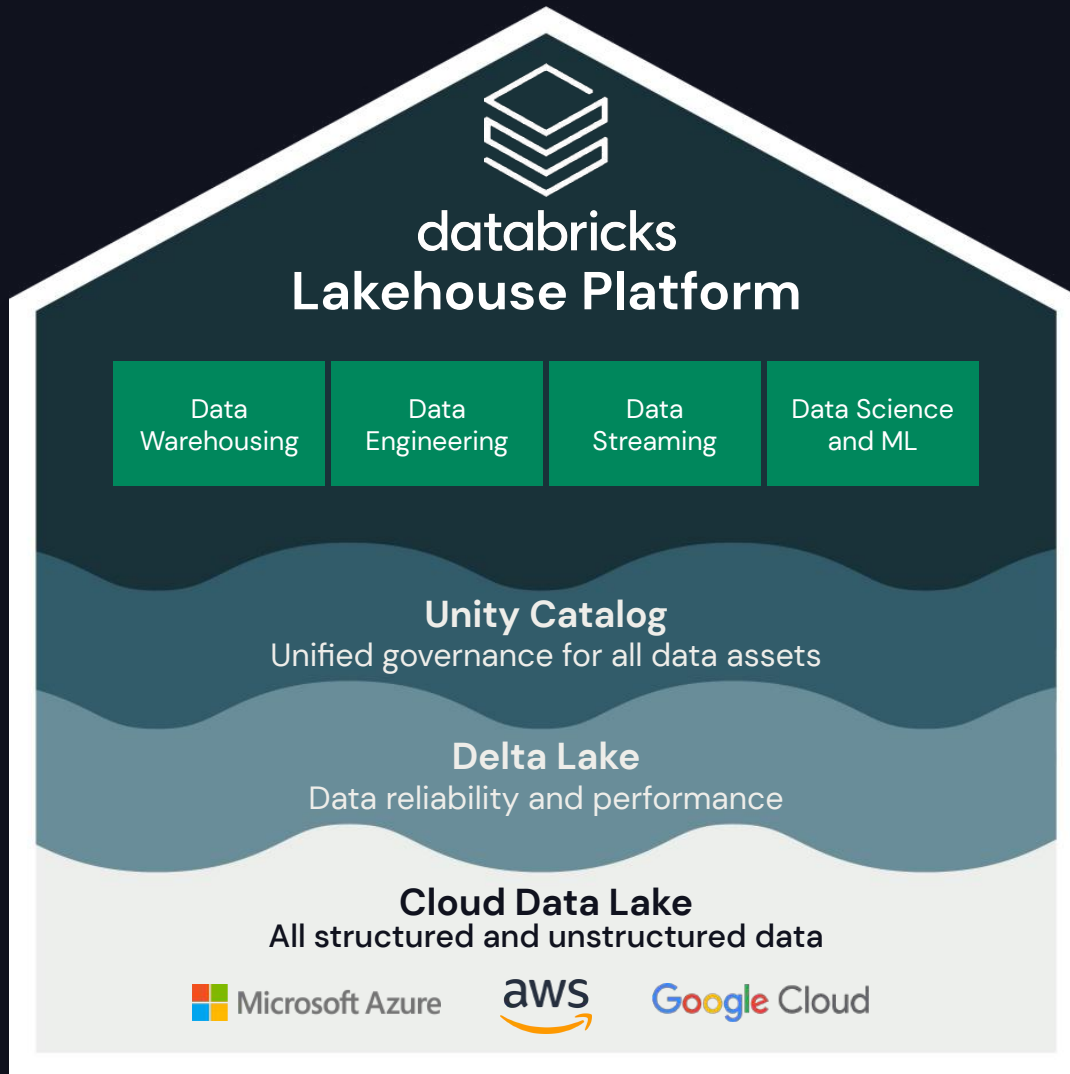**In-game player activity**, interactions feed data into gaming platforms

Technology & Software

Smart cities, smart homes, and connected cars using telemetry on connected IoT devices

Media & Entertainment

Content recommendations based on consumer choices

Customers love and trust Databricks

HSBC

HSBC uses Databricks to quickly deliver new data products for its mobile app, improving customer experience and retention

Use case

- Personalized mobile banking
- Modernize product development for consumer mobile apps by leveraging data and ML

Why Databricks?

- A **unified platform for data science, engineering and business analysts to build new data products** on a shared, single source of data
- Process and de-identify **streaming and batch data from millions of users** in real-time

Impact

- Improved mobile app **engagement by 4.5x**
- Data processing shortened **from 6 hours to 6 sec**
- Replacing 14+ databases
- with 1 data lake to **improve productivity**

Lakehouse: Build a new generation of data and AI applications

Demo: Divvy bike availability
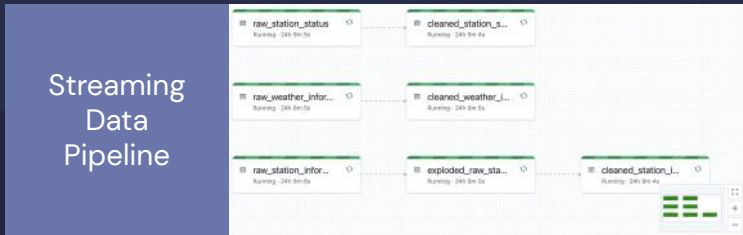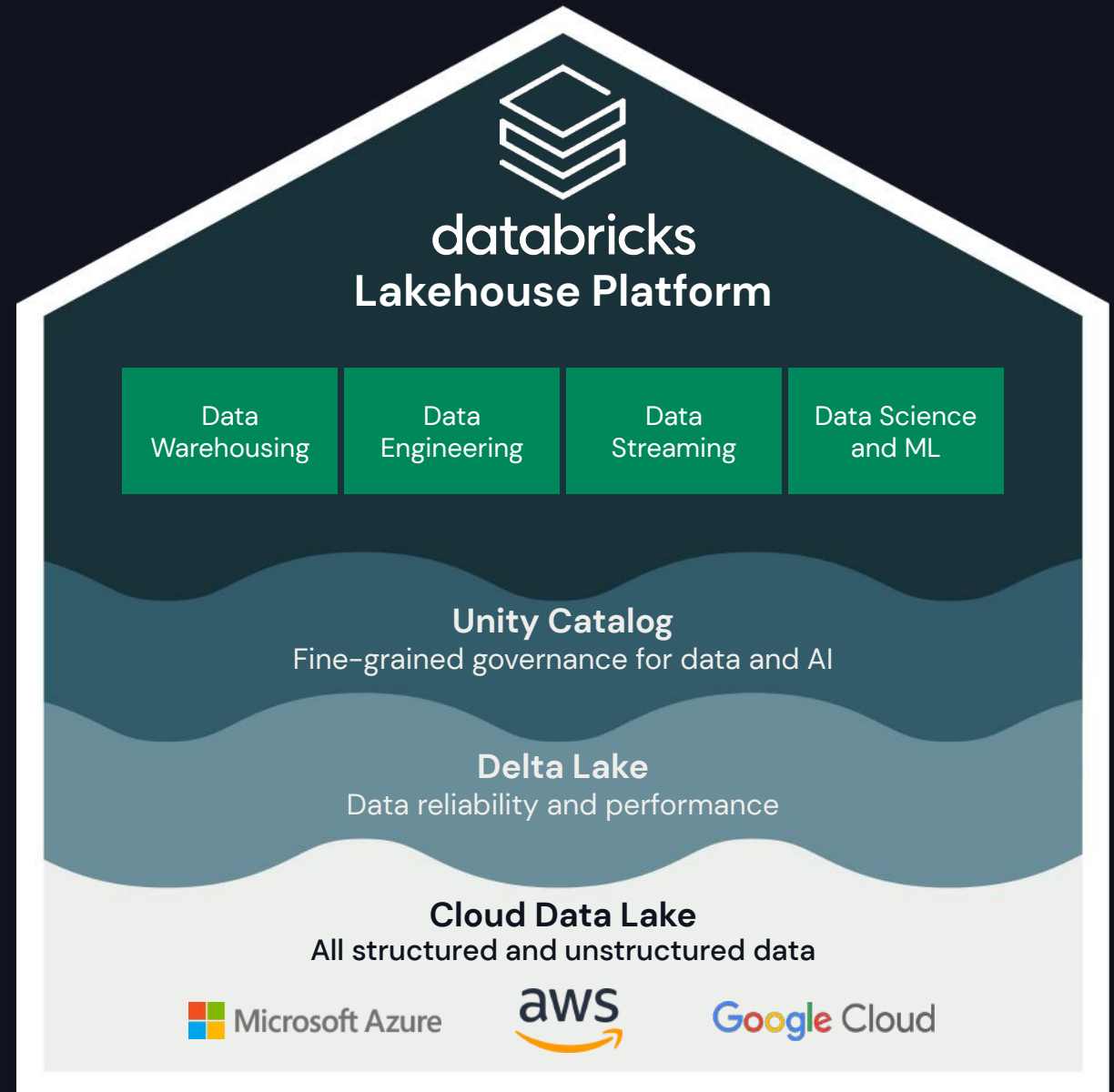
Unified governance for all data & AI assets

Data analyst

Data engineer

Data scientist

Unity Catalog

Single permission model for all assets
ANSI SQL GRANTs
Centralized auditing + lineage

Data Lake

Metadata

Data warehouse

ML models, dashboards
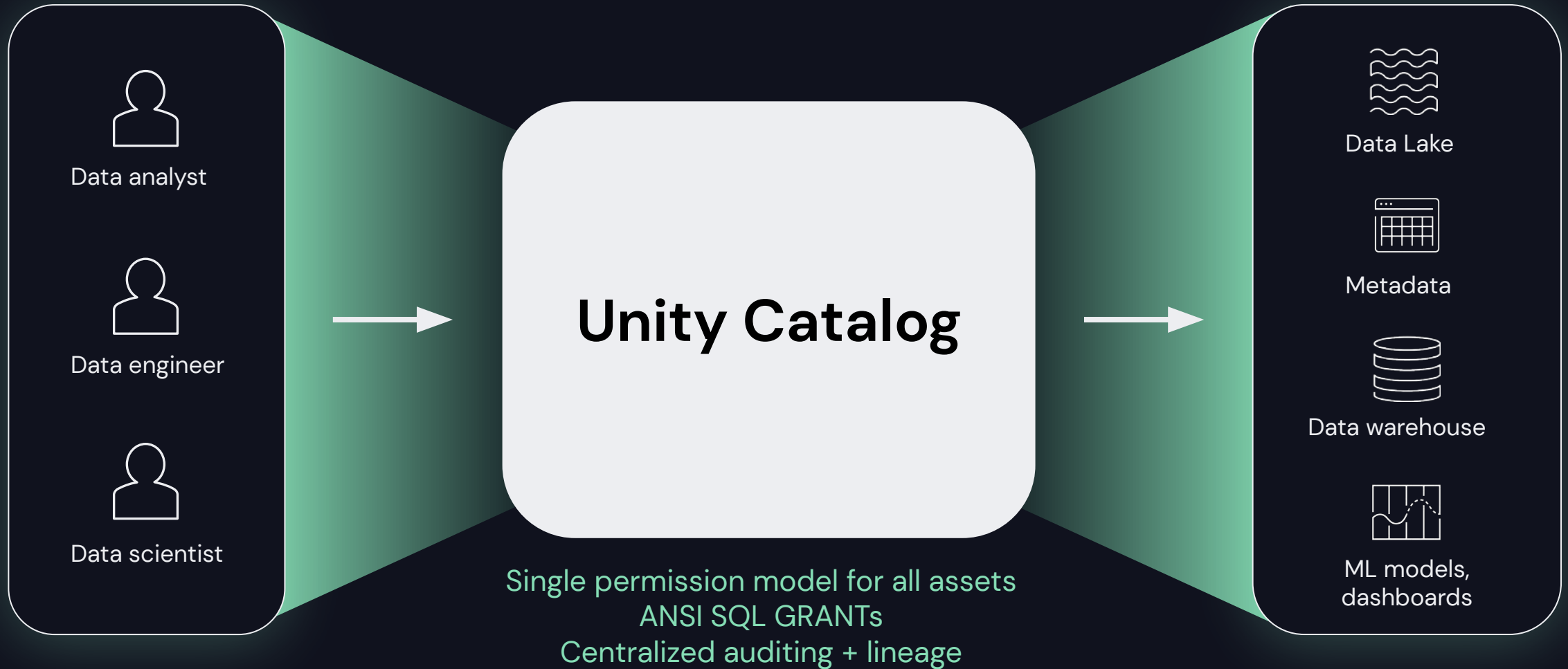
DATA+AI
SUMMIT 2022

Data applications

All your workloads in one platform

Tens of millions of production workloads run daily on Databricks

Data engineering

The best data warehouse is a lakehouse

Data warehousing

Democratize streaming for everyone

Data streaming

Data-native and collaborative solution for the full ML lifecycle

Data Science & ML

Data engineering

Tens of millions of production workloads run daily on Databricks

Simplified data **ingestion** with Auto Loader and COPY INTO

Automated **ETL** with Delta Live Tables for batch and streaming data

Reliable **orchestration** for data, analytics, and AI with Databricks Workflows

End-to-end **observability and monitoring**
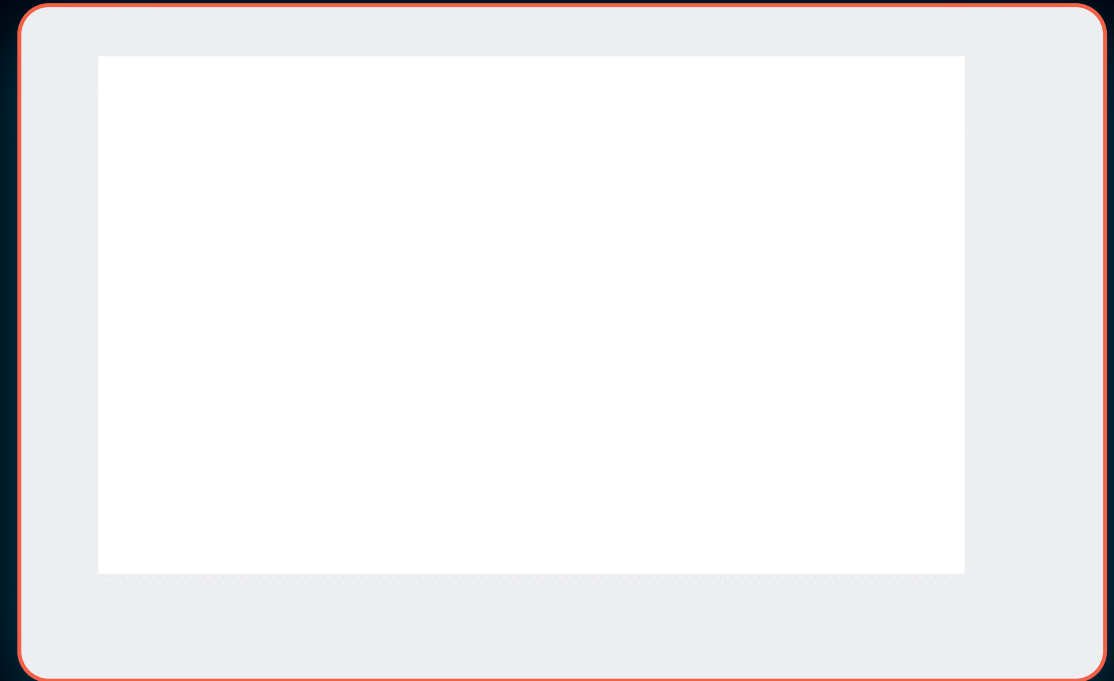
Data warehousing *(powered by Databricks SQL)*

The best data warehouse is a lakehouse

Up to **12x better price/performance**

Lower infrastructure costs and get instant, elastic **serverless** compute

Securely discover, explore, share and govern data assets with **standard SQL**

Connect from any tool (Fivetran, dbt, PowerBI, Tableau, Go, node.js, REST API,...)

Democratize ML with **Python UDFs**
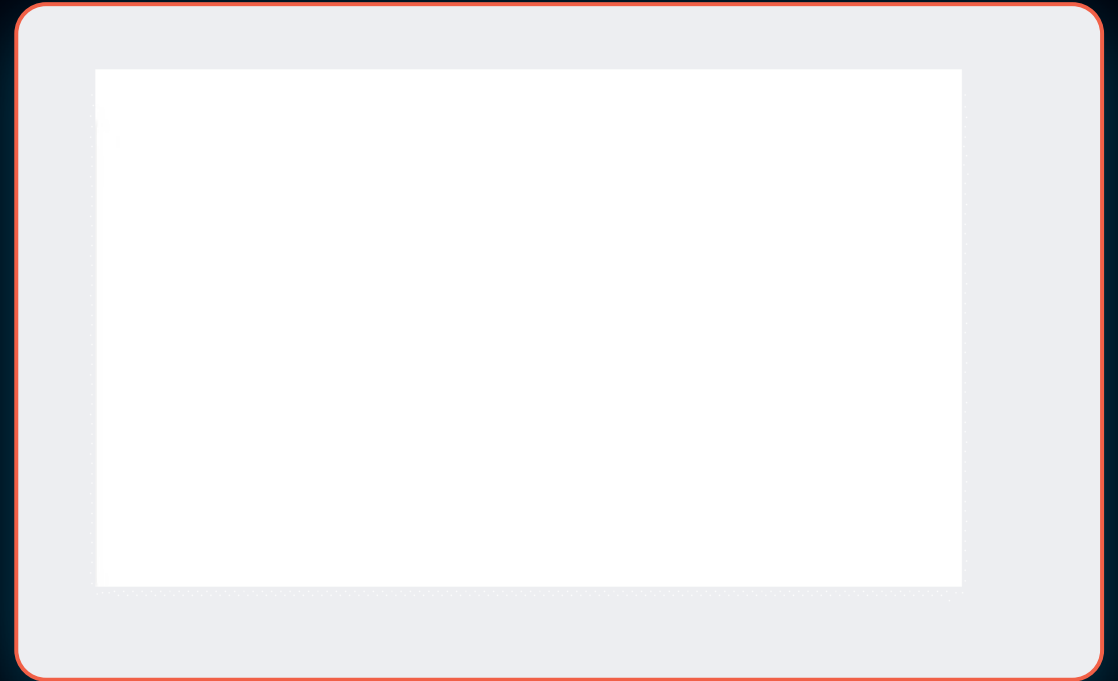
Data streaming

Democratize streaming
for everyone

Unified batch and streaming

Provide diverse users with their **favorite tools** to work with streaming data

Easily configure the **optimal cost structure** for each of your streaming workloads

Out-of-the-box integrations with all popular message buses such as Apache Kafka, Kinesis, EventHub, etc

Create Raw Station Status - Bronze Table - Auto Loader & DLT SQL

```
1   -- Create the bronze station status table containing the raw JSON
2   CREATE STREAMING LIVE TABLE raw_station_status
3   COMMENT "The raw station status data, ingested from /FileStore/DivvyBikes/api_response/stat
4   TBLPROPERTIES ("quality" = "bronze")
5   AS
6   SELECT * FROM cloud_files("/FileStore/DivvyBikes/api_response/station_status", "json", map(
```

