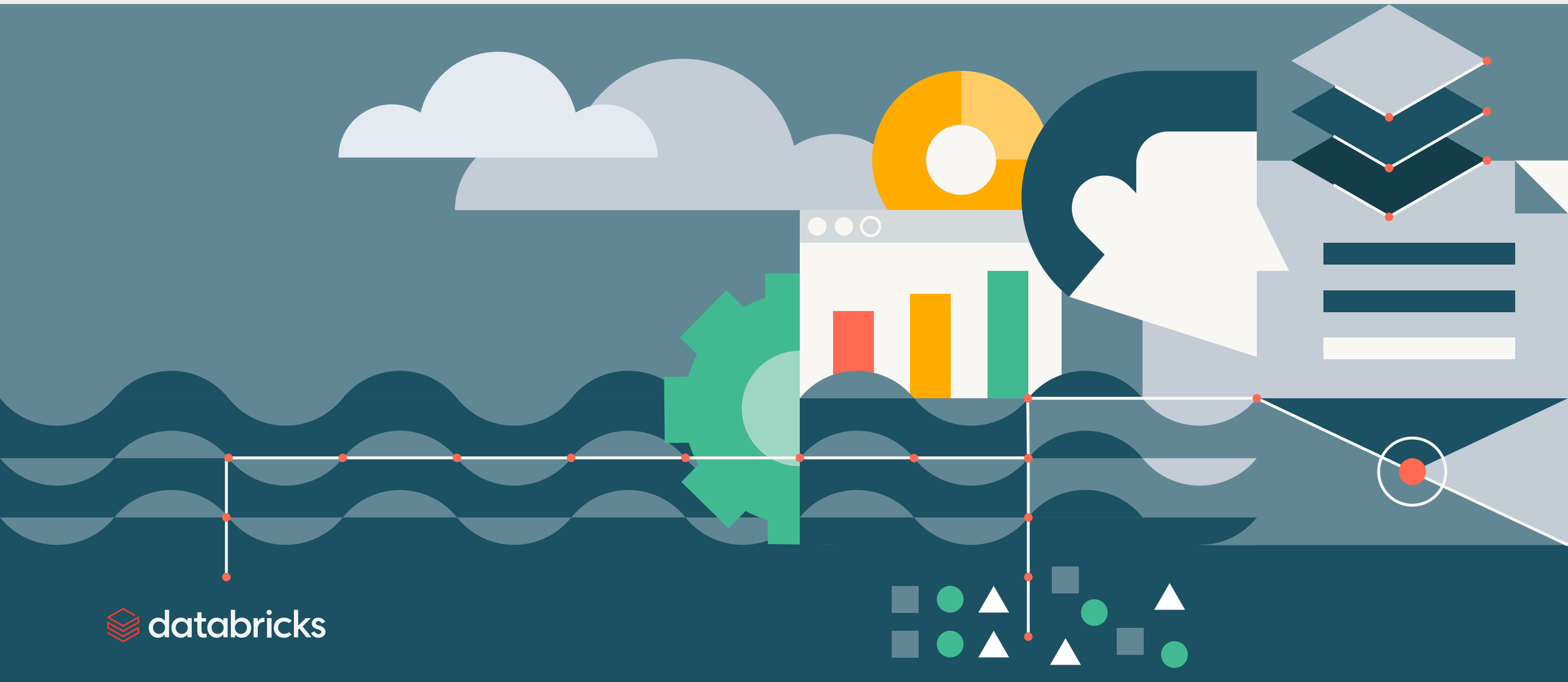
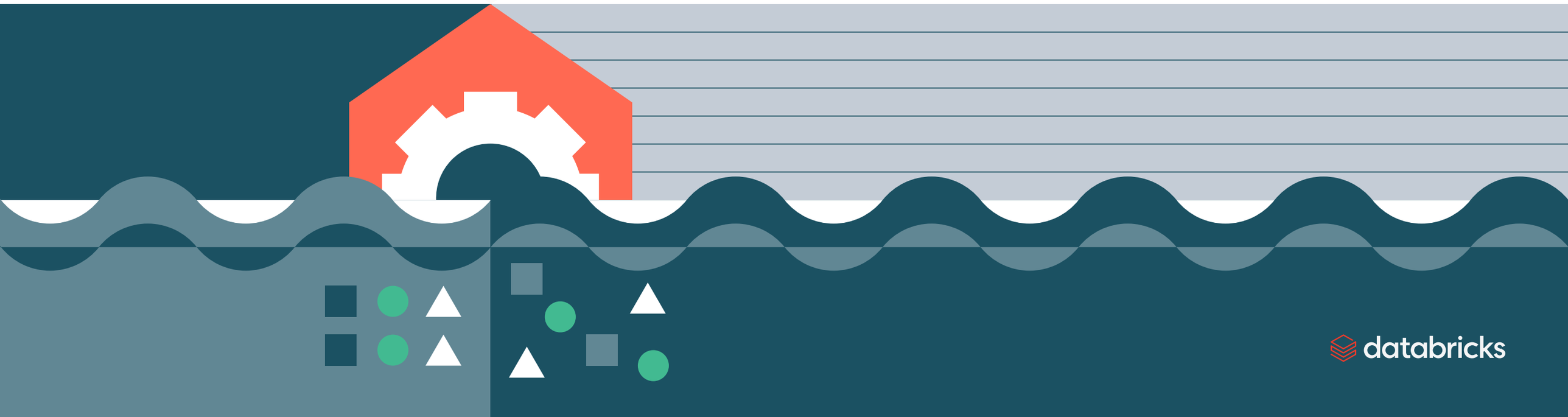
eBook