Cmd 4

Create Cleaned Station Status - Silver Table - DLT SQL

```
1   -- Create the silver station status table by exploding on station and picking the desired f
2   CREATE STREAMING LIVE TABLE cleaned_station_status (
3     CONSTRAINT valid_station_id EXPECT (station_id IS NOT NULL) ON VIOLATION DROP ROW,
4     CONSTRAINT over_24hr_old_data EXPECT (secs_since_last_reported < 86400)
5     )
6   PARTITIONED BY (last_updated_date)
7   COMMENT "The cleaned station status data with valid station_ids and partitioned by station
8   TBLPROPERTIES ("quality" = "silver")
```

Data science and machine learning

End-to-end machine learning and data science at scale

Built-in **data prep** and **DataOps**

Experimentation and **ModelOps** with MLFlow and Runtime for Machine Learning

DevOps and **end-to-end lineage** with Serverless Model Endpoint, MLFlow, and Model Monitoring
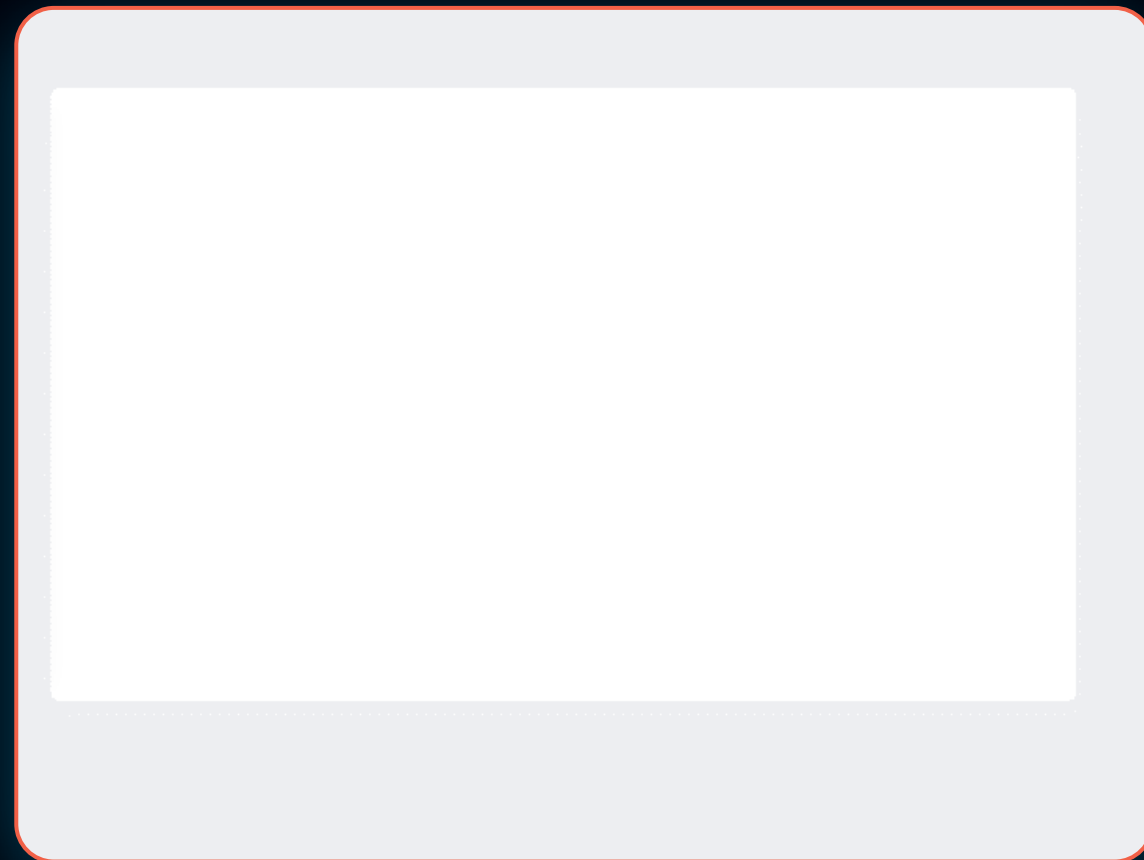
Databricks Lakehouse Platform

Watch these sessions!

Data warehousing on the Lakehouse

Tuesday, June 28 @ 11:30 AM
Jonathan Keller | Haylee Creech |
Franco Patano

Dive deeper into data engineering on Databricks

Wednesday, June 29 @ 10:45 AM
Paul Lappas | Frank Munz

Realizing the promise of streaming with the Databricks Lakehouse platform

Wednesday, June 29 @ 2:50 PM
Ray Zhu | Erika Lee | Steven Yu

ML on the Lakehouse: bringing data and ML together to accelerate AI use cases

Wednesday, June 29 @ 10:45 AM
Kasey Uhlenhuth | Prem Prakash |
Sanjay Bhagat

DATA+AI
SUMMIT 2022

The Data Team's Guide to the Databricks Lakehouse Platform

Download for a deeper dive to the Databricks Lakehouse Platform

Discover how the Lakehouse simplifies complex data engineering, data warehousing, data streaming, data science and machine learning.

https://dbricks.co/3HPWfrG

Demo
Screenshots

SQL

SQL query editor

Create a new query and explore your data in a SQL editor.

Create a query

Sample dashboards

Explore sample dashboards containing rich visualizations and queries.

Visit gallery

Sample data

Analyze a collection of pre-loaded data samples.

Open the Data Explorer

Partner Connect

Fivetran, dbt

Tableau, Power BI

View all partners

Recents

Name	Last viewed
Divvy Bike Real-Time Monitoring	a minute ago
Divvy Bike Hourly Forecast	a minute ago
UC agg query 1	2 hours ago
New query	2 hours ago
Divvy Bike Finance Dashboard	2 days ago
Divvy Bike Marketing Dashboard	2 days ago
Divvy Bike Station Trends	2 days ago
Station & Weather Headlines for Now from Whenever	2 days ago
Station Map Basic from Whenever	2 days ago
Prediction DOW	6 days ago

Favorites

Name	Created at
Divvy Bike Real-Time Monitoring	2022-01-08 10:50
Divvy Bike Station Trends	2022-02-16 16:23
Divvy Bike DLT Data Quality Monitoring	2022-03-23 11:03
Divvy Bike Hourly Forecast	2022-06-17 12:17

Documentation

Databricks SQL user guide

Learn how to generate, visualize and share insights using the built-in SQL editor, dashboards and alerts

Databricks SQL adminstration guide

Learn how to administer Databricks SQL and connect your data to popular third-party BI Tools

SQL reference for Databricks SQL

Release notes

Databricks SQL release notes

Platform release notes

Blog posts

Introducing Databricks SQL on Google Cloud – Now in Public Preview

May 10, 2022

Get to Know Your Queries With the New Databricks SQL Query Profile!

February 23, 2022

Building Data Applications on the Lakehouse With the

⭐ **Divvy Bike Real-Time Monitoring** `real-time` `dlt` `auto-loader` ✏️

Share   Scheduled ⌄   🔄 1 minute ⌄   ⋮

Station List
-- ALL STA... ✕ ⌄

Date and Time
Today/Now ⚡

Station Statistics

78.44
Temp F
🔄 just now

3.43
Wind MPH
🔄 just now

View **real-time and historical stats** for now or any date and time in the past.

Visit the official Chicago Divvy Bikes website.

3,884
Total Bikes Available
🔄 just now

12,404
Total Docks Available
🔄 just now

Weather Description

scattered clouds

clear sky
broken clouds
overcast clouds

🔄 just now

Zero Availability Map - Bikes & Docks

Wheeling
Northbrook

Bikes & Docks Availability - Last 24hrs

Total Bikes Available: 4200, 4000, 3800, 3600, 3400

12:00 Jun 22, 2022 — 15:00 — 18:00 — 21:00 — 00:00 Jun 23, 2022 — 03:00 — 06:00 — 09:00 — 12:00

Date & Time

Total Docks Available: 12.8k, 12.6k, 12.4k, 12.2k, 12k

— Bikes Available   — Docks Available

🔄 just now

Weather Statistics by Hour - Last 24hrs

Avg Temp F: 90, 85, 80, 75, 70

Hour: 12 13 14 15 16 17 18 19 20 21 22 23 24 1 2 3 4 5 6 7 8 9 10 11

Avg Wind MPH: 6, 4, 2, 0

— Temp F   — Wind MPH   ■ Snow IN   ■ Rain IN

🔄 just now

Real-Time Station & Weather Status

Search station_name...

station_name	station_type	weather_desc	temp_F	rain_1h	snow_1h	capacity	bikes_available	docks_available	ebike

⭐ **Divvy Bike Real-Time Monitoring** `real-time` `dlt` `auto-loader` ✏️

Share | Scheduled ⌄ | 🔄 1 minute ⌄ | ⋮

scattered clouds

clear sky

broken clouds
overcast clouds

Hour

▬ Temp F ▬ Wind MPH ■ Snow IN ■ Rain IN

🔄 just now

🔄 just now

Zero Availability Map - Bikes & Docks

🔄

19
2
35
176
17
50   39
223

Leaflet | © OpenStreetMap contributors
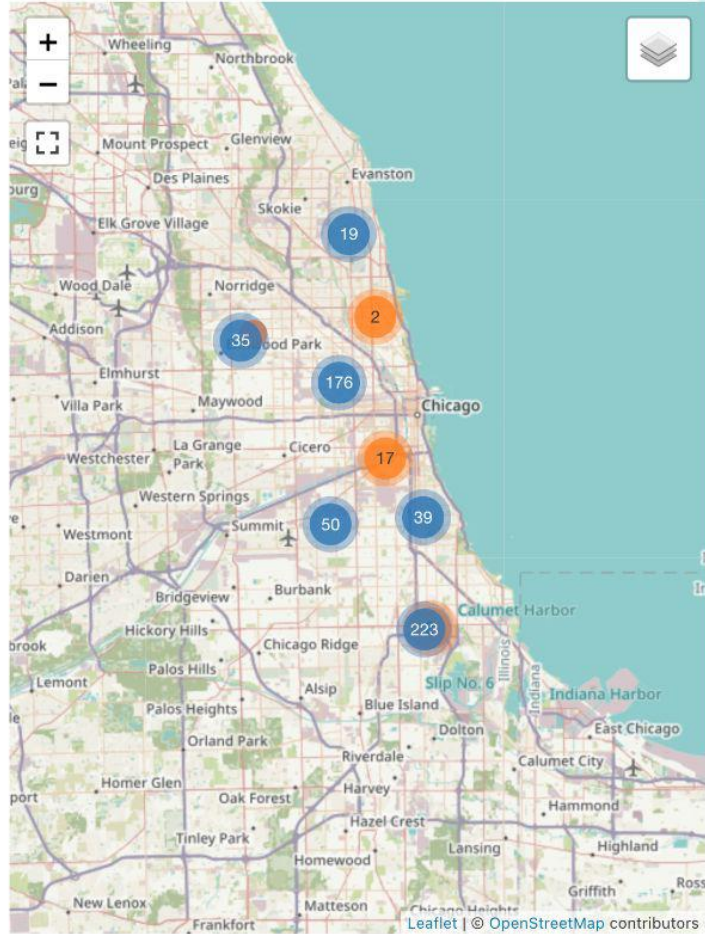
🔄 a minute ago

Real-Time Station & Weather Status

🔄

Search station_name...

station_name	station_type	weather_desc	temp_F	rain_1h	snow_1h	capacity	bikes_available	docks_available	ebike
Canal St & Jackson Blvd	classic	*NULL*	*NULL*	*NULL*	*NULL*	43	36	5	
Clark St & Randolph St	classic	*NULL*	*NULL*	*NULL*	*NULL*	43	33	4	
Ravenswood Ave & Lawrence Ave	classic	scattered clouds	77.43	0.00	0.00	47	32	11	
Paulina St & Flournoy St	classic	scattered clouds	77.95	0.00	0.00	31	29	1	
Larrabee St & Kingsbury St	classic	scattered clouds	77.79	0.00	0.00	43	29	7	
Franklin St & Lake St	classic	*NULL*	*NULL*	*NULL*	*NULL*	27	24	0	
LaSalle St & Jackson Blvd	classic	scattered clouds	77.79	0.00	0.00	35	24	2	
Dearborn St & Monroe St	classic	broken clouds	79.39	0.00	0.00	39	24	11	
Dearborn St & Monroe St	classic	scattered clouds	77.86	0.00	0.00	39	24	11	
University Ave & 57th St	classic	scattered clouds	78.33	0.00	0.00	31	21	2	
Wolcott Ave &									

1 2 3 4 5 ··· 40 >

🔄 a minute ago

⭐ **Divvy Bike Real-Time Monitoring** `real-time` `dlt` `auto-loader` ✏️   Share   Scheduled ⌄   🔄 1 minute ⌄ ⋮

Station List
-- ALL STA... ✕ ⌄

Date and Time
2022-02-02 09:00 ⚡

| ≪ < | Feb 2022 | > ≫ | 2022-02-02 09:00 |

Su	Mo	Tu	We	Th	Fr	Sa
30	31	1	**2**	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	1	2	3	4	5
6	7	8	9	10	11	12

09	00
10	01
11	02
12	03
13	04
14	05
15	06
16	07

Now   Ok

Station Statis...

View **real-time and histor...**
for now or any date and tir...
past.

Visit the official Chicago D...
website.

kes & Docks Availability - Last 24hrs

4200
4000
3800
3600
3400

12.8k
12.6k
12.4k
12.2k
12k

Total Docks Available

12:00
Jun 22, 2022
15:00   18:00   21:00   00:00
Jun 23, 2022
03:00   06:00   09:00   12:00

Date & Time

— Bikes Available — Docks Available
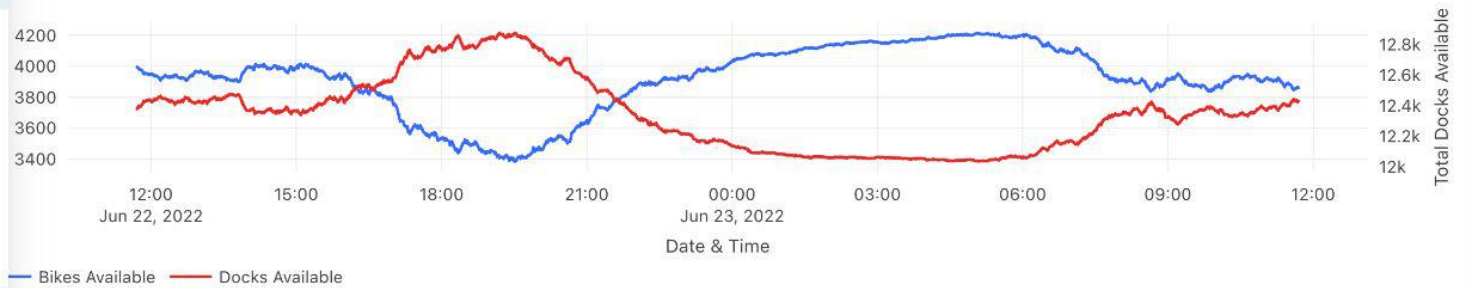
Just now

3,860
Total Bikes Available
🔄 just now

12,430
Total Docks Available
🔄 just now

Weather Statistics by Hour - Last 24hrs

90
85
80
75
70

Avg Temp F

6
4
2
0

Avg Wind MPH

12 13 14 15 16 17 18 19 20 21 22 23 24 1 2 3 4 5 6 7 8 9 10 11
Hour

— Temp F — Wind MPH ■ Snow IN ■ Rain IN

🔄 just now

Weather Description

scattered clouds
clear sky
broken clouds
overcast clouds

🔄 just now

Zero Availability Map - Bikes & Docks
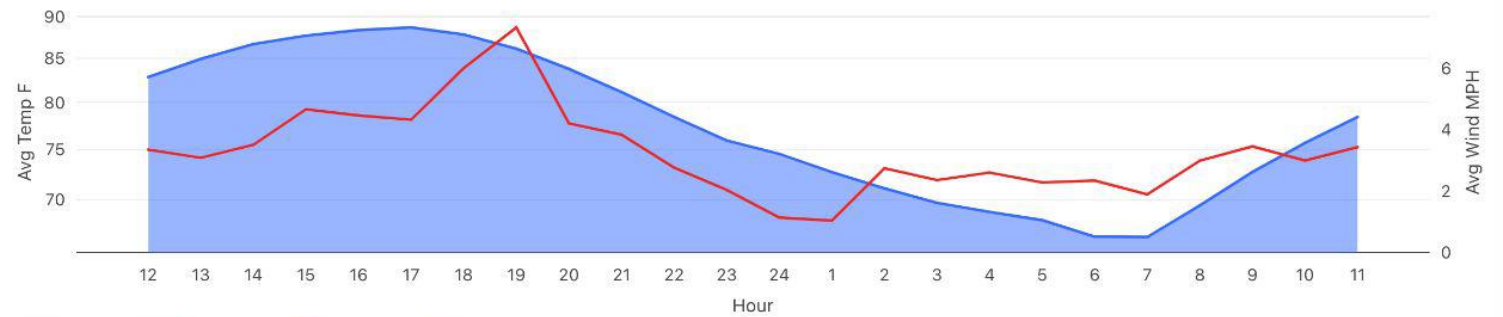
+ Wheeling
Northbrook

Real-Time Station & Weather Status

Search station_name...

| station_name | station_type | weather_desc | temp_F | rain_1h | snow_1h | capacity | bikes_available | docks_available | ebik |

⭐ **Divvy Bike Real-Time Monitoring** `real-time` `dlt` `auto-loader` ✏️   Share | Scheduled ⌄ | ↻ 1 minute ⌄ | ⋮

Station List
-- ALL STA... ✕ ⌄
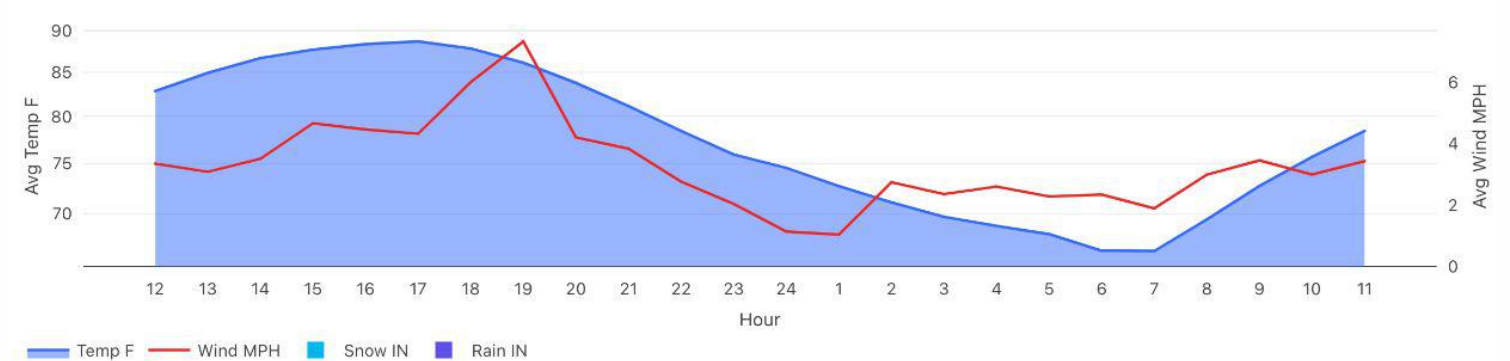
Date and Time
2022-02-02 09:00 ⚡

Station Statistics

Station Statistics

View **real-time and historical stats** for now or any date and time in the past.

Visit the official Chicago Divvy Bikes website.

↻ just now

23.58
Temp F
↻ just now

3.25
Wind MPH
↻ just now

2,196
Total Bikes Available
↻ just now

3,709
Total Docks Available
↻ just now

Weather Description

snow
light snow
mist

↻ just now

Zero Availability Map - Bikes & Docks

Bikes & Docks Availability - Last 24hrs

— Bikes Available   — Docks Available

↻ just now

Weather Statistics by Hour - Last 24hrs

— Temp F   — Wind MPH   ■ Snow IN   ■ Rain IN

↻ just now

Real-Time Station & Weather Status

Search station_name...

station_name	station_type	weather_desc	temp_F	rain_1h	snow_1h	capacity	bikes_available	docks_available	ebike

databricks

e Real-Time Monitoring   real-time   dlt   auto-loader   ✏️

Share   Scheduled ⌄   ⟳ Refresh ⌄   ⋮

| S | SQL | ▲ |

D	Data Science & Engineering	📌
M	Machine Learning	📌
S	SQL	📌
▦	Dashboards	
🔔	Alerts	
△	Data	
⧉	SQL Warehouses	
🕐	Query History	

⬡	Partner Connect
?	Help
⚙	Settings
👤	dbdemo richard.tomlinson@d...
▢	Menu options

Date and Time

ay/Now   ⚡

82.64
Temp F
⟳ just now

2.73
Wind MPH
⟳ just now

tatistics

d historical stats
e and time in the

hicago Divvy Bikes

341
s Available

12,487
Total Docks Available
⟳ just now

iption

w clouds

clear sky
overcast clouds
broken clouds

Bikes & Docks Availability - Last 24hrs
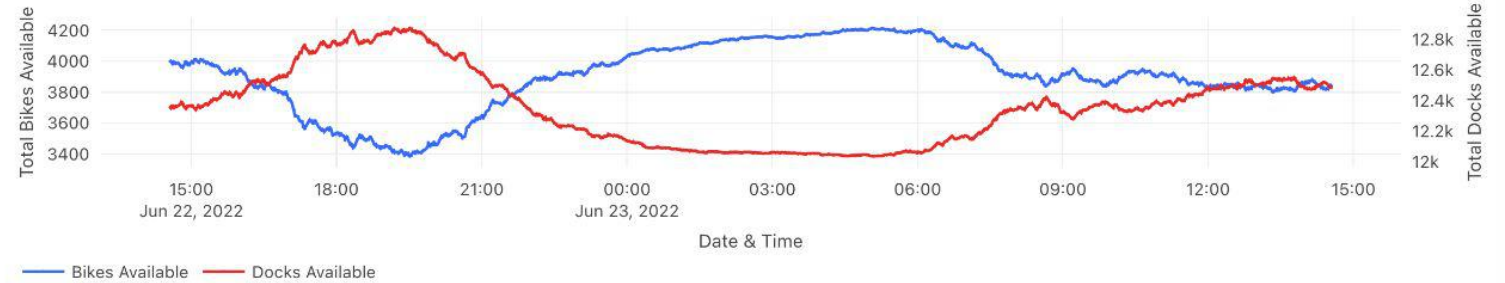
⟳ just now

Weather Statistics by Hour - Last 24hrs
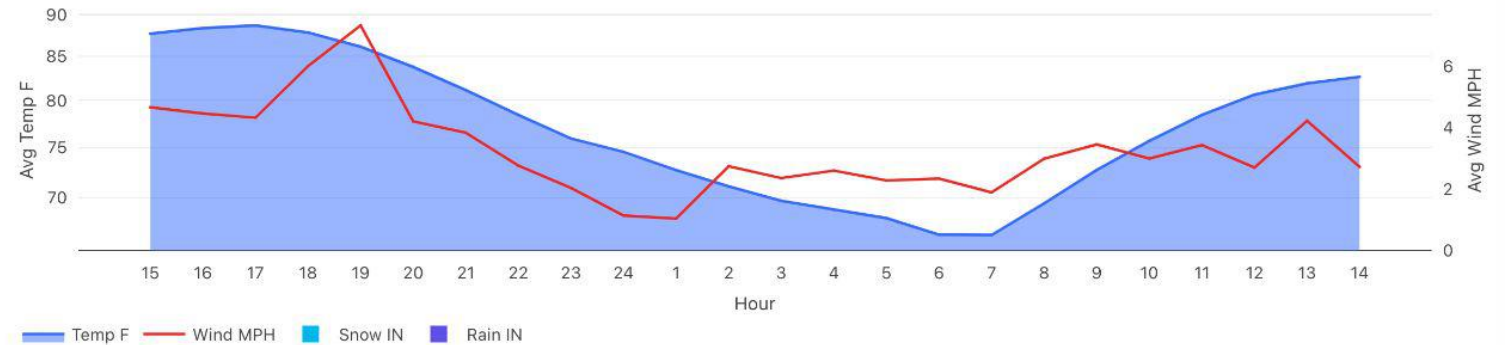
⟳ just now

Real-Time Station & Weather Status

Search station_name...

| station_name | station_type | weather_desc | temp_F | rain_1h | snow_1h | capacity | bikes_available | docks_available | ebike |

y Map - Bikes & Docks
Highland Park

https://adb-8204380433552505.5.azuredatabricks.net/?o=8204380433552505#

Data Science & Engineering

Notebook
Create a new notebook for querying, data processing, and machine learning.

Create a notebook

Data import
Quickly import data, preview its schema, create a table, and query it in a notebook.

Browse files

Partner Connect
Fivetran, dbt

Tableau, Power BI

View all partners

Guide: Quickstart tutorial
Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

Start tutorial

Recents

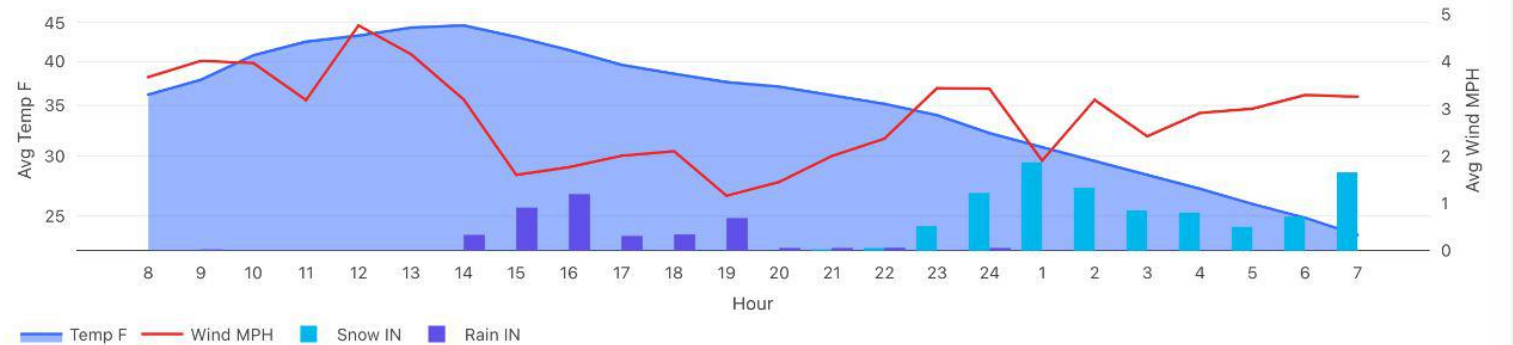| Name	Last viewed
Create Divvy Bike Tables for UC	2 days ago
Create Divvy Bike Aggs for UC	2 days ago
sql-aggs-divvybikes1	2 days ago
prepare_forecast_dataset_for_training	2 days ago
lineage	2 days ago
Titanic FS Demo	5 days ago
Final-Prophet-Cloned	6 days ago
22-06-16-21:46-Prophet-f23578de88f5cad2903d637e69f22f58	7 days ago

Documentation

Get started guide
This tutorial gets you going with Azure Databricks Data Science & Engineering

Best practices
Get the best performance when using Azure Databricks

Data guide
How to work with data in Azure Databricks

More documentation

Release notes

Runtime release notes

Azure Databricks preview releases

Platform release notes

More release notes

Blog posts

Building ETL pipelines for the cybersecurity lakehouse with Delta Live Tables
June 3, 2022

Streaming Windows Event Logs into the Cybersecurity Lakehouse
May 5, 2022

Speed Up Streaming Queries With Asynchronous State Checkpointing
May 2, 2022

More blog posts

databricks

D Data Science & En... ▾

⊕ Create

▦ Workspace

Repos

Recents

Search

Data

Compute

Workflows

Partner Connect

? Help

⚙ Settings

dbdemo
richard.tomlinson@d...

Menu options

Workspace

🏠 Home

Users	▾
🏠 richard.tomlinson@databricks.com	
■ 6fa1ca0d-aee5-4410-ad1d-0efb...	
■ abraham.pabbathi@databricks.c...	
■ amit.kara@databricks.com	▾
■ bharath.gowda@databricks.com	▾
■ chris.hoshinofish@databricks.com	
■ franco.patano@databricks.com	▾
■ frank.munz@databricks.com	▾
■ fred.abood@databricks.com	▾
■ harsh.panchal@databricks.com	▾
■ jeffery.annor@databricks.com	▾
■ jimmy.xu@databricks.com	▾
■ john.wiederhold@databricks.com	▾
■ justin.tang@databricks.com	▾
◀ kyle.hale@databricks.com	▾
■ leo.furlong@databricks.com	▾
■ liwen.sun@databricks.com	▾
■ max.nienu@databricks.com	▾
■ paul.roome@databricks.com	▾
■ quentin.ambard@databricks.com	▾
■ samantha.steiny@databricks.com	▾
■ sangbae.lim@databricks.com	▾
■ sean.owen@databricks.com	▾
■ shant.hovsepian@databricks.com	▾
🗀 chris.hoshinofish@databricks.co...	

richard.tomlinson@databricks.com	▾
🗑 Trash	
🗀 Batch-Inference	
🗀 databricks_automl	
🗀 Divvy Bikes	
📄 Querying the Delta Live Tables	
📄 SQL Scratchpad 1	
🗀 Titanic	▾
🗀 UC Demo	▾

Create	▶
Clone	
Import	
Export	▶
Permissions	
Copy Link Address	

Notebook
Library
Folder
MLflow Experiment

Partner Connect

Fivetran, dbt

Tableau, Power BI

View all partners

Guide: Quickstart tutorial

Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

Start tutorial

	Last viewed
	2 days ago
	2 days ago
	2 days ago
	2 days ago
	2 days ago
	5 days ago
	6 days ago
	7 days ago

se notes

me release notes

Databricks preview releases

rm release notes

release notes

Blog posts

Building ETL pipelines for the cybersecurity lakehouse with Delta Live Tables
June 3, 2022

Streaming Windows Event Logs into the Cybersecurity Lakehouse
May 5, 2022

Speed Up Streaming Queries With Asynchronous State Checkpointing
May 2, 2022

More blog posts

databricks

D Data Science & En... ▾

⊕ Create

▤ Workspace

⊡ Repos

◷ Recents

Q Search

△ Data

⧉ Compute

☱ Workflows

Repos

Add Repo

Repos ⌄

richard.tomlinson@databricks.com ⌄

▪ richard.tomlinson@databricks.com

⬚ divvybikes   ⑂ main   ▾

✛ Add folder

Git...

Create   ▸

Rename

Delete

Import

Export   ▸

Permissions

Copy Link Address

preview its schema.

Partner Connect

Fivetran, dbt

Tableau, Power BI

View all partners

Guide: Quickstart tutorial

Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

Start tutorial

Last viewed

2 days ago

2 days ago

2 days ago

2 days ago

2 days ago

5 days ago

6 days ago

7 days ago

se notes

me release notes

Databricks preview releases

rm release notes

release notes

Blog posts

Building ETL pipelines for the cybersecurity lakehouse with Delta Live Tables
June 3, 2022

Streaming Windows Event Logs into the Cybersecurity Lakehouse
May 5, 2022

Speed Up Streaming Queries With Asynchronous State Checkpointing
May 2, 2022

More blog posts

⬚ Partner Connect

⑦ Help

⚙ Settings

◉ dbdemo
   richard.tomlinson@d...

▭ Menu options

Compute   Preview   **Provide feedback**

All-purpose clusters   Job clusters   Pools   Cluster policies ⊘

Create Cluster

Created by me | **Accessible by me** | 🔍 Search within clusters you have access to

		Name		Policy		Runtime	Active memory	Active cores	Active DBU / h	Source	Creator				
📌	⊘	MLCluster1		-		10.5	42 GB	12 cores	2.25	UI	richard.tomlinson@databricks.com	-	⋮	■
📌	⊘	Photon Cluster Standard 2		-		10.4	384 GB	48 cores	13.8	UI	richard.tomlinson@databricks.com	-	⋮	■
📌	⊘	Single Node Jobs 1		-		9.1	14 GB	4 cores	0.75	UI	richard.tomlinson@databricks.com	-	⋮	■
📌	●	AutoML		-		10.5	-	-	-	UI	franco.patano@databricks.com	-	⋮	▶

1

Workflows

Jobs   Job runs   Delta Live Tables

Create Job   🔍 Filter

Owned by me   Accessible by me   🔍 Filter

	Name ⇅	Job ID ⇅	Created by ⇅	Task ⇅	Cluster ⇅	Schedule ⇅	Last run ⇅	Actions
⚪	INGEST: Get Station Availability	97	richard.tomlinson@databr...	realtime-divvybike-api-ingest-stationstatus	⊘ Single Node Jobs 1	Every minute (UTC)	Succeeded	▶ 🗑
⚪	INGEST: Get Station Weather Info	487	richard.tomlinson@databr...			At 1 minutes past th...	Succeeded	▶ 🗑
🟢	AGGS: Create UC Aggregate Tables	435706635...	richard.tomlinson@databr...	Create Divvy Bike Tables for UC	⊘ Photon Cluster Standard 2	At 45 minutes past t...	Succeeded	■ 🗑
🟢	ML: Build Training Dataset and Retrain Model	106841396...	richard.tomlinson@databr...			At 45 minutes past t...	Running	■ 🗑

1-4 of 4 items   1

Workflows > Jobs > INGEST: Get Station Weather Info

INGEST: Get Station Weather Info

Run now

Runs   Tasks

Runs

Matrix

Jun 22, 11:01

Jun 23, 11:01

32m 7s

16m 3s

Job Runs

Tasks

Get_Station_Info
Get_Weather_Info
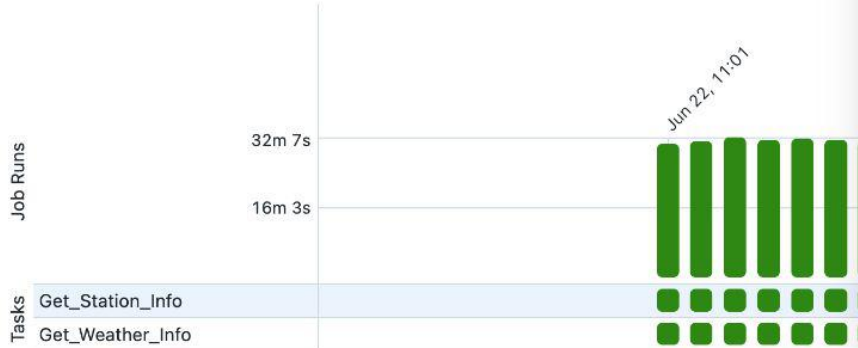
Get_Station_Info

Job ID	487
Job run ID	27250140
Task run ID	27250777
Status	⊘ Succeeded
Started	2022-06-23 11:01:26 CDT
Duration	8s
Notebook	/Users/richard.tomlinson@databricks.com/Divvy Bikes/realtime-divvybike-api-ingest-stationinformation
Cluster	⊘ Single Node Jobs 1

Driver: Standard_DS3_v2, Workers: Standard_DS3_v2, 0 workers, 9.1 LTS (includes Apache Spark 3.1.2, Scala 2.12)

View cluster   Spark UI

Logs   Metrics

ⓘ **Job details**

Job ID	487
Creator	richard.tomlinson@databricks.com
Run as ❓	richard.tomlinson@databricks.com
Tags ❓	+ Tag

Git

Add Git settings

Schedule

At 1 minutes past the hour (UTC+00:00 — UTC)

Edit schedule   Pause

Clusters

⊘ Single Node Jobs 1

Driver: Standard_DS3_v2, Workers: Standard_DS3_v2, 0 workers, 9.1 LTS (includes Apache Spark 3.1.2, Scala 2.12)

View cluster   Swap   Spark UI   Logs   Metrics

Notifications ❓

No notifications

Edit notifications

Permissions ❓

richard.tomlinson@databricks.com

https://adb-8204380433552505.5.azuredatabricks.net/?o=8204380433552505#job/487/run/27250777

Workflows > Jobs > INGEST: Get Station Weather Info

INGEST: Get Station Weather Info

⋮   **Run now** ⌄

Runs   **Tasks**

487 ⎘

Get_Station_Info

🗀 ...api-ingest-stationinformation

⛓ Single Node Jobs 1

Get_Weather_Info

🗀 ...altime-weatherinfo-api-ingest

⛓ Single Node Jobs 1

ⓘ **Job details**

Job ID	487 ⎘
Creator	richard.tomlinson@databricks.com
Run as ❓	richard.tomlinson@databricks.com
Tags ❓	+ Tag

⑂ **Git**

Add Git settings

📅 **Schedule**

At 1 minutes past the hour (UTC+00:00 — UTC)

Edit schedule   Pause

⛓ **Clusters**

⊘ Single Node Jobs 1

Driver: Standard_DS3_v2, Workers: Standard_DS3_v2, 0 workers, 9.1 LTS (includes Apache Spark 3.1.2, Scala 2.12)

View cluster   Swap   Spark UI   Logs   Metrics

🔔 **Notifications** ❓

No notifications

Edit notifications

🔒 **Permissions** ❓

richard.tomlinson@databricks.com

Workflows > Jobs > INGEST: Get Station Weather Info

INGEST: Get Station Weather Info

⋮   **Run now** ▾

Runs   **Tasks**

Job details

Job ID   487 ⎘

Creator   richard.tomlinson@databricks.com

Run as ⓘ   richard.tomlinson@databricks.com

Tags ⓘ   + Tag

Task name * ⓘ

Get_Weather_Info

Type *
Notebook   ▾

Source * ⓘ
Workspace

Path * ⓘ

...tomlinson@databricks.com/Divvy Bikes/realtime-weatherinfo-api-ingest ⧉ ▢   ⎘

Cluster * ⓘ

Single Node Jobs 1 (14.00 GB | 4 Cores | DBR 9.1 LTS | Spark 3.1.2 | Scala 2.... ⧉ ▾

⚠ Jobs running on all-purpose clusters are considered all-purpose compute. Learn more

Parameters ⓘ   UI | JSON
Add

Depends on

Get_Station_Info ✕   ▾

⌄ Advanced options

Cancel   **Save task**

Git

Add Git settings

Schedule

At 1 minutes past the hour (UTC+00:00 — UTC)

Edit schedule   Pause

Clusters

⊘ Single Node Jobs 1

Driver: Standard_DS3_v2, Workers: Standard_DS3_v2, 0 workers, 9.1
LTS (includes Apache Spark 3.1.2, Scala 2.12)