A Compact Guide to Building on Databricks



databricks

Contents



Introduction to the Lakehouse

A lakehouse is an architectural approach that combines the best elements of data lakes and data warehouses to help you reduce costs and deliver your data and AI initiatives faster. It simplifies your data strategy by eliminating the silos that historically complicate data and AI.

The Databricks Data Intelligence Platform, built on the lakehouse architecture, is a unified, scalable and open platform for storing and analyzing data. It fosters collaboration across teams by enabling sharing of data assets while also maintaining strict security and governance. Teams can scale the workloads they need for everything from data pipelines to AI. In addition, organizations avoid lock-in with the open architecture and can utilize the best advances in the future.

Why choose a lakehouse?

In the past, most of the data that went into a company's products or decision-making was structured data from operational systems, whereas today, many products incorporate AI in the form of computer vision and speech models, text mining, and others. A lakehouse gives you data versioning, governance, security and ACID properties that are needed even for unstructured data in one unified platform.

Having multiple systems leads to unconnected data silos, complicated pipelines or both. Developers spend time and effort maintaining these systems rather than creating new value, and downstream consumers struggle to get a single source of truth. This fragmented system becomes very expensive, and decision-making speed and quality are negatively impacted.

Therefore, unifying these systems with a lakehouse architecture can be transformational for organizations seeking to unlock the value of their data.

What can you build on the Databricks Data Intelligence Platform?

Users apply the data in a lakehouse to derive value for their organization. Whether you are writing queries to questions like "How much did we sell last month?" or analyzing video feed for defective parts, the Databricks Data Intelligence Platform can help. Common data assets that uses build include:

- Tables
- Reports
- Dashboards
- Pipelines
- Machine learning models

From developing simple scripts to sophisticated machine learning, the Databricks Platform provides the means to develop these products in a cost-effective, performant and governed manner.

In this guide, we walk through how to choose the right development tools for your data asset needs, how to get those assets into production, and best practices to follow along the way.

An example:

Movie Data Analysis

In this compact guide, we will use various tools on the Databricks Data Intelligence Platform to answer questions about a movie dataset from "The Movie Database (TMDB)." This dataset contains both quantitative data and qualitative data on 10,000 movies — and like most datasets you will encounter, it has numbers, strings, dates and missing values.

As we walk through examples to answer key questions, you can use the accompanying notebook to try out these tools yourself. Download the notebook with the test data here. **Let's get started!**

Getting Started

We will begin in your Databricks workspace. If you don't have one, your administrator can set one up for you, or you can create your own with the free trial. The workspace is a virtual place where your team can collaborate on data projects. If you have access to more than one workspace in the same account, you can switch among them in the UI.