View cluster   Swap   Spark UI   Logs   Metrics

Notifications ⓘ

No notifications

Edit notifications

Permissions ⓘ
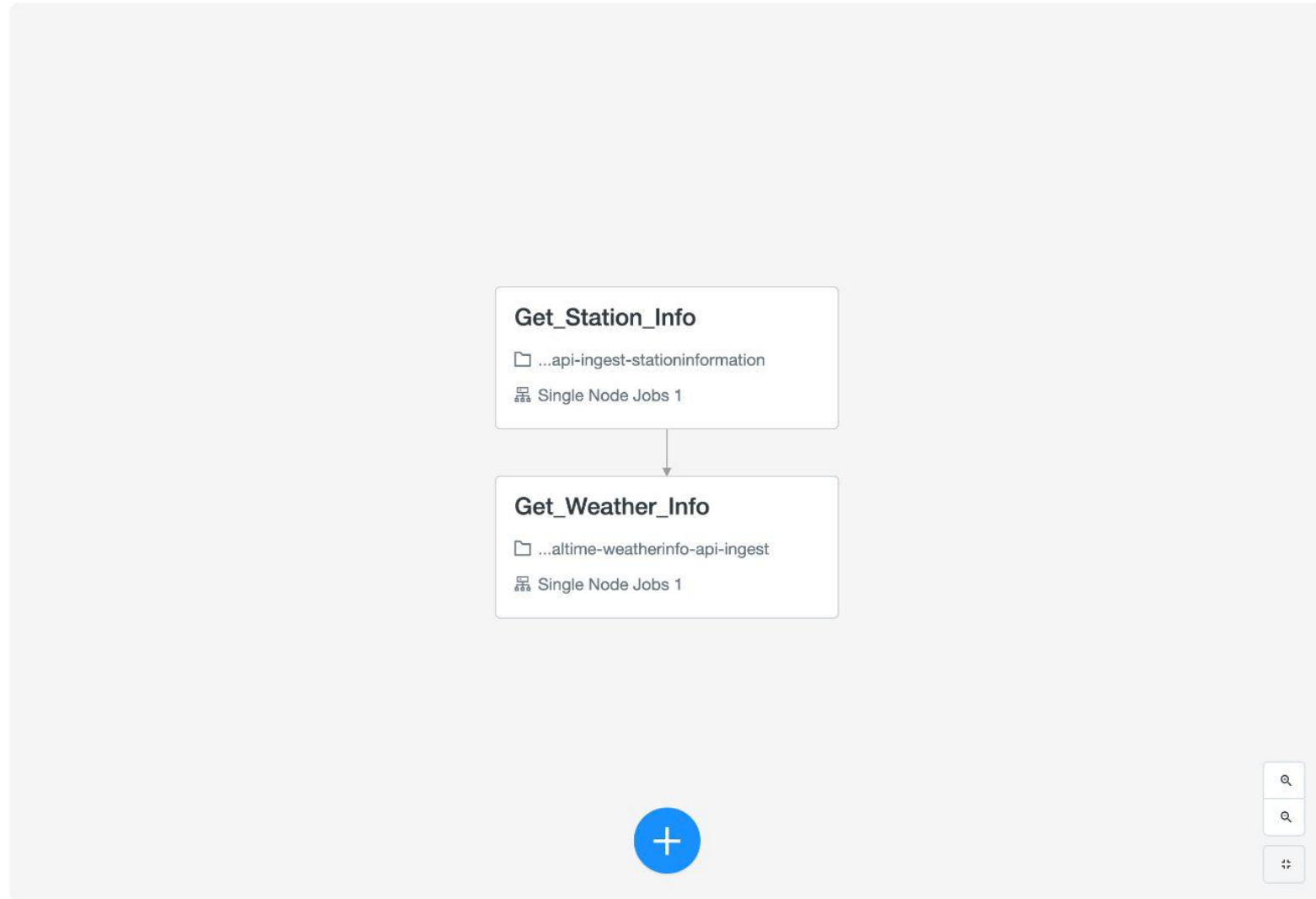
richard.tomlinson@databricks.com

Workflows

Jobs   Job runs   **Delta Live Tables**

Create Pipeline   All   Owned by me

Name	Recent updates	ID	Owner	Actions
TRANSFORM: Live Divvy Bike Station and Weather Pipeline	⊖ ⊖ ⊖ ⊖ ↻	34cbef77-a6d2-4577-97cb-717f5636af3c	richard.tomlinson@databricks.com	■ 🗑

Workflows / Delta Live Tables / Pipeline details

TRANSFORM: Live Divvy Bike Station and Weather Pipeline

Development | Production ❓ Delete   Permissions   Settings   Schedule ⌄   **Stop**

6/21/2022, 2:01:20 PM · 🔄 Running ⌄

| ⊞ raw_station_status 🔄 | ⊞ cleaned_station_s... 🔄 |
| Running · 45h 49m 26s | Running · 45h 49m 26s |

| ⊞ raw_weather_infor... 🔄 | ⊞ cleaned_weather_i... 🔄 |
| Running · 45h 49m 27s | Running · 45h 49m 26s |

| ⊞ raw_station_infor... 🔄 | ⊞ exploded_raw_sta... 🔄 | ⊞ cleaned_s... |
| Running · 45h 49m 27s | Running · 45h 49m 26s | Running · 45h 4 |
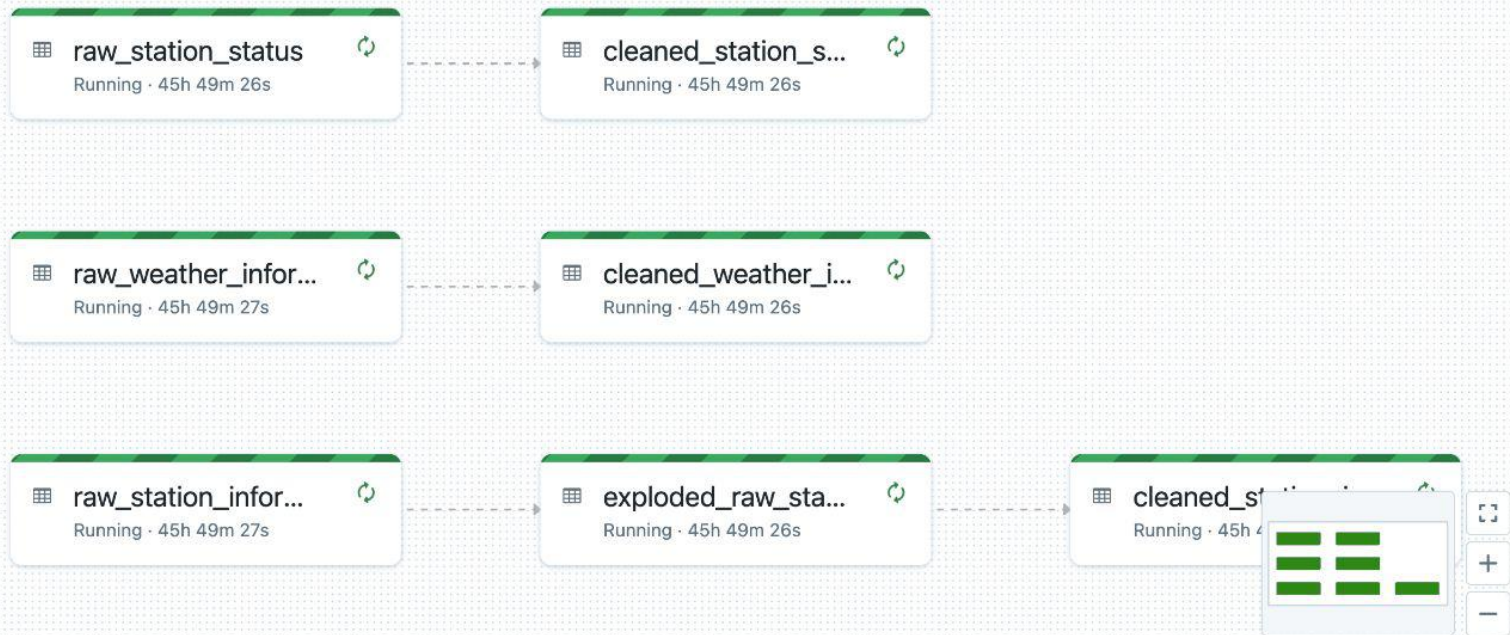
Pipeline details

Name	TRANSFORM: Live Divvy Bike Station and Weather Pipeline
Pipeline ID	34cbef77-a6d2-4577-97cb-717f5636af3c
Paths	/Users/richard.tomlinson@databricks.com/Divvy Bikes/sql-autoloader-divvybikes1 ↗
Run as ❓	richard.tomlinson@databricks.com

Update details

Update ID	99054520-a02b-43af-b1fa-50f165ff3352
Status	🔄 Running
Creation time	6/21/2022, 2:01:20 PM
Duration	45h 49m 41s
Start time	6/21/2022, 2:01:25 PM
Run time	45h 49m 36s

All | ⊘ Info | 🟡 Warning | ⊗ Error   🔍 Filter...   ✕

⊘ 2 days ago	flow_progress	Flow 'exploded_raw_station_information' is RUNNING.
⊘ 2 days ago	flow_progress	Flow 'cleaned_station_status' is STARTING.
⊘ 2 days ago	flow_progress	Flow 'cleaned_station_status' is RUNNING.
⊘ 2 days ago	flow_progress	Flow 'cleaned_station_information' is STARTING.
⊘ 2 days ago	flow_progress	Flow 'cleaned_station_information' is RUNNING.

sql-autoloader-divvybikes1 [SQL]

Schedule ˅   **Share**

Detached | File ▾   Edit ▾   View: Standard ▾   Run All   Clear ▾   Comments   Experiment   Revision history

Cmd 1

Divvy Bikes Data Pipeline (Main)   [Markdown]

SQL-based Delta Live Tables script leveraging Auto Loader to handle streaming (and batch) data.

Cmd 2

Global Settings

```sql
1  -- Increase the default # files used to infer schema as rain & snow do not occur frequently
2  SET spark.databricks.cloudFiles.schemaInference.sampleSize.numFiles = 100000
```

Cmd 3

Create Raw Station Status - Bronze Table - Auto Loader & DLT SQL

```sql
1  -- Create the bronze station status table containing the raw JSON
2  CREATE STREAMING LIVE TABLE raw_station_status
3  COMMENT "The raw station status data, ingested from /FileStore/DivvyBikes/api_response/station_status."
4  TBLPROPERTIES ("quality" = "bronze")
5  AS
6  SELECT * FROM cloud_files("/FileStore/DivvyBikes/api_response/station_status", "json", map("cloudFiles.inferColumnTypes", "true"));
```

Cmd 4

Create Cleaned Station Status - Silver Table - DLT SQL

```sql
1   -- Create the silver station status table by exploding on station and picking the desired fields.
2   CREATE STREAMING LIVE TABLE cleaned_station_status (
3     CONSTRAINT valid_station_id EXPECT (station_id IS NOT NULL) ON VIOLATION DROP ROW,
4     CONSTRAINT over_24hr_old_data EXPECT (secs_since_last_reported < 86400)
5     )
6   PARTITIONED BY (last_updated_date)
7   COMMENT "The cleaned station status data with valid station_ids and partitioned by station id."
8   TBLPROPERTIES ("quality" = "silver")
9   AS
10  SELECT
11    stations.station_id,
12    stations.num_bikes_available,
13    stations.num_bikes_disabled,
14    stations.num_docks_available,
15    stations.num_docks_disabled,
```

sql-autoloader-divvybikes1 SQL   Schedule ⌄   **Share**

Detached   File ▾   Edit ▾   View: Standard ▾   Run All   Clear ▾   Comments   Experiment   Revision history

```sql
24    last_updated,
25    CAST(last_updated AS timestamp) AS last_updated_ts
26  FROM STREAM(LIVE.raw_station_information)
27  );
```

Cmd 9

Create Cleaned Station Information - Silver Table - DLT SQL

```sql
1   -- Create the silver station information table.
2   CREATE STREAMING LIVE TABLE cleaned_station_information
3   COMMENT "The cleaned station information data."
4   TBLPROPERTIES ("quality" = "silver");
```

Cmd 10

Merge (Upsert) Cleaned Station Information - Silver Table - DLT SQL

```sql
1   -- Upsert/merge new stations into silver station information table.
2   APPLY CHANGES INTO LIVE.cleaned_station_information FROM STREAM(LIVE.exploded_raw_station_information)
3     KEYS (station_id)
4     SEQUENCE BY last_updated;
```

Shift+Enter to run

Data Explorer

+ 　🔍 Search tables　　　　　Browse DBFS　　🔷 Primary SQL Endpoint (L)　⊘ ⌄

Data ⌃

> 🗄 hive_metastore
> 🗄 samples
> 🗄 asean_catalog
> 🗄 bmathew_unity
> 🗄 catalogauo
> 🗄 deepaksekardemo
> 🗄 delta_sharing_aa
> 🗄 delta_sharing_david_radford
> 🗄 deltasharing
> 🗄 dsatty_uc
> 🗄 field_demos
> 🗄 jiri
> 🗄 main
> 🗄 migrationtestrs
> 🗄 mla_demos
> 🗄 quickstart_catalog
> 🗄 quickstart_catalog_2
> 🗄 quickstart_catalog_taka
> 🗄 shuwu_demo
> 🗄 sumit_uc_demo
> 🗄 uc_ajay
> 🗄 uc_mina
> 🗄 uc_roy
> 🗄 volker_uc
> 🗄 vuongnguyen

Storage Credentials ⌄

🗄 Catalogs

Create catalog

25 Catalogs　　　　　　　　　　　　　　　　　🔍 Filter catalogs...

Name	Created at	Owner
hive_metastore		
samples		
asean_catalog	2022-06-12 04:52:07	mendelsohn.chan@databricks.com
bmathew_unity	2022-06-09 16:44:56	binu.mathew@databricks.com
catalogauo	2022-06-23 03:00:16	yunna.wei@databricks.com
deepaksekardemo	2022-06-22 23:24:46	deepak.sekar@databricks.com
delta_sharing_aa	2022-05-25 06:10:48	vuong.nguyen@databricks.com
delta_sharing_david_radford	2022-06-22 14:32:15	david.radford@databricks.com
deltasharing	2022-04-28 10:47:21	mattia.zeni@databricks.com
dsatty_uc	2022-05-05 16:36:11	derek.satterley@databricks.com
field_demos	2022-04-29 07:45:11	quentin.ambard@databricks.com
jiri	2022-06-02 02:53:53	jiri.harazim@databricks.com
main	2022-04-11 15:55:13	paul.roome@databricks.com
migrationtestrs	2022-06-09 16:19:41	roberto.salcido@databricks.com
mla_demos	2022-05-31 12:10:07	matthieu.lamairesse@databricks.com
quickstart_catalog	2022-04-11 16:27:09	paul.roome@databricks.com
quickstart_catalog_2	2022-06-20 03:56:28	vuong.nguyen@databricks.com
quickstart_catalog_taka	2022-06-01 02:16:27	takaaki.yayoi@databricks.com

1　2　>　　20 / page ⌄

Data Explorer

Browse DBFS   Primary SQL Endpoint (L)

Data

- hive_metastore
- samples
- asean_catalog
- bmathew_unity
- catalogauo
- deepaksekardemo
- delta_sharing_aa
- delta_sharing_david_radford
- deltasharing
- dsatty_uc
- field_demos
- jiri
- main
  - default
  - divvy_bikes
    - agg_top_stations_and_weather_last_30_days
    - cleaned_station_information
    - cleaned_station_status
    - cleaned_weather_information
    - exploded_raw_station_information
    - raw_station_information
    - raw_station_status
    - raw_weather_information
  - tpch_lineage_demo
- migrationtestrs

Storage Credentials

Catalogs > main >

main.divvy_bikes

Comment:

Owner: richard.tomlinson@databricks.com

Tables   Details   Permissions

8 Tables

Filter tables...

Name	Created at	Owner
agg_top_stations_and_weather_last_30_days	2022-06-14 10:43:49	richard.tomlinson@databricks.com
cleaned_station_information	2022-06-14 09:48:25	richard.tomlinson@databricks.com
cleaned_station_status	2022-06-13 21:45:36	richard.tomlinson@databricks.com
cleaned_weather_information	2022-06-14 09:28:48	richard.tomlinson@databricks.com
exploded_raw_station_information	2022-06-23 11:48:16	richard.tomlinson@databricks.com
raw_station_information	2022-06-14 09:30:16	richard.tomlinson@databricks.com
raw_station_status	2022-06-13 21:39:34	richard.tomlinson@databricks.com
raw_weather_information	2022-06-14 09:26:59	richard.tomlinson@databricks.com

Data Explorer

Data   ⌃

> 🗄 hive_metastore
> 🗄 samples
> 🗄 asean_catalog
> 🗄 bmathew_unity
> 🗄 catalogauo
> 🗄 deepaksekardemo
> 🗄 delta_sharing_aa
> 🗄 delta_sharing_david_radford
> 🗄 deltasharing
> 🗄 dsatty_uc
> 🗄 field_demos
> 🗄 jiri
> 🗄 main
  > 🗄 default
  > 🗄 divvy_bikes
      ⊞ agg_top_stations_and_weather_last_30_days
      ⊞ cleaned_station_information
      ⊞ cleaned_station_status
      ⊞ cleaned_weather_information
      ⊞ exploded_raw_station_information
      ⊞ raw_station_information
      ⊞ raw_station_status
      ⊞ raw_weather_information
  > 🗄 tpch_lineage_demo
> 🗄 migrationtestrs

Storage Credentials   ⌄

Catalogs > main > divvy_bikes >

⊞ **main.divvy_bikes.agg_top_stations_and_weather_last_30_days** Delta△

📄 Comment:

⊛ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions   History   Lineage Preview

Filter columns...

Column	Type ⇕	Comment
ᴬᴮc station_name	string	✎
ᴬᴮc station_id	string	✎
ᴬᴮc station_type	string	✎
ᴬᴮc weather_station	string	✎
1.2 avg_temp_F	double	✎
1.2 avg_snow_1h	double	✎
1.2 avg_rain_1h	double	✎
1²₃ max_capacity	bigint	✎
1²₃ last_bikes_available	bigint	✎
1²₃ first_bikes_available	bigint	✎
1²₃ min_bikes_available	bigint	✎
1²₃ max_bikes_available	bigint	✎

Data Explorer

+ | Search tables | Browse DBFS | Primary SQL Endpoint (L)

Data

> hive_metastore
> samples
> asean_catalog
> bmathew_unity
> catalogauo
> deepaksekardemo
> delta_sharing_aa
> delta_sharing_david_radford
> deltasharing
> dsatty_uc
> field_demos
> jiri
> main
  > default
  > divvy_bikes
      ⊞ agg_top_stations_and_weather_last_30_days
      ⊞ cleaned_station_information
      ⊞ cleaned_station_status
      ⊞ cleaned_weather_information
      ⊞ exploded_raw_station_information
      ⊞ raw_station_information
      ⊞ raw_station_status
      ⊞ raw_weather_information
  > tpch_lineage_demo
> migrationtestrs

Storage Credentials

Catalogs > main > divvy_bikes >

⊞ **main.divvy_bikes.agg_top_stations_and_weather_last_30_days** Delta△

▤ Comment:

◎ Owner: richard.tomlinson@databricks.com ✎

Schema | **Sample Data** | Details | Permissions | History | Lineage Preview

ᴬᴮᴄ station_name	ᴬᴮᴄ station_id	ᴬᴮᴄ station_type	ᴬᴮᴄ weather_station	1.2 avg_temp_F	1.2 avg_snow_1h	1.2 avg_rain_1h	1²₃ n
Canal St & Adams St	a3a77c8d-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	60.88999999988121	0	0	47
Millennium Park	a3a50ec8-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	77.21600000001428	0	0	47
Columbus Dr & Randolph St	a3a78bc0-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	81.60799999987215	0	0	55
Canal St & Adams St	a3a77c8d-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	57.90199999983916	0	0	47
Field Museum	a3a53324-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	89.00599999984802	0	0	55
Canal St & Adams St	a3a77c8d-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	57.88399999987915	0	0	47
Canal St & Adams St	a3a77c8d-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	69.13399999983578	0	0	47
Shedd Aquarium	a3a37378-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	59.576000000076306	0	0	55
Millennium Park	a3a50ec8-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	57.91999999993947	0	0	47
Shedd Aquarium	a3a37378-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	53.05999999989158	0	0	55
Columbus Dr & Randolph St	a3a78bc0-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	91.61600000021899	0	0	55
Millennium Park	a3a50ec8-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	84.75800000003332	0	0	47
Shedd Aquarium	a3a37378-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	62.3480000001636	0	0	55
Columbus Dr & Randolph St	a3a78bc0-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	97.55599999989761	0	0	55
Shedd Aquarium	a3a37378-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	87.18799999988643	0	0	55
Shedd Aquarium	a3a37378-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	73.61600000017782	0	0	55
Canal St & Adams St	a3a77c8d-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	97.60999999993456	0	0	47
Canal St & Adams St	a3a77c8d-a135-11e9-9cda-0a87ae2ba916	classic	Chicago	57.434000000092276	0	0	47

Data Explorer

Browse DBFS   🔷 Primary SQL Endpoint (L)

Data

- › 🗄 hive_metastore
- › 🗄 samples
- › 🗄 asean_catalog
- › 🗄 bmathew_unity
- › 🗄 catalogauo
- › 🗄 deepaksekardemo
- › 🗄 delta_sharing_aa
- › 🗄 delta_sharing_david_radford
- › 🗄 deltasharing
- › 🗄 dsatty_uc
- › 🗄 field_demos
- › 🗄 jiri
- › 🗄 main
  - › 🗃 default
  - › 🗃 divvy_bikes
    - ⊞ agg_top_stations_and_weather_last_30_days
    - ⊞ cleaned_station_information
    - ⊞ cleaned_station_status
    - ⊞ cleaned_weather_information
    - ⊞ exploded_raw_station_information
    - ⊞ raw_station_information
    - ⊞ raw_station_status
    - ⊞ raw_weather_information
  - › 🗃 tpch_lineage_demo
- › 🗄 migrationtestrs

Storage Credentials

Catalogs › main › divvy_bikes ›

⊞ main.divvy_bikes.agg_top_stations_and_weather_last_30_days Delta△

📄 Comment:

👤 Owner: richard.tomlinson@databricks.com ✎

Schema | **Sample Data** | **Details** | **Permissions** | **History** | **Lineage** Preview

Name	agg_top_stations_and_weather_last_30_days
Catalog Name	main
Schema Name	divvy_bikes
Table Type	MANAGED
Storage Location	abfss://unity@unitydemo.dfs.core.windows.net/b86c6879-8c55-4e70-a585-18d16a4fa6e9/tables/76a015c0-2d58-4c0d-9c9e-a0c6c4e62f9f
Owner	richard.tomlinson@databricks.com
Properties	delta.lastCommitTimestamp=1656002973000 delta.lastUpdateVersion=217 delta.minReaderVersion=1 delta.minWriterVersion=2
Metastore Id	b86c6879-8c55-4e70-a585-18d16a4fa6e9
Full Name	main.divvy_bikes.agg_top_stations_and_weather_last_30_days
Data Access Configuration Id	00000000-0000-0000-0000-000000000000
Created At	6/14/2022, 10:43:49 AM
Created By	richard.tomlinson@databricks.com
Updated At	6/23/2022, 11:49:35 AM
Updated By	richard.tomlinson@databricks.com
Table Id	76a015c0-2d58-4c0d-9c9e-a0c6c4e62f9f

Data Explorer

🔍 Search tables   Browse DBFS   ⬢ Primary SQL Endpoint (L)

Data   ⌃

> 🗄 hive_metastore
> 🗄 samples
> 🗄 asean_catalog
> 🗄 bmathew_unity
> 🗄 catalogauo
> 🗄 deepaksekardemo
> 🗄 delta_sharing_aa
> 🗄 delta_sharing_david_radford
> 🗄 deltasharing
> 🗄 dsatty_uc
> 🗄 field_demos
> 🗄 jiri
> 🗄 main
  > 🗄 default
  > 🗄 divvy_bikes
      ▦ agg_top_stations_and_weather_last_30_days
      ▦ cleaned_station_information
      ▦ cleaned_station_status
      ▦ cleaned_weather_information
      ▦ exploded_raw_station_information
      ▦ raw_station_information
      ▦ raw_station_status
      ▦ raw_weather_information
  > 🗄 tpch_lineage_demo
> 🗄 migrationtestrs

Storage Credentials   ⌄

Catalogs > main > divvy_bikes >

▦ main.divvy_bikes.agg_top_stations_and_weather_last_30_days Delta△

📄 Comment:

⊕ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   **Permissions**   History   Lineage **Preview**

[Grant] [Revoke]   Type a user name or a group name   ⌄

☐   **Principal**   ⇕   **Privilege**   ⇕

No permissions granted yet.

Data Explorer

Browse DBFS | Primary SQL Endpoint (L)

Search tables

Data

- hive_metastore
- samples
- asean_catalog
- bmathew_unity
- catalogauo
- deepaksekardemo
- delta_sharing_aa
- delta_sharing_david_radford
- deltasharing
- dsatty_uc
- field_demos
- jiri
- main
  - default
  - divvy_bikes
    - agg_top_stations_and_weather_last_30_days
    - cleaned_station_information
    - cleaned_station_status
    - cleaned_weather_information
    - exploded_raw_station_information
    - raw_station_information
    - raw_station_status
    - raw_weather_information
  - tpch_lineage_demo
- migrationtestrs

Storage Credentials

Catalogs > main > divvy_bikes >

⊞ main.divvy_bikes.agg_top_stations_and_weather_last_30_days Delta△

📄 Comment:

Owner: richard.tomlinson@databricks.com ✎

Schema | Sample Data | Details | Permissions | History | Lineage Preview

Version	Timestamp	User Id	User Name	Operation	Operation Parameters	Job
217	2022-06-23T16:49:33	6487310572433869	richard.tomlinson@databricks.com	CREATE OR REPLACE TABLE AS SELECT	▸ { ... } // 4 items	{"jobId":"435706635065724","jobName":"AGGS: Create UC Aggrega
216	2022-06-23T15:50:00	6487310572433869	richard.tomlinson@databricks.com	CREATE OR REPLACE TABLE AS SELECT	▸ { ... } // 4 items	{"jobId":"435706635065724","jobName":"AGGS: Create UC Aggrega
215	2022-06-23T14:50:11	6487310572433869	richard.tomlinson@databricks.com	CREATE OR REPLACE TABLE AS SELECT	▸ { ... } // 4 items	{"jobId":"435706635065724","jobName":"AGGS: Create UC Aggrega
214	2022-06-23T13:50:28	6487310572433869	richard.tomlinson@databricks.com	CREATE OR REPLACE TABLE AS SELECT	▸ { ... } // 4 items	{"jobId":"435706635065724","jobName":"AGGS: Create UC Aggrega
213	2022-06-23T12:50:05	6487310572433869	richard.tomlinson@databricks.com	CREATE OR REPLACE TABLE AS SELECT	▸ { ... } // 4 items	{"jobId":"435706635065724","jobName":"AGGS: Create UC Aggrega
212	2022-06-23T11:50:34	6487310572433869	richard.tomlinson@databricks.com	CREATE OR REPLACE TABLE AS	▸ { ... } // 4 items	{"jobId":"435706635065724","jobName":"AGGS: Create UC Aggrega

Data Explorer

+   🔍 Search tables   Browse DBFS   🔴 Primary SQL Endpoint (L)   ⊘ ⌄

Data   ⌃

- › 🗄 hive_metastore
- › 🗄 samples
- › 🗄 asean_catalog
- › 🗄 bmathew_unity
- › 🗄 catalogauo
- › 🗄 deepaksekardemo
- › 🗄 delta_sharing_aa
- › 🗄 delta_sharing_david_radford
- › 🗄 deltasharing
- › 🗄 dsatty_uc
- › 🗄 field_demos
- › 🗄 jiri
- › 🗄 main
  - › ⊟ default
  - › ⊟ divvy_bikes
    - ▦ agg_top_stations_and_weather_last_30_days
    - ▦ cleaned_station_information
    - ▦ cleaned_station_status
    - ▦ cleaned_weather_information
    - ▦ exploded_raw_station_information
    - ▦ raw_station_information
    - ▦ raw_station_status
    - ▦ raw_weather_information
  - › ⊟ tpch_lineage_demo
- › 🗄 migrationtestrs

Storage Credentials   ⌄

Catalogs › main › divvy_bikes ›

▦ main.divvy_bikes.agg_top_stations_and_weather_last_30_days   Delta△

📄 Comment:

⊕ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions   History   **Lineage** Preview

| See Lineage Graph | | Upstream | Downstream | Filter... |

	Table Name
▦ Tables	main.divvy_bikes.cleaned_station_information
▯ Notebooks	main.divvy_bikes.cleaned_station_status
⊱ Workflows	main.divvy_bikes.cleaned_weather_information
▪ Dashboards	

1

main.divvy_bikes.**cleaned_station_information**

main.divvy_bikes.**cleaned_station_status**

main.divvy_bikes.**cleaned_weather_information**

main.divvy_bikes.**agg_top_stations_and_weather_last_30_days**

Data Explorer

Browse DBFS | ⬡ Primary SQL Endpoint (L) | ⊘ ∨

Data ∧

- > ⊟ hive_metastore
- > ⊟ samples
- > ⊟ asean_catalog
- > ⊟ bmathew_unity
- > ⊟ catalogauo
- > ⊟ deepaksekardemo
- > ⊟ delta_sharing_aa
- > ⊟ delta_sharing_david_radford
- > ⊟ deltasharing
- > ⊟ dsatty_uc
- > ⊟ field_demos
- > ⊟ jiri
- > ⊟ main
  - > ⧉ default
  - > ⧉ divvy_bikes
    - ⊞ agg_top_stations_and_weather_last_30_days
    - ⊞ cleaned_station_information
    - ⊞ cleaned_station_status
    - ⊞ cleaned_weather_information
    - ⊞ exploded_raw_station_information
    - ⊞ raw_station_information
    - ⊞ raw_station_status
    - ⊞ raw_weather_information
  - > ⧉ tpch_lineage_demo
- > ⊟ migrationtestrs

Storage Credentials ∨

Catalogs > main > divvy_bikes >

⊞ **main.divvy_bikes.agg_top_stations_and_weather_last_30_days** Delta△

📄 Comment:

⊕ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions   History   **Lineage** Preview

[See Lineage Graph]

Upstream | Downstream | Filter...

| Tables |
| Notebooks |
| Workflows |
| Dashboards |

Notebook Name

Create Divvy Bike Tables for UC ⧉

1

Data Explorer

+   🔍 Search tables   Browse DBFS   ⬡ Primary SQL Endpoint (L)   ⊘ ⌄

Data   ⌃

> 🗄 hive_metastore
> 🗄 samples
> 🗄 asean_catalog
> 🗄 bmathew_unity
> 🗄 catalogauo
> 🗄 deepaksekardemo
> 🗄 delta_sharing_aa
> 🗄 delta_sharing_david_radford
> 🗄 deltasharing
> 🗄 dsatty_uc
> 🗄 field_demos
> 🗄 jiri
> 🗄 main
    > 🗄 default
    > 🗄 divvy_bikes
        ⊞ agg_top_stations_and_weather_last_30_days
        ⊞ cleaned_station_information
        ⊞ cleaned_station_status
        ⊞ cleaned_weather_information
        ⊞ exploded_raw_station_information
        ⊞ raw_station_information
        ⊞ raw_station_status
        ⊞ raw_weather_information
    > 🗄 tpch_lineage_demo
> 🗄 migrationtestrs

Storage Credentials   ⌄

Catalogs › main › divvy_bikes ›

⊞ main.divvy_bikes.agg_top_stations_and_weather_last_30_days Delta△

📄 Comment:

◉ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions   History   **Lineage** Preview

[See Lineage Graph]   Upstream | Downstream   Filter...

⊞ Tables

□ Notebooks

☰ Workflows

▦ Dashboards

Job Name

AGGS: Create Aggregate Tables ⧉

1

Data Explorer

+   Search tables   Browse DBFS   ⬢ Primary SQL Endpoint (L)   ⊘ ⌄

Data   ⌃

> ⊟ hive_metastore
> ⊟ samples
> ⊟ asean_catalog
> ⊟ bmathew_unity
> ⊟ catalogauo
> ⊟ deepaksekardemo
> ⊟ delta_sharing_aa
> ⊟ delta_sharing_david_radford
> ⊟ deltasharing
> ⊟ dsatty_uc
> ⊟ field_demos
> ⊟ jiri
> ⊟ main
>> ⊟ default
>> ⊟ divvy_bikes
>>> ⊞ agg_top_stations_and_weather_last_30_days
>>> ⊞ cleaned_station_information
>>> ⊞ cleaned_station_status
>>> ⊞ cleaned_weather_information
>>> ⊞ exploded_raw_station_information
>>> ⊞ raw_station_information
>>> ⊞ raw_station_status
>>> ⊞ raw_weather_information
>> ⊟ tpch_lineage_demo
> ⊟ migrationtestrs

Storage Credentials   ⌄

Catalogs > main > divvy_bikes >

⊞ main.divvy_bikes.agg_top_stations_and_weather_last_30_days Delta△

📄 Comment:

⊕ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions   History   **Lineage** Preview

| See Lineage Graph | Upstream | **Downstream** | Filter... |

	Dashboard Name
⊞ Tables	
▯ Notebooks	Divvy Bike Finance Dashboard ⧉
☰ Workflows	Divvy Bike Marketing Dashboard ⧉
▦ **Dashboards**	

1

Data Explorer

+ 🔍 Search tables   Browse DBFS   🔴 Primary SQL Endpoint (L)   ⊘ ⌄

Data   ⌃

> 🗄 hive_metastore
> 🗄 samples
> 🗄 asean_catalog
> 🗄 bmathew_unity
> 🗄 catalogauo
> 🗄 deepaksekardemo
> 🗄 delta_sharing_aa
> 🗄 delta_sharing_david_radford
> 🗄 deltasharing
> 🗄 dsatty_uc
> 🗄 field_demos
> 🗄 jiri
> 🗄 main
  > 🗃 default
  > 🗃 divvy_bikes
      ⊞ agg_top_stations_and_weather_last_30_days
      ⊞ cleaned_station_information
      ⊞ cleaned_station_status
      ⊞ cleaned_weather_information
      � exploded_raw_station_information
      ⊞ raw_station_information
      ⊞ raw_station_status
      ⊞ raw_weather_information
  > 🗃 tpch_lineage_demo
> 🗄 migrationtestrs

Storage Credentials   ⌄

Catalogs > main > divvy_bikes >

⊞ main.divvy_bikes.cleaned_station_status Delta⊿

📄 Comment: The cleaned station status data with valid station_ids and partitioned by station id.

◉ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions   History   Lineage Preview

Filter columns...

Column	Type ⇕	Comment
ᴬᴮ𝒸 station_id	string	✎
¹²₃ num_bikes_available	bigint	✎
¹²₃ num_bikes_disabled	bigint	✎
¹²₃ num_docks_available	bigint	✎
¹²₃ num_docks_disabled	bigint	✎
¹²₃ num_ebikes_available	bigint	✎
ᴬᴮ𝒸 station_status	string	✎
¹²₃ is_renting	bigint	✎
¹²₃ is_returning	bigint	✎
¹²₃ last_reported	bigint	✎
📅 last_reported_ts	timestamp	✎
¹²₃ last_updated	bigint	✎

Data Explorer

num_bikes_available ✕

Data ⌃

> ▤ hive_metastore
> ▤ samples
> ▤ asean_catalog
> ▤ bmathew_unity
> ▤ catalogauo
> ▤ deepaksekardemo
> ▤ delta_sharing_aa
> ▤ delta_sharing_david_radford
> ▤ deltasharing
> ▤ dsatty_uc
> ▤ field_demos
> ▤ jiri
> ▤ main
  > ⊟ default
  > ⊟ divvy_bikes
    ⊞ agg_top_stations_and_weather_last_30_days
    ⊞ cleaned_station_information
    ⊞ cleaned_station_status
    ⊞ cleaned_weather_information
    ⊞ exploded_raw_station_information
    ⊞ raw_station_information
    ⊞ raw_station_status
    ⊞ raw_weather_information
  > ⊟ tpch_lineage_demo
> ▤ migrationtestrs

Storage Credentials ⌄

Catalogs > main > divvy_bikes >

⊞ **main.divvy_bikes.cleaned_static**

📄 Comment: The cleaned station status data with valid st

ⓣ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions

Filter columns...

Column

ᴬᴮᑦ station_id

¹²₃ num_bikes_available

¹²₃ num_bikes_disabled

¹²₃ num_docks_available

¹²₃ num_docks_disabled

¹²₃ num_ebikes_available

ᴬᴮᑦ station_status

¹²₃ is_renting

¹²₃ is_returning

¹²₃ last_reported

🗓 last_reported_ts

¹²₃ last_updated

Column lineage Preview

| See Lineage Graph | Upstream | Downstream | Filter... |

| **Tables** | | |

Table Name		Column Name
main.divvy_bikes.raw_station_status		_rescued_data
main.divvy_bikes.raw_station_status		data
main.divvy_bikes.raw_station_status		last_updated
main.divvy_bikes.raw_station_status		ttl

1

Data Explorer

Data ∧

> hive_metastore
> samples
> asean_catalog
> bmathew_unity
> catalogauo
> deepaksekardemo
> delta_sharing_aa
> delta_sharing_david_radford
> deltasharing
> dsatty_uc
> field_demos
> jiri
> main
  > default
  > divvy_bikes
    ⊞ agg_top_stations_and_weather_last_30_days
    ⊞ cleaned_station_information
    ⊞ cleaned_station_status
    ⊞ cleaned_weather_information
    ⊞ exploded_raw_station_information
    ⊞ raw_station_information
    ⊞ raw_station_status
    ⊞ raw_weather_information
  > tpch_lineage_demo
> migrationtestrs

Storage Credentials ∨

Catalogs > main > divvy_bikes >

⊞ **main.divvy_bikes.cleaned_static**

📄 Comment: The cleaned station status data with valid st

⊕ Owner: richard.tomlinson@databricks.com ✎

Schema   Sample Data   Details   Permissions

Filter columns...