Homepage

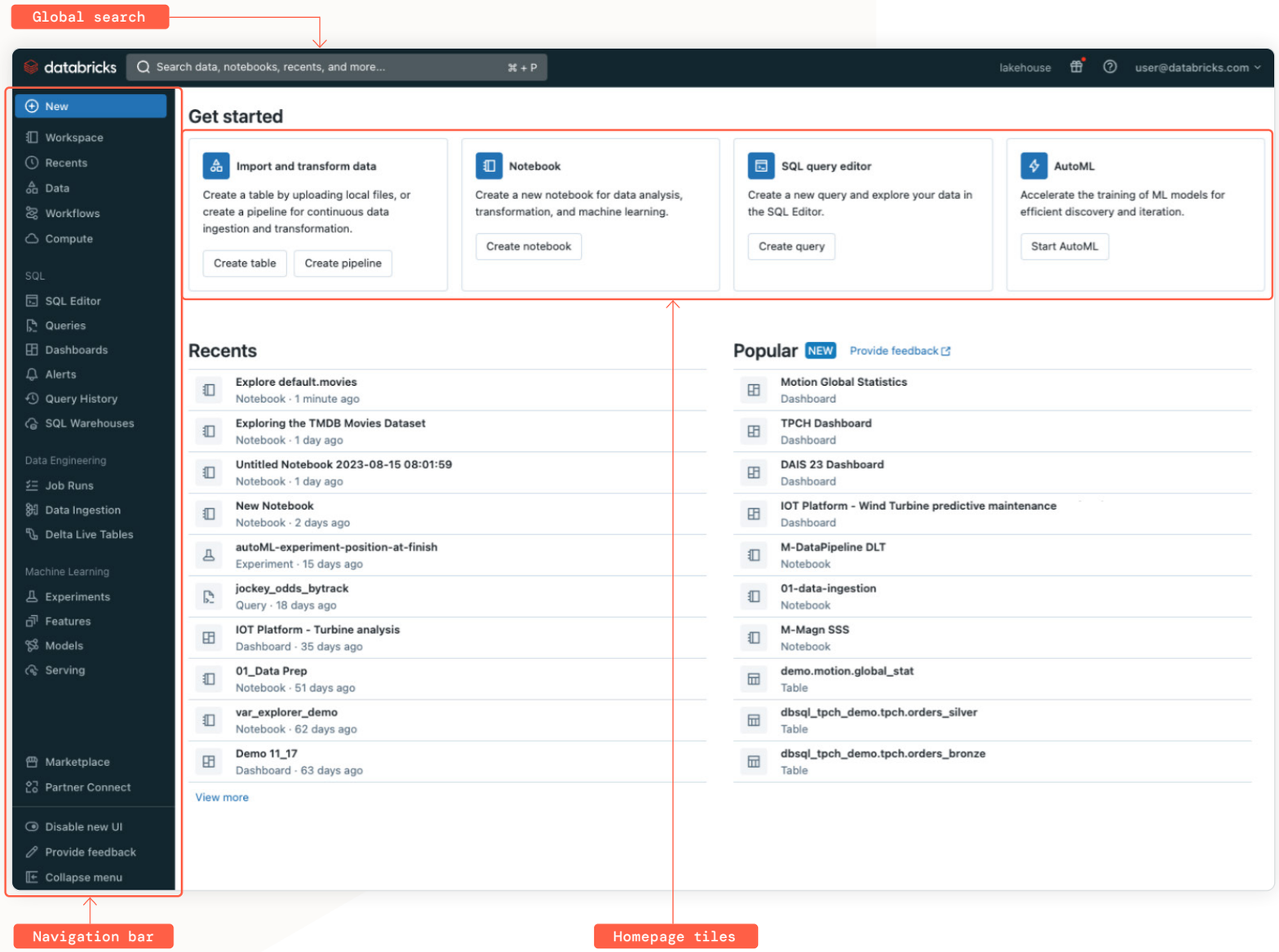
The homepage has tiles for the most common tasks — ingesting data, creating a notebook, making a query or training an ML model. Recents makes it easy to resume your work, and Popular helps you discover the most used data assets within your organization.

Global search

In one place, you can locate notebooks, tables, queries, dashboards, alerts, libraries, folders, repos and files from a single, always-accessible location. You can see contextual metadata and filter results.