Column

ᴬᴮc station_id

¹²₃ num_bikes_available

¹²₃ num_bikes_disabled

¹²₃ num_docks_available

¹²₃ num_docks_disabled

¹²₃ num_ebikes_available

ᴬᴮc station_status

¹²₃ is_renting

¹²₃ is_returning

¹²₃ last_reported

🗓 last_reported_ts

¹²₃ last_updated

---

num_bikes_available ✕

Column lineage Preview

[See Lineage Graph]

⊞ Tables

| | Upstream | **Downstream** | Filter... |

Table Name	Column Name
main.divvy_bikes.agg_top_stations_and_weather_last_30_days	avg_bikes_available
main.divvy_bikes.agg_top_stations_and_weather_last_30_days	first_bikes_available
main.divvy_bikes.agg_top_stations_and_weather_last_30_days	last_bikes_available
main.divvy_bikes.agg_top_stations_and_weather_last_30_days	max_bikes_available
main.divvy_bikes.agg_top_stations_and_weather_last_30_days	min_bikes_available

1

SQL

SQL query editor

Create a new query and explore your data in a SQL editor.

Create a query

Sample dashboards

Explore sample dashboards containing rich visualizations and queries.

Visit gallery

Sample data

Analyze a collection of pre-loaded data samples.

Open the Data Explorer

Partner Connect

Fivetran, dbt

Tableau, Power BI

View all partners

Recents

Name	Last viewed
Divvy Bike Hourly Forecast	25 minutes ago
Divvy Bike Real-Time Monitoring	27 minutes ago
New query	21 hours ago
Divvy Bike Finance Dashboard	2 days ago
Divvy Bike Marketing Dashboard	2 days ago
UC agg query 1	2 days ago
Divvy Bike Station Trends	2 days ago
Station & Weather Headlines for Now from Whenever	2 days ago
Station Map Basic from Whenever	2 days ago
Prediction DOW	6 days ago

Favorites

Name	Created at
Divvy Bike Real-Time Monitoring	2022-01-08 10:50
Divvy Bike Station Trends	2022-02-16 16:23
Divvy Bike DLT Data Quality Monitoring	2022-03-23 11:03
Divvy Bike Hourly Forecast	2022-06-17 12:17

Documentation

Databricks SQL user guide

Learn how to generate, visualize and share insights using the built-in SQL editor, dashboards and alerts

Databricks SQL adminstration guide

Learn how to administer Databricks SQL and connect your data to popular third-party BI Tools

SQL reference for Databricks SQL

Learn about the SQL commands supported in Databricks SQL

Release notes

Databricks SQL release notes

Platform release notes

Blog posts

Introducing Databricks SQL on Google Cloud – Now in Public Preview

May 10, 2022

Get to Know Your Queries With the New Databricks SQL Query Profile!

February 23, 2022

Building Data Applications on the Lakehouse With the Databricks SQL Connector for Python

databricks

S SQL   ▾

⊕ Create

▭ SQL Editor

▱ Queries

▦ Dashboards

△ Alerts

▱ Data

⬡ SQL Warehouses

◷ Query History

QL query editor
reate a new query and explore your data in a
QL editor.

eate a query

Sample dashboards
Explore sample dashboards containing rich
visualizations and queries.

Visit gallery

Sample data
Analyze a collection of pre-loaded data
samples.

Open the Data Explorer

Partner Connect
Fivetran, dbt

Tableau, Power BI

View all partners

Favorites

		Last viewed		Name	Created at
	ourly Forecast	26 minutes ago	▦	Divvy Bike Real-Time Monitoring	2022-01-08 10:50
	eal-Time Monitoring	27 minutes ago	▦	Divvy Bike Station Trends	2022-02-16 16:23
		21 hours ago	▦	Divvy Bike DLT Data Quality Monitoring	2022-03-23 11:03
	inance Dashboard	2 days ago	▦	Divvy Bike Hourly Forecast	2022-06-17 12:17
	larketing Dashboard	2 days ago			
	y 1	2 days ago			
	tation Trends	2 days ago			
	eather Headlines for Now from Whenever	2 days ago			
	Basic from Whenever	2 days ago			
	OW	6 days ago			

⬠ Partner Connect

⊘ Help

⚙ Settings

◯ dbdemo
    richard.tomlinson@d...

▯ Menu options

r guide
te, visualize and share insights using the
ashboards and alerts

minstration guide
ster Databricks SQL and connect your
-party BI Tools

Databricks SQL

Release notes

Databricks SQL release notes

Platform release notes

Blog posts

Introducing Databricks SQL on Google Cloud – Now in
Public Preview
May 10, 2022

Get to Know Your Queries With the New Databricks SQL
Query Profile!
February 23, 2022

Building Data Applications on the Lakehouse With the

SQL Warehouses

Create SQL Warehouse

Name	State	Size	Active / Max	Actions
Primary SQL Endpoint	⊘ Running	Large	1 / 1	◉ Stop ⋮
FIVETRAN_ENDPOINT	◉ Stopped	X-Small	0 / 1	▶ Start ⋮
Secondary SQL Endpoint	◉ Stopped	Medium	0 / 1	▶ Start ⋮
Starter Endpoint	◉ Stopped	Small	0 / 1	▶ Start ⋮

Schema browser   Past executions

never   ⊘ Weather Desc for Whenever   ⊘ Station & Weather Headlines for N...   ⊘ Copy of: Availability for Last 24hrs...   ⊘ Forecast Map   ⊘ UC agg query 1   ✕   ⊘ New query   ...   +

🔷 Secondary SQL Endpoint (M)   ⟳ ▾   ⋮   Share   Save   ► Run (limit 1000) ▾

hive_metast... > divvy_bi...   ⟳

Filter tables & columns...

```
 1  SELECT
 2      station_name,
 3      station_type,
 4      AVG(avg_temp_F),
 5      AVG(avg_snow_1h),
 6      AVG(avg_rain_1h),
 7      AVG(avg_bikes_available),
 8      AVG(avg_ebikes_available),
 9      timestamp_last_updated_hour
10  FROM
11      main.divvy_bikes.agg_top_stations_and_weather_last_30_days
12  GROUP BY
13      station_name,
14      station_type,
15      timestamp_last_updated_hour
```

⊞ __apply_changes_storage_...
⊞ agg_station_status_weather
⊞ agg_station_status_weath...
⊞ agg_station_status_weath...
⊞ agg_station_status_weath...
⊞ agg_station_status_weath...
⊞ agg_top_stations_and_we...
⊞ cleaned_station_geo
⊞ cleaned_station_information
⊞ cleaned_station_status
⊞ cleaned_weather_informati...
⊞ forecast_prediction_0264...
⊞ forecast_prediction_1407...
⊞ forecast_prediction_output1
⊞ raw_station_geo
⊞ raw_station_information
⊞ raw_station_status
⊞ raw_weather_information

{} ⊟ ⚡ ⌨   ⟳ Refresh schedule Never

Table ⋮   + Add visualization

station_name	station_type	avg(avg_temp_F)	avg(avg_snow_1h)	avg(avg_rain_1	avg(avg_bikes_available)	avg(avg_ebikes_available)	timestamp_last_updated_h
Field Museum	classic	47.28	0.00	0.00	0.00	0.00	2022-05-23 (
Field Museum	classic	64.35	0.00	0.00	1.00	0.00	2022-06-18 '
Shedd Aquarium	classic	71.33	0.00	0.00	11.80	6.34	2022-06-11 '
Field Museum	classic	54.36	0.00	0.00	3.00	1.00	2022-05-27 '
Field Museum	classic	73.35	0.00	0.00	1.48	0.00	2022-06-21 '
Field Museum	classic	82.47	0.00	0.00	2.17	0.00	2022-06-17 (
Columbus Dr & Randolph St	classic	93.25	0.00	0.00	1.02	0.23	2022-06-15 '
Canal St & Adams St	classic	67.66	0.00	0.00	9.12	3.60	2022-05-28 '
Field Museum	classic	73.22	0.00	0.00	1.00	1.00	2022-05-31 '
Field Museum	classic	72.99	0.00	0.00	1.00	0.00	2022-06-17 (

1   2   3   4   5   · · ·   40   >

Last fetch: just now ⟳   ⋮ ⊟ ✎ Edit visualization   1000 rows 3.09 s runtime   Refreshed 2 days ago

Bar 1 ✕

Visualization type

📊 Bar	⌄

General   X axis   Y axis   Series   Colors   Data labels

☐ Horizontal chart

X column

🗓 timestamp_last_updated_hour	⌄		None	⌄

Y columns ⋮

1.2 avg(avg_ebikes_available)	⌄		Sum	⌄		-

Add column

Group by

Choose column...	⌄

Error column

Add column

Stacking

	⌄

☐ Normalize values to percentage

Missing and NULL values

Convert to 0 and display in chart	⌄

Cancel   **Save**
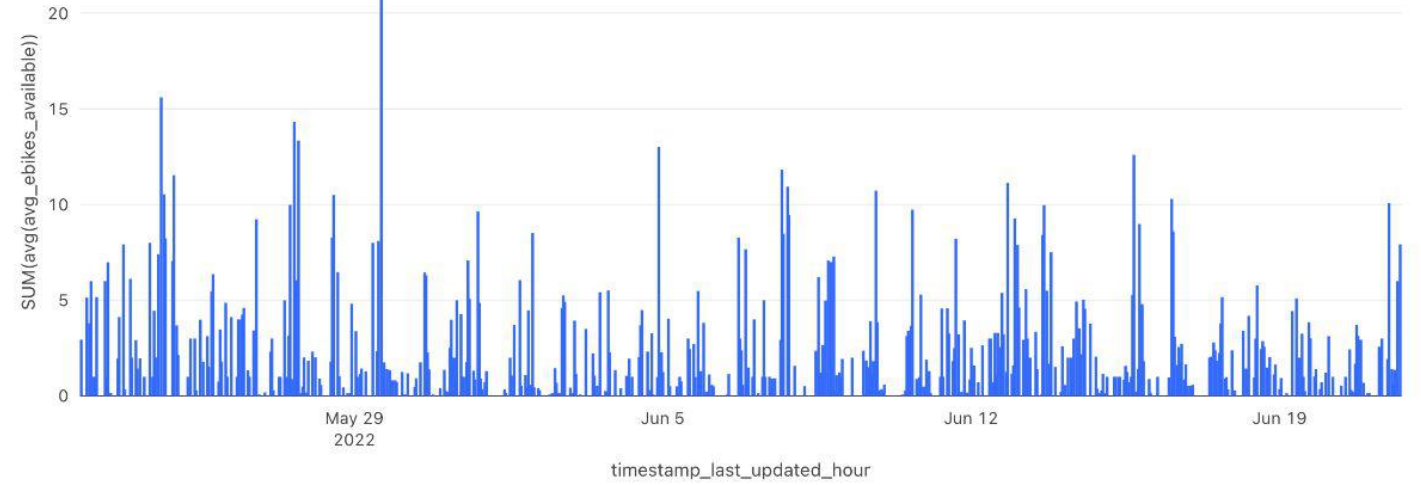
Schema browser   Past executions

hive_metast... > divvy_bi...

Filter tables & columns...

⊞ __apply_changes_storage_...
⊞ agg_station_status_weather
⊞ agg_station_status_weath...
⊞ agg_station_status_weath...
⊞ agg_station_status_weath...
⊞ agg_station_status_weath...
⊞ agg_top_stations_and_we...
⊞ cleaned_station_geo
⊞ cleaned_station_information
⊞ cleaned_station_status
⊞ cleaned_weather_informati...
⊞ forecast_prediction_0264...
⊞ forecast_prediction_1407...
⊞ forecast_prediction_output1
⊞ raw_station_geo
⊞ raw_station_information
⊞ raw_station_status
⊞ raw_weather_information

...never   ⊘ Weather Desc for Whenever   ⊘ Station & Weather Headlines for N...   ⊘ Copy of: Availability for Last 24hrs...   ⊘ Forecast Map   ⊘ UC agg query 1 ✕   ⊘ New query   ··· +

Secondary SQL Endpoint (M)   Share   Save   ► Run (limit 1000)

```sql
1   SELECT
2     station_name,
3     station_type,
4     AVG(avg_temp_F),
5     AVG(avg_snow_1h),
6     AVG(avg_rain_1h),
7     AVG(avg_bikes_available),
8     AVG(avg_ebikes_available),
9     timestamp_last_updated_hour
10  FROM
11    main.divvy_bikes.agg_top_stations_and_weather_last_30_days
12  GROUP BY
13    station_name,
14    station_type,
15    timestamp_last_updated_hour
```

Refresh schedule Never

Table ⋮   Bar 1 ⋮   + Add visualization
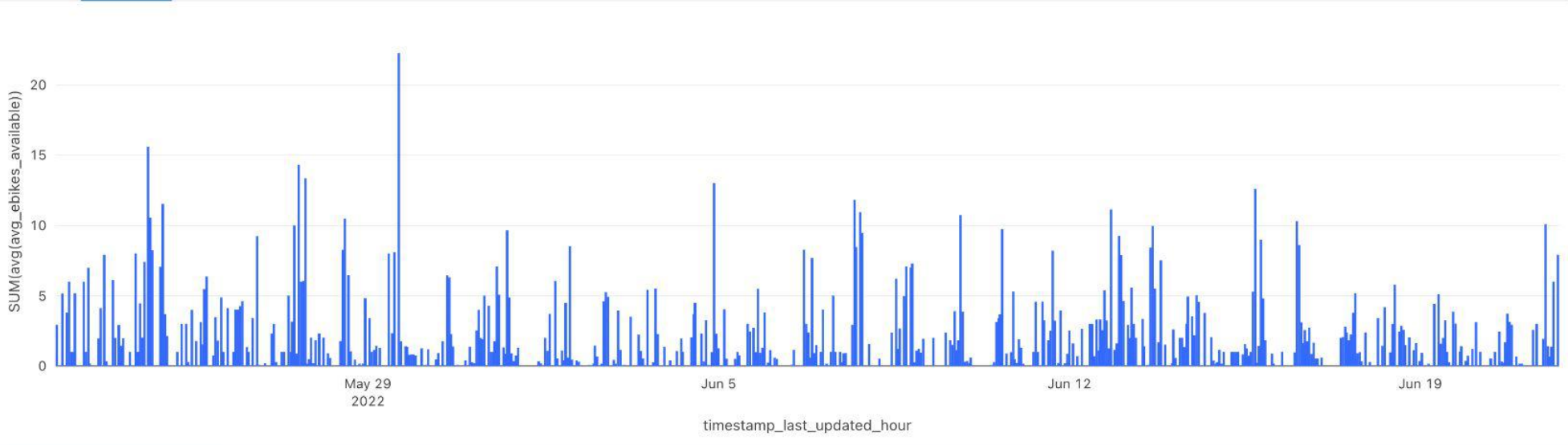
⋮   ✎ Edit visualization   582 aggregated rows   Refreshed 2 days ago

Last fetch: a minute ago ↻

Dashboards

Create dashboard

| Name ⇕	Created by	Created at ⇕
Divvy Bike Finance Dashboard divvy-bikes delta-live-tables finance	Richard Tomlinson	2022-06-21 15:42
Divvy Bike Marketing Dashboard delta-live-tables divvy-bikes marketing	Richard Tomlinson	2022-06-21 15:38
⭐ **Divvy Bike Hourly Forecast** divvy-bikes delta-live-tables automl	Richard Tomlinson	2022-06-17 12:17
Delta Live Tables - Data Quality Monitoring delta-live-tables monitoring real-time divvy-bikes	Richard Tomlinson	2022-05-19 12:03
⭐ **Divvy Bike DLT Data Quality Monitoring** delta-live-tables monitoring real-time divvy-bikes	Richard Tomlinson	2022-03-23 11:03
⭐ **Divvy Bike Station Trends** divvy-bikes delta-live-tables historical	Richard Tomlinson	2022-02-16 16:23
Retail Revenue & Supply Chain Sample	Richard Tomlinson	2022-01-08 14:24
NYC Taxi Trip Analysis Sample	Richard Tomlinson	2022-01-08 14:20
⭐ **Divvy Bike Real-Time Monitoring** real-time dlt auto-loader	Richard Tomlinson	2022-01-08 10:50

1   20 / page ⌄

Sample dashboards

Explore sample dashboards containing rich visualizations and queries. Visit gallery

Search dashboards...

■ All dashboards

👤 My dashboards

⭐ Favorites

🗑 Trash

👥 Admin view

Tags

delta-live-tables	6
divvy-bikes	6
real-time	3
Sample	2
monitoring	2
auto-loader	1
automl	1
dlt	1
finance	1
historical	1
marketing	1
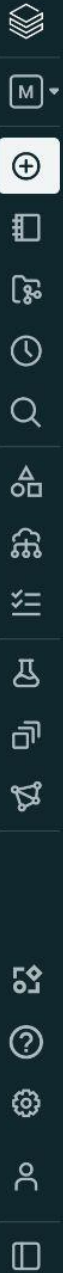
Machine Learning

Notebook
Create a notebook for querying, data processing, and ML.
Create a notebook

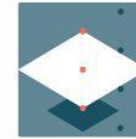**AutoML**
Quickly train ML models for discovery and iteration.
Start AutoML

Guide: Training
Get started with a tutorial on training and tuning ML models.
Start guide

Feature Store
Learn how to use the Feature Store.
Get started

Recents

Name	Type	Last viewed
sql-autoloader-divvybikes1	Notebook	20 minutes ago
Divvy Bike forecast for DAIS - 1	Experiment	2 days ago
Create Divvy Bike Tables for UC	Notebook	2 days ago
Create Divvy Bike Aggs for UC	Notebook	2 days ago
sql-aggs-divvybikes1	Notebook	2 days ago

1   2   3   >

Documentation

Getting started with machine learning on Databricks
Targeted tutorials for different machine learning settings

Machine learning and deep learning guide
Documentation for model training and inference on Databricks

MLflow guide
Managing the machine learning lifecycle on Databricks

Learn more about Databricks

Release notes

Runtime Release Notes

MLflow Release Notes

Platform Release Notes

More release notes

Blog posts

Architecting MLOps on the Lakehouse
Here at Databricks, we have helped thousands of customers put Machine Learning (ML) into production...

Build Reliable Production Data and ML Pipelines With Git Support for Databricks Workflows
We are happy to announce native support for Git in Databricks Workflows, which enables our customers to build reliable production data and ML workflows using modern software engineering best pract...

Introduction to Analyzing Crypto Data Using Databricks
The market capitalization of cryptocurrencies increased from $17 billion

databricks

M Machine Learning ▼

⊕ Create

🔲 Workspace

[ᵍ] Repos

🕘 Recents

🔍 Search

Data

🔗 Compute

☰ Workflows

⚗ Experiments

🗄 Feature Store

📡 Models

⚏ Partner Connect

❓ Help

⚙ Settings

👤 dbdemo
richard.tomlinson@d...

▢ Menu options

:arning

otebook

reate a notebook for querying,
ta processing, and ML.

reate a notebook

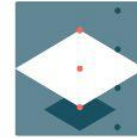**AutoML**

Quickly train ML models for
discovery and iteration.

Start AutoML

Guide: Training

Get started with a tutorial on
training and tuning ML models.

Start guide

Feature Store

Learn how to use the Feature
Store.

Get started

	Type	Last viewed
er-divvybikes1	Notebook	20 minutes ago
recast for DAIS - 1	Experiment	2 days ago
Bike Tables for UC	Notebook	2 days ago
Bike Aggs for UC	Notebook	2 days ago
vybikes1	Notebook	2 days ago

1 2 3 >

machine learning on Databricks

r different machine learning settings

nd deep learning guide

model training and inference on Databricks

ine learning lifecycle on Databricks

Databricks

Release notes

Runtime Release Notes

MLflow Release Notes

Platform Release Notes

More release notes

Blog posts

Architecting MLOps on the Lakehouse

Here at Databricks, we have helped thousands of customers put
Machine Learning (ML) into production...

Build Reliable Production Data and ML Pipelines With Git Support for
Databricks Workflows

We are happy to announce native support for Git in Databricks
Workflows, which enables our customers to build reliable production
data and ML workflows using modern software engineering best pract...

Introduction to Analyzing Crypto Data Using Databricks

The market capitalization of cryptocurrencies increased from $17 billion

Experiments

| Create AutoML Experiment ∨ | Compare (0) | | Owned by me | Accessible by me | 🔍 Search experiments | Search |

	Name ⇅	Location ⇅	Last Modified ⇅	Created by	Notes	Actions
☐	Divvy Bike forecast for DAIS - 1	/Users/richard.tomlinson@databricks.com/databricks_automl/Divvy Bike for...	2022-06-23 11:45:56 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-16 17:00:31 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-13 17:43:13 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-13 13:18:20 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-11 14:32:27 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-11 14:21:45 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-11 14:16:41 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-11 14:12:24 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-11 14:04:25 CDT	richard.tomlinson@databricks.com		⋮
☐	avg_bikes_available_agg_top_stations_and_wea...	/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_av...	2022-06-10 13:45:01 CDT	richard.tomlinson@databricks.com		⋮

1 > 10 / page ∨

Experiments > Configure AutoML experiment

Configure AutoML experiment

① Configure ——————— ② Train ——————————————— ③ Evaluate

AutoML Experiment Configuration ⌃

* Cluster (Databricks Runtime 8.3 ML or above) ⓘ

[Select a cluster | ⌄]

* ML problem type

[Classification   ⌄]

* Dataset

[Browse] [Select training data   ]

* Prediction target

[Select a prediction target   ⌄]

* Experiment name ⓘ

[Experiment name   ]

Advanced Configuration (optional) ⌄

[Start AutoML]

ⓘ **Certain AutoML features require a minimum runtime version.**

The latest Databricks Runtime version is recommended to get the newest features of AutoML. Learn more about AutoML.

Databricks Runtime for Machine Learning version	Feature
11.0	**AutoML can use more memory to process larger datasets** Options to increase sample size: use a cluster with memory optimized instance types; set `spark.task.cpus` on the cluster to a larger number (up to the number of CPUs in the work instance).
10.5	**Output forecast predictions** AutoML saves predictions of the best model to a new table in the specified database. **Improved memory usage** AutoML can train on larger datasets.
10.4	**Custom imputation of missing values** Specify how null values are imputed.
10.3	**Framework selection** Select ML algorithms to use for model development and evaluation. **Column selection** Specify which columns AutoML should use for training. **Auto-ARIMA** Forecasting support for ARIMA models.
10.1	**Warnings dashboard** Display warnings for potential issues with the dataset.
10.0	**Forecasting** A new problem type.

Experiments > Configure AutoML experiment

Configure AutoML experiment

① Configure ─────────── ② Train ─────────── ③ Evaluate

AutoML Experiment Configuration ∧

* Cluster (Databricks Runtime 8.3 ML or above) ⊙

| MLCluster1 | ∨ |

* ML problem type

| Forecasting | ∨ |

* Dataset

| Browse | divvy_bikes.agg_top_stations_and_weather_last_30_days |

* Prediction target

| avg_bikes_available | ∨ |

* Time column ⊙

| timestamp_last_updated_hour | ∨ |

Time series identifiers (for multi-series forecasting) ⊙

| station_name × |

Forecast horizon and frequency ⊙

| 10 | | Hours | ∨ |

Output Database ⊙

| Browse | Select output database |

* Experiment name ⊙

| avg_bikes_available_agg_top_stations_and_weather_last_30_days-2022_06_2 |

Schema:

Column name ⇅	Data type ⇅
station_name	string
station_id	string
station_type	string
weather_station	string
avg_temp_F	double
avg_snow_1h	double
avg_rain_1h	double
max_capacity	bigint
last_bikes_available	bigint
first_bikes_available	bigint
min_bikes_available	bigint
max_bikes_available	bigint
avg_bikes_available	double
last_ebikes_available	bigint
first_ebikes_available	bigint
min_ebikes_available	bigint
max_ebikes_available	bigint
avg_ebikes_available	double
timestamp_last_up...	timestamp

Experiments > /Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_available_agg_top_stations_and_weather_last_30_days-2022_06_23-12_17

/Users/richard.tomlinson@databricks.com/databricks_automl/avg_bikes_available_agg_top_stations_and_weather_last_30_days-2022_06_23-12_17 ⎘

⋮   Share

ⓘ Track machine learning training runs in experiments. Learn more   ✕

Experiment ID: 4279996897990329

▸ Description Edit

▾ AutoML

⊘ Configure ────────────────────────── ⟩ Train Starting AutoML...   ③ Evaluate
                                          We're getting things
                                          ready for training

Overview   Warnings (0)

⟳ Refresh   Compare   Delete   Download CSV⬇   ↓ Start Time ∨   All time ∨

☰ ⊞   ⚙ Columns   Only show differences ⬤ ❓   🔍 metrics.rmse < 1 and params.model = "tree"   Search   ≡ Filter   Clear

Showing 0 matching runs

	↓ Start Time	Duration	Run Name	User	Source	Version	Models	
☐								

No runs yet. Learn more about how to create ML model training
runs in this experiment.

Experiments > /Users/richard.tomlinson@databricks.com/databricks_automl/Divvy Bike forecast for DAIS - 1

/Users/richard.tomlinson@databricks.com/databricks_automl/Divvy Bike forecast for DAIS - 1

Share

ⓘ Track machine learning training runs in experiments. Learn more ✕

Experiment ID: 2732297975964096

▸ Description Edit

▾ AutoML

✓ Configure ——————————————— ✓ Train ——————————————— ③ Evaluate

Overview Warnings (0)

AutoML Evaluation ✓ complete
All runs have completed, and have been added to the table below. Click a specific run to view details or review the data exploration notebook.

Model with best val_smape
The model is ready to be registered and deployed. Or, access the source code for the model training to make modifications by clicking a notebook under the Source column in the table below.

[↗ View notebook for best model] [↗ View data exploration notebook]

[⟳ Refresh] [Compare] [Delete] [Download CSV ⬇] [↑ val_smape ⌄] [All time ⌄]

[☰] [⊞] [⚙ Columns] Only show differences ◯ ❓ 🔍 metrics.rmse < 1 and params.model = "tree" [Search] [≡ Filter] [Clear]

Showing 100 matching runs

	Start Time	Duration	Run Name	User	Source	Version	Models	Metrics				Parameters		Tags	
								val_coverage	val_mae	val_mape	↑ val_smape	holiday_countr	interval_width	estimator_nam	sparkD
☐	✓ 6 days ago	4.0min	PROPHET	richard.tom...	🗎 22-06-...	-	🗎 pyfunc	0.925	3.735	0.593	0.421	US	0.95	Prophet	-
☐	✓ 7 days ago	4.0min	PROPHET	richard.tom...	🗎 22-06-...	-	🗎 pyfunc	0.92	3.735	0.593	0.421	US	0.95	Prophet	-
☐	✓ 7 days ago	4.2min	PROPHET	richard.tom...	🗎 22-06-...	-	🗎 pyfunc	0.92	3.735	0.593	0.421	US	0.95	Prophet	-
☐	✓ 8 days ago	3.8min	PROPHET	richard.tom...	🗎 22-06-...	-	🗎 pyfunc	0.92	3.735	0.593	0.421	US	0.95	Prophet	-
☐	✓ 8 days ago	4.1min	PROPHET	richard.tom...	🗎 22-06-...	-	🗎 pyfunc	0.92	3.735	0.593	0.421	US	0.95	Prophet	-

22-06-14-16:43-Prophet-0d72d9970caff2f226fffce5b064d95a (Python)

🕐   Schedule ⌄   **Share**

⛓ Detached | ⌄   ▤ File ▾   ✏ Edit ▾   🖼 View: Standard ▾   ▶ Run All   ▱ Clear ▾   ⌨   💬 Comments   ⚗ Experiment   ↻ Revision history

Cmd 1

(Markdown) 📊 ⌄ — ✖

Prophet training

This is an auto-generated notebook. To reproduce these results, attach this notebook to the **MLCluster1** cluster and rerun it.

- Compare trials in the MLflow experiment
- Navigate to the parent notebook here (If you launched the AutoML experiment using the Experiments UI, this link isn't very useful.)
- Clone this notebook into your project folder by selecting **File > Clone** in the notebook toolbar.

Runtime Version: *10.5.x-cpu-ml-scala2.12*

Cmd 2

```python
1   import mlflow
2   import databricks.automl_runtime
3
4   target_col = "avg_bikes_available"
5   time_col = "timestamp_last_updated_hour"
6   unit = "hours"
7
8   id_cols = ["station_name"]
9
10  horizon = 10
```

Cmd 3

Load Data

Cmd 4

```python
1   from mlflow.tracking import MlflowClient
2   import os
3   import uuid
4   import shutil
5   import pandas as pd
6   import pyspark.pandas as ps
7
8   # Create temp directory to download input data from MLflow
9   input_temp_dir = os.path.join("/dbfs/tmp/", str(uuid.uuid4())[:8])
10  os.makedirs(input_temp_dir)
11
```

22-06-14-16:43-Prophet-0d72d9970caff2f226fffce5b064d95a (Python)

Schedule ∨   Share

Detached | ∨   📄 File ▾   📝 Edit ▾   🖼 View: Standard ▾   ▶ Run All   🩹 Clear ▾   ⌨   💬 Comments   ⚗ Experiment   🕑 Revision history
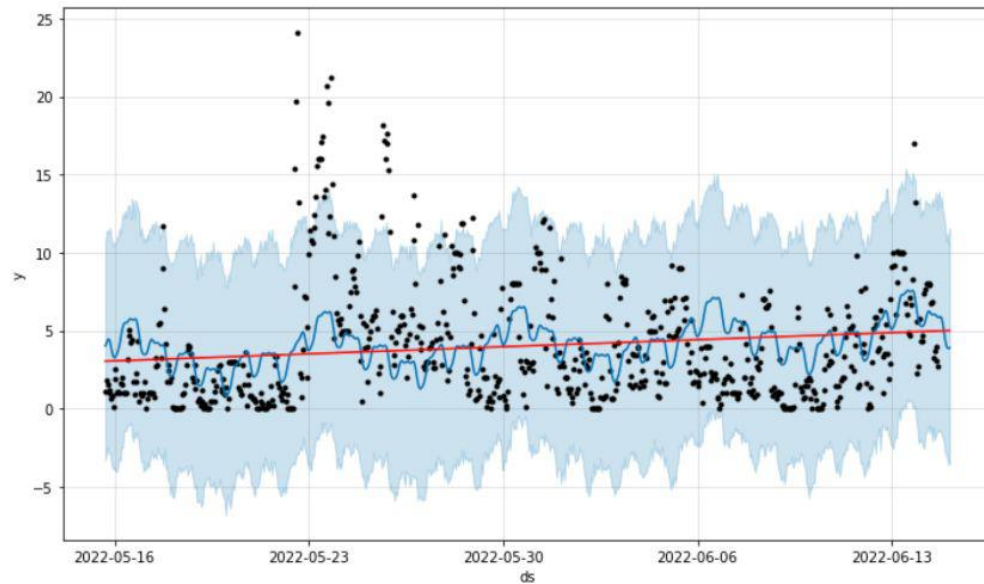
Cmd 18

Plot the forecast with change points and trend

Cmd 19

```python
from prophet.plot import add_changepoints_to_plot, plot_plotly

if use_plotly:
    fig = plot_plotly(model, predict_pd, changepoints=True, trend=True, figsize=(1200, 600))
else:
    fig = model.plot(predict_pd)
    a = add_changepoints_to_plot(fig.gca(), model, predict_pd)
fig
```

Out[12]:


```
<Figure size 720x432 with 1 Axes>
```

Cmd 20

22-06-14-16:43-Prophet-0d72d9970caff2f226fffce5b064d95a (Python)

Schedule ⌄   Share

⊞ Detached | ⌄   📄 File ▾   ✏️ Edit ▾   🖼️ View: Standard ▾   ▶ Run All   ✏️ Clear ▾   ⌨️ 💬 Comments 🧪 Experiment 🕘 Revision history
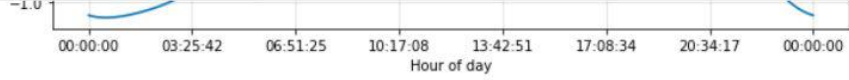
Cmd 22

Show the predicted results

Cmd 23

Python ▶▾ 📊 ⌄ — ✕

```python
1  predict_cols = ["ds", "ts_id", "yhat"]
2  forecast_pd = forecast_pd.reset_index()
3  display(forecast_pd[predict_cols].tail(horizon))
```

Table   Data Profile

	ds ▲	ts_id ▲	yhat ▲
1	2022-06-14T17:00:00.000+0000	Millennium Park	24.776106
2	2022-06-14T18:00:00.000+0000	Millennium Park	26.081785
3	2022-06-14T19:00:00.000+0000	Millennium Park	28.07161
4	2022-06-14T20:00:00.000+0000	Millennium Park	29.506031
5	2022-06-14T21:00:00.000+0000	Millennium Park	29.238937
6	2022-06-14T22:00:00.000+0000	Millennium Park	27.203274
7	2022-06-14T23:00:00.000+0000	Millennium Park	24.464035

Showing all 10 rows.

⊞ 📊 ▾ ⬇️

Shift+Enter to run

Experiments > /Users/richard.tomlinson@databricks.com/databricks_automl/Divvy Bike forecast for DAIS - 1

/Users/richard.tomlinson@databricks.com/databricks_automl/Divvy Bike forecast for DAIS - 1 ⧉

[⋮] [Share]

> ⓘ Track machine learning training runs in experiments. Learn more   ✕

Experiment ID: 2732297975964096

▶ Description   Edit

▼ AutoML

⊘ Configure ─────────────── 2 ──── ⊘ Train ──────────── 3 Evaluate

Overview   Warnings (0)

AutoML Evaluation ⊘ complete
All runs have completed, and have been added to the table below. Click a specific run to view details or review the data exploration notebook.

Model with best val_smape
The model is ready to be registered and deployed. Or, access the source code for the model training to make modifications by clicking a notebook under the Source column in the table below.

[⧉ View notebook for best model]   [⧉ View data exploration notebook]

[⟳ Refresh] [Compare] [Delete] [Download CSV ⬇] [↑ val_smape ∨] [All time ∨]

[☰] [⊞] [⚙ Columns]   Only show differences ◯   ⑦   [🔍 metrics.rmse < 1 and params.model = "tree"   Search] [≡ Filter] [Clear]

Showing 100 matching runs

								Metrics >					Parameters		Tags	
☐	Start Time	Duration	Run Name	User	Source	Version	Models	val_coverage	val_mae	val_mape	↑ val_smape	holiday_countr	interval_width	estimator_nam	sparkD
☐	⊘ 9 days ago	4.7min	PROPHET	richard.tom...	🔖 Noteboo...	-	🔖 pyfunc	0.93	3.735	0.593	0.421	US	0.95	Prophet	-
☐	⊘ 9 days ago	29.8s	ARIMA	richard.tom...	🔖 Noteboo...	-	🔖 pyfunc	0.923	3.775	0.512	0.425	-	-	ARIMA	-
☐	⊘ 9 days ago	3.5min	PROPHET	richard.tom...	🔖 Noteboo...	-	🔖 pyfunc	0.885	3.758	0.579	0.429	-	0.8	Prophet	-
☐	⊘ 9 days ago	3.3min	PROPHET	richard.tom...	🔖 Noteboo...	-	🔖 pyfunc	0.935	3.821	0.628	0.44	-	0.95	Prophet	-
☐	⊘ 17 hours ago	3.3min	PROPHET	richard.tom...	⋮≡ run 267...	-	🔖 pyfunc	0.97	2.595	1.466	0.487	US	0.95	Prophet	path=d

Experiments > /Users/richard.tomlinson@databricks.com/databricks_automl/Divvy Bike forecast for DAIS - 1 > PROPHET

PROPHET

⋮

Reproduce Run

Date: 2022-06-14 11:50:59

Source: 📄 Notebook: Prophet

User: richard.tomlinson@databricks.com

Duration: 4.7min

Status: FINISHED

Lifecycle Stage: active

▸ Description   Edit

▸ Parameters (2)

▸ Metrics (7)

▸ Tags (1)

▾ Artifacts

▾ 📁 model
  📄 MLmodel
  📄 conda.yaml
  📄 python_model.pkl
  📄 requirements.txt

Full Path:dbfs:/databricks/mlflow-tracking/2732297975964096/e76f1b998180473e9dc052fdd4ee5df9/artifacts/model 📋

Register Model

MLflow Model

The code snippets below demonstrate how to make predictions using the logged model. You can also register it to the model registry to version control and deploy as a REST endpoint for real time serving.

Model schema

Input and output schema for your model. Learn more

Name	Type

No schema. See MLflow docs for how to include input and output schema with your model.

Make Predictions

Predict on a Spark DataFrame: 📋

```
import mlflow
logged_model = 'runs:/e76f1b998180473e9dc052fdd4ee5df9/model'

# Load model as a Spark UDF. Override result_type if the model does not return double values.
loaded_model = mlflow.pyfunc.spark_udf(spark, model_uri=logged_model, result_type='double')

# Predict on a Spark DataFrame.
columns = list(df.columns)
```

Experiments > /Users/richard.tomlinson@databricks.com/databricks_automl/Divvy Bike forecast for DAIS - 1 > PROPHET

PROPHET

Reproduce Run

Date: 2022-06-14 11:50:59

Source: 📄 Notebook: Prophet

User: richard.tomlinson@databricks.com

Duration: 4.7min

Status: FINISHED

Lifecycle Stage: active

▸ Description   Edit

▸ Parameters (2)

▸ Metrics (7)

▸ Tags (1)

▼ Artifacts

▼ 📁 model
   📄 MLmodel
   📄 conda.yaml
   📄 python_model.pkl
   📄 requirements.txt

Full Path: dbfs:/databri͏...ts/model 📋

Register Model

Register Model   ✕

* Model

＋ Create New Model ⌄

* Model Name

DAIS123

Cancel   Register

MLflow Model

The code snippets below demonstrate how to make predictions using the logged model. You can also register it to the model registry to version control and deploy as a REST endpoint for real time serving.

Model schema

Input and output schema for your model. Learn more

Name	Type

No schema. See MLflow docs for how to include input and output schema with your model.

Make Predictions

Predict on a Spark DataFrame:

📋

```
import mlflow
logged_model = 'runs:/e76f1b998180473e9dc052fdd4ee5df9/model'

# Load model as a Spark UDF. Override result_type if the model does not return double valu
es.
loaded_model = mlflow.pyfunc.spark_udf(spark, model_uri=logged_model, result_type='doubl
e')

# Predict on a Spark DataFrame.
columns = list(df.columns)
```

Registered Models

Permissions

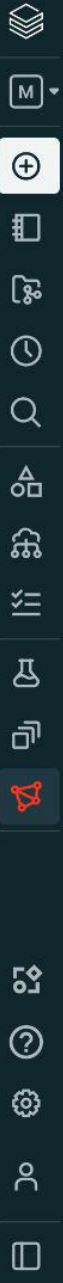ℹ️ Share and serve machine learning models. Learn more   ✕

Create Model

🔍 Search by model name   Search   ≡ Filter   Clear

Name	Latest Version	Staging	Production	Last Modified	Tags	Serving ❓
DAIS123	Version 1	–	–	2022-06-23 14:06:24	–	–
Hourly Forecast for Popular Stations 1	Version 1	–	–	2022-06-12 16:09:35	–	● Ready
titanic_packaged	Version 1	–	–	2022-05-27 15:42:38	–	–

‹ | 1 | › | 10 / page

Registered Models > DAIS123

DAIS123

⋮ Permissions **Use model for inference**

Details　　Serving

Notify me about ⓘ All new activity ⌄

Created Time : 2022-06-23 14:06:24　　　　　Last Modified : 2022-06-23 14:06:24　　　　　Creator : richard.tomlinson@databricks.com

▸ Description Edit

▸ Tags

▾ Versions | All | Active 0 | Compare

		Version	Registered at ▾	Created by	Stage	Pending Requests	Description
☐	⊘ 🔔	Version 1	2022-06-23 14:06:24	richard.tomlinson@databricks.com	None	—	

‹ **1** ›

Registered Models > DAIS123

DAIS123

⋮   Permissions   **Use model for inference**

Details   Serving

Notify me about ⓘ   All new activity ⌄

Created Time: 2022-06-23 14:06:24   Creator: richard.tomlinson@databricks.com

▶ Description   Edit

▶ Tags

▼ Versions   All   Active 0   Compare

	Version	Registered at		age	Pending Requests	Description
☐ ⊘ 🔔	Version 1	2022-06-23 14:06:24		one	—	

< 1 >

Set up model inference   Preview   ✕

Select either batch inference or real-time inference.

Batch inference   Real-time

Generates a notebook in your home folder that you can edit.

* Model version

Version 1   ⌄

* Input table

divvy_bikes.agg_top_stations_and_weather_last_30_days	Browse

* Output table location

/FileStore/batch-inference/	📁 DAIS123

The default output path on DBFS is accessible to everyone in this
Workspace. Modify the notebook to disable writing data to DBFS.

Cancel   **Use model for batch inference**

Inference-Autogenerated (Python)

Schedule ∨   **Share**

Detached | ∨   File ▾   Edit ▾   View: Standard ▾   Run All   Clear ▾   Comments   Experiment   Revision history

Cmd 1

Markdown

This is an auto-generated notebook to perform batch inference on a Spark DataFrame using a selected model from the model registry. This feature is in preview, and we would greatly appreciate any feedback through this form: https://databricks.sjc1.qualtrics.com/jfe/form/SV_1H6Ovx38zgCKAR0.

Instructions:

1. Run the notebook against a cluster with Databricks ML Runtime version 10.5.x-cpu, to best re-create the training environment.
2. Add additional data processing on your loaded table to match the model schema if necessary (see the "Define input and output" section below).
3. "Run All" the notebook.
4. Note: If the `%pip` does not work for your model (i.e. it does not have a `requirements.txt` file logged), modify to use `%conda` if possible.

Cmd 2

Python

```
1   model_name = "DAIS123"
```

Cmd 3

Environment Recreation

Run the notebook against a cluster with Databricks ML Runtime version 10.5.x-cpu, to best re-create the training environment.. The cell below downloads the model artifacts associated with your model in the remote registry, which include `conda.yaml` and `requirements.txt` files. In this notebook, `pip` is used to reinstall dependencies by default.

(Optional) Conda Instructions

Models logged with an MLflow client version earlier than 1.18.0 do not have a `requirements.txt` file. If you are using a Databricks ML runtime (versions 7.4-8.x), you can replace the `pip install` command below with the following lines to recreate your environment using `%conda` instead of `%pip`.

```
conda_yml = os.path.join(local_path, "conda.yaml")
%conda env update -f $conda_yml
```

Cmd 4

```
1   from mlflow.store.artifact.models_artifact_repo import ModelsArtifactRepository
2   import os
3
4   model_uri = f"models:/{model_name}/1"
5   local_path = ModelsArtifactRepository(model_uri).download_artifacts("") # download model from remote registry
6
```

Workflows

Jobs Job runs Delta Live Tables

Create Job   Owned by me   Accessible by me   🔍 Filter

	Name ⇕	Job ID ⇕	Created by ⇕	Task ⇕	Cluster ⇕	Schedule ⇕	Last run ⇕	Actions
●	INGEST: Get Station Availability	97	richard.tomlinson@databr...	realtime-divvybike-api-ingest-stationstatus	⊘ Single Node Jobs 1	Every minute (UTC)	Succeeded	▶ 🗑
●	INGEST: Get Station Weather Info	487	richard.tomlinson@databr...			At 1 minutes past th...	Running	■ 🗑
●	AGGS: Create Aggregate Tables	435706635...	richard.tomlinson@databr...	Create Divvy Bike Tables for UC	⊘ Photon Cluster Standard 2	At 45 minutes past t...	Succeeded	▶ 🗑
●	ML: Build Training Dataset and Retrain Model	106841396...	richard.tomlinson@databr...			At 45 minutes past t...	Succeeded	▶ 🗑

1-4 of 4 items   1

Workflows > Jobs > ML: Build Training Dataset and Retrain Model

ML: Build Training Dataset and Retrain Model

⋮   **Run now** ⌄

Runs   Tasks

Runs

Jun 22, 13:45 - Jun 23, 13:45   ‹ ›   Table   **Matrix**

Job Runs

4m 56s

2m 28s

Jun 22, 13:45   Jun 22, 19:45   Jun 23, 01:45   Jun 23, 07:45   Jun 23, 13:45

Tasks
Refresh_Model_Training_Data...
Retrain_Prophet_Model
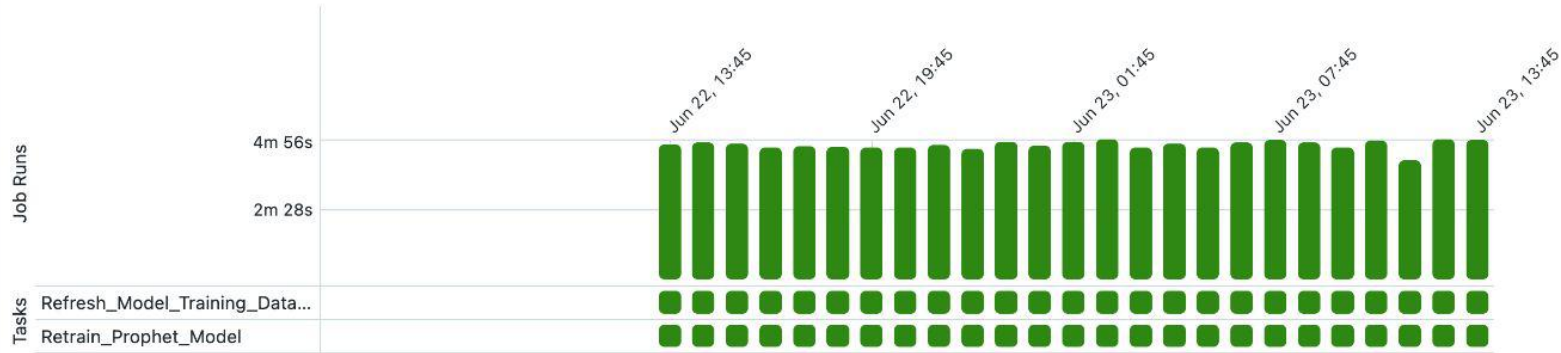
Job details

Job ID	1068413960741001 ⎘
Creator	richard.tomlinson@databricks.com
Run as ❓	richard.tomlinson@databricks.com
Tags ❓	+ Tag

Git

Add Git settings

Schedule

At 45 minutes past the hour (UTC+00:00 — UTC)

Edit schedule   Pause

Clusters

⊘ Photon Cluster Standard 2

Driver: Standard_E16_v3, Workers: Standard_E16_v3, 2-16 workers,
10.4 LTS Photon (includes Apache Spark 3.2.1, Scala 2.12)

View cluster   Swap   Spark UI   Logs   Metrics

⊘ MLCluster1

Driver: Standard_DS3_v2, Workers: Standard_DS3_v2, 2-8 workers,
10.5 ML (includes Apache Spark 3.2.1, Scala 2.12)

View cluster   Swap   Spark UI   Logs   Metrics

Notifications ❓

No notifications

★ **Divvy Bike Hourly Forecast**   divvy-bikes   delta-live-tables   automl ✏️   Share   Schedule ⌄   ↻ Refresh ⌄   ⋮
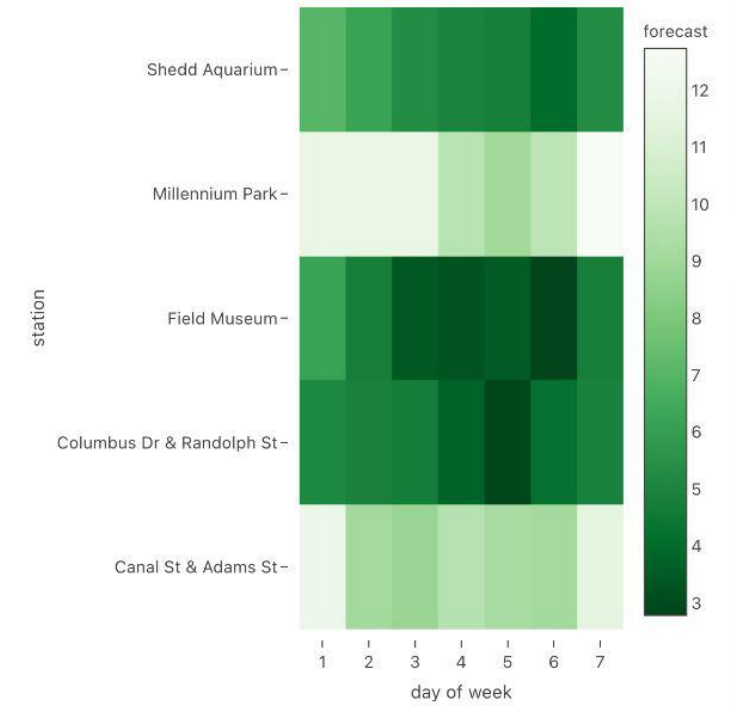
DIVVY®

Station Availability Forecast - Next 10 Hours

Forecasted availability for next 10 hours (Trend)
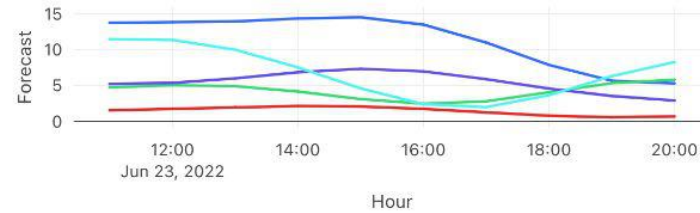
↻ just now

### Forecasted Availability by Day of Week

↻ just now

Forecasted availability for next 10 hours (by Station)

station_name	day_hr	2022-06-23 12:00	2022-06-23 13:00	2022-06-23 11:00	2022-06-23 14:00	2022-06-23 15:00	2022-06-23 16:00	2022-06-23 20:00	2022-06-23 17:00	2022-06-23 19:00	2022-06-23 18:00
Canal St & Adams St		13.87	13.96	13.77	14.37	14.53	13.52	5.31	11.00	5.67	7.88
Columbus Dr & Randolph St		1.78	2.02	1.54	2.15	2.08	1.75	0.70	1.26	0.57	0.79
Field Museum		5.05	4.92	4.75	4.17	3.11	2.46	5.82	2.81	5.34	4.03
Millennium Park		5.38	6.01	5.25	6.87	7.34	6.98	2.91	5.89	3.55	4.58
Shedd Aquarium		11.38	10.01	11.48	7.54	4.61	2.40	8.27	2.00	6.32	3.65

↻ just now

Forecast vs Actual Availability (Last 30 days + next 10 hours)
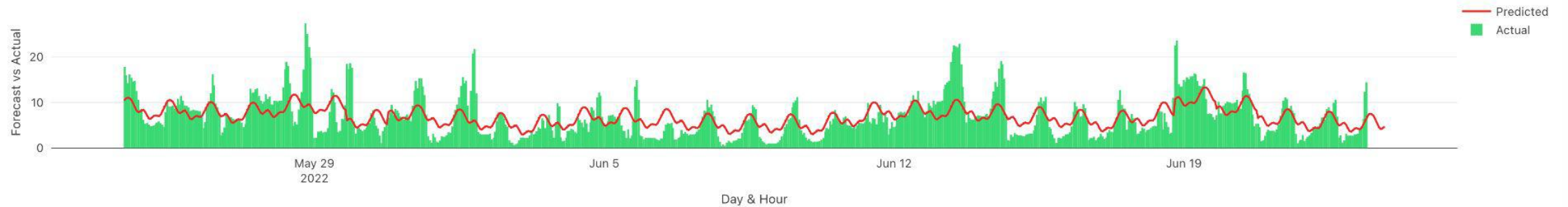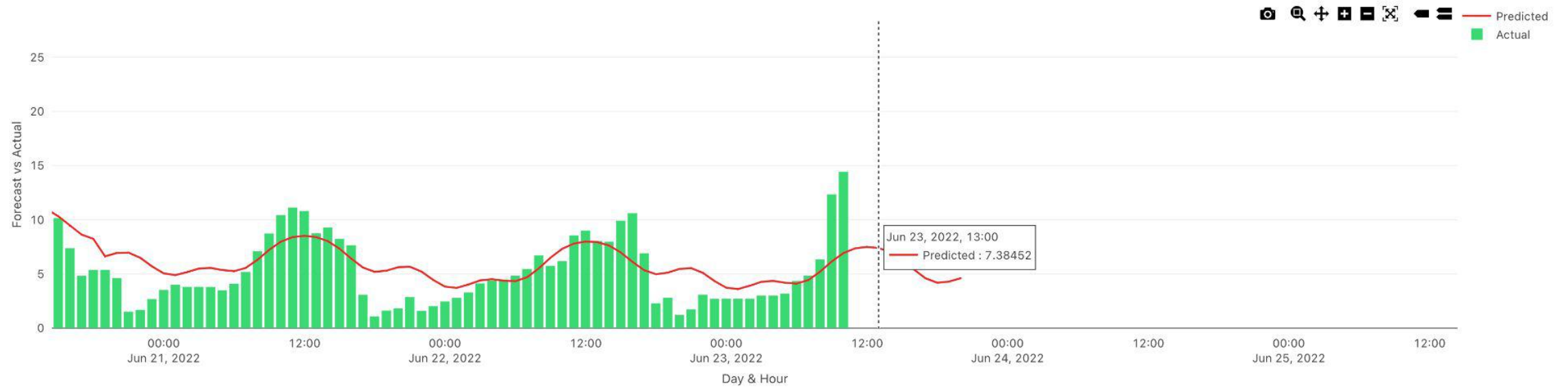

↻ just now

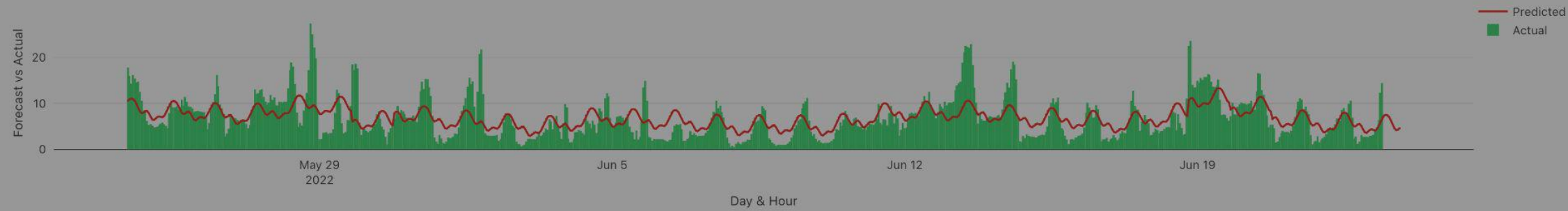⭐ **Divvy Bike Hourly Forecast** divvy-bikes delta-live-tables automl ✏️

Share   Schedule ∨   ↻ Refresh ∨   ⋮

Forecast vs Actual Availability (Last 30 days + next 10 hours)   ✕

— Predicted
■ Actual

Jun 23, 2022, 13:00
— Predicted : 7.38452

Forecast vs Actual

25
20
15
10
5
0

00:00   12:00   00:00   12:00   00:00   12:00   00:00   12:00   00:00   12:00
Jun 21, 2022   Jun 22, 2022   Jun 23, 2022   Jun 24, 2022   Jun 25, 2022

Day & Hour

Close

↻ just now

Forecast vs Actual Availability (Last 30 days + next 10 hours)

— Predicted
■ Actual

Forecast vs Actual

20
10
0

May 29   Jun 5   Jun 12   Jun 19
2022

Day & Hour

↻ just now

Partner Connect

Try partner solutions within minutes, even if you don't have a partner account.

Search by partner name ⌄

All categories ⌄

Data ingestion

Fivetran

Fivetran automated data integration adapts as schemas and APIs change, ensuring reliable data access and simplified analysis with ready-to-query schemas.

arcion

Arcion unlocks the value of transactional and operational data with real-time, distributed CDC that has built-in heterogeneous schema management, HA, and auto-scaling.

Rivery

Rivery provides a cloud-native ELT+ platform that accelerates the entire Databricks workflow through data ingestion, transformation, orchestration, and reverse ETL.

Data preparation and transformation

dbt

dbt enables teams to collaborate on data transformation following software engineering best practices like modularity, testing, and version control for increased productivity.

Prophecy

Prophecy is a low code product to visually build Apache Spark workflows backed by code on Git and includes metadata search, lineage, and scheduling.

MATILLION

Matillion makes the world's data useful. Unleash the full benefits and value of your Databricks Platform with Matillion ETL - the only purpose-built data integration platform for AWS and Azure.

BI and visualization

Microsoft Power BI

Quickly find meaningful insights within your data and easily build rich, visual analytic reports.

+ableau

Tableau helps people see and understand data with the world's broadest and deepest analytics platform.

HEX

Hex is a modern Data Workspace for teams. With Hex, Analysts and Data Scientists can explore, analyze, and visualize in collaborative data

Partner Connect

Try partner solutions within minutes, even if you don't have a partner account.

Search by partner name ▽   All categories ▽

Data ingestion

Fivetran

Fivetran automated data integration adapts as schemas and APIs change, ensuring reliable data access and simplified analysis with ready-to-query schemas.

Arcion unlocks the v
operational data wit
that has built-in het
management, HA, a

Connect to partner   ✕

dbt

dbt enables teams to collaborate on data transformation following software engineering best practices like modularity, testing, and version control for increased productivity.

By clicking Next, Databricks will create the following resources.

User ⓘ   DBT_CLOUD_USER

Personal access token ⓘ   <Autogenerated and managed>

Compute ⓘ   DBT_CLOUD_ENDPOINT (X-Small)   Edit

Cancel   Next

Data preparation and transformation

dbt

dbt enables teams to collaborate on data transformation following software engineering best practices like modularity, testing, and version control for increased productivity.

Prophecy is a low co
Apache Spark workf
and includes metad
scheduling.

BI and visualization

Microsoft Power BI

Quickly find meaningful insights within your data and easily build rich, visual analytic reports.

tableau

Tableau helps people see and understand data with the world's broadest and deepest analytics platform.

HEX

Hex is a modern Data Workspace for teams. With Hex, Analysts and Data Scientists can explore, analyze, and visualize in collaborative data