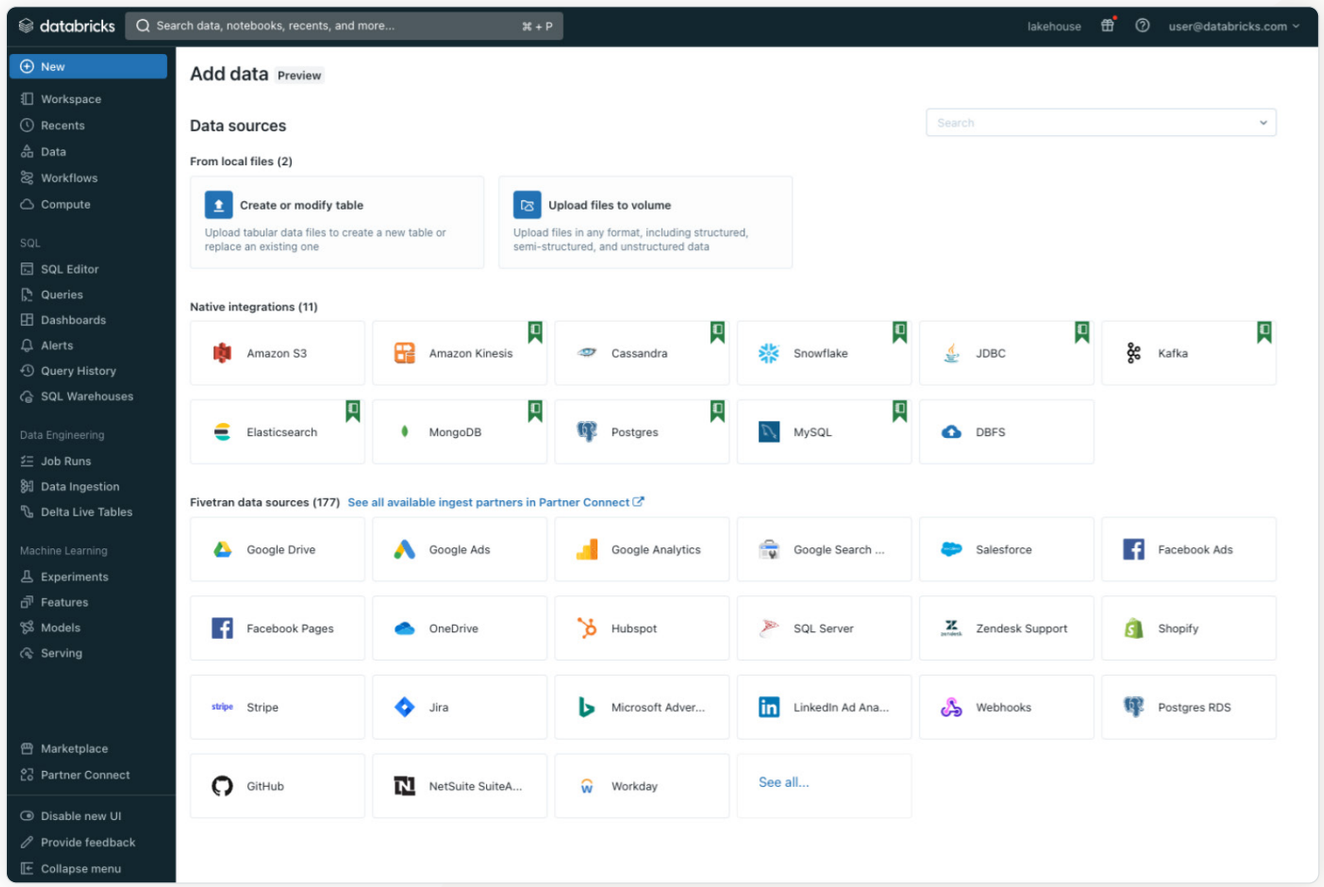
Navigation bar

On the left, you can access all your Databricks tools. You can expand or collapse the sidebar.



Adding Data

There are a number of ways to add data to Databricks for analysis. The Databricks Data Intelligence Platform supports unstructured, semi-structured and structured data and stores that data in cloud object storage in AWS, Azure or GCP. You can interact with these files through the Databricks Filesystem (DBFS).



Add data from a variety of sources to the Databricks Data Intelligence Platform.

Unity Catalog

Databricks Unity Catalog is the unified governance solution for data and AI on the lakehouse. It allows organizations to govern their data, machine learning models, notebooks, dashboards and files on any cloud or platform. You can use Unity Catalog to securely collaborate on trusted data and AI assets.

Create a table

This is the simplest way to upload local CSV, TSV or JSON files and store them in a Delta Lake table. Delta Lake is the open source storage format used by default for tables on Databricks.

Upload to Volume

You can upload files in any format to a Volume, including structured, semi-structured and unstructured data. While much of the data in a lakehouse is governed through tables, there are many use cases, particularly for machine learning and data science workloads, which require access to non-tabular data, such as text, image, audio, video, PDF or XML files. You can also store libraries, certificates and other configuration files.

Load data into Databricks using third-party tools

Databricks validates technology partner integrations that enable you to load data into Databricks. These integrations enable low-code, scalable data ingestion from a variety of sources. Some technology partners are featured in Databricks Partner Connect, which provides a UI that simplifies connecting third-party tools to your lakehouse data.

Load data from external cloud storage

You can add data from external sources through the add data UI. Once you provide a cloud storage location and credentials, Auto Loader processes new data files as they arrive in cloud storage without additional setup.

Accessing Compute

Databricks separates data storage and compute, and you must have both to process and analyze data.
Choosing your compute resources will depend on your project's needs for convenience, speed, scale and cost.

Serverless compute

Serverless compute removes the need to manually configure your compute. There is no need to spin it up or shut it down after use, getting you to insights faster and eliminating expensive idle time. Serverless compute resources include SQL Serverless and Model Serving. Serverless is the easiest way to get started if it's available to you. More serverless capabilities and regions will be rolled out in the future.

Personal compute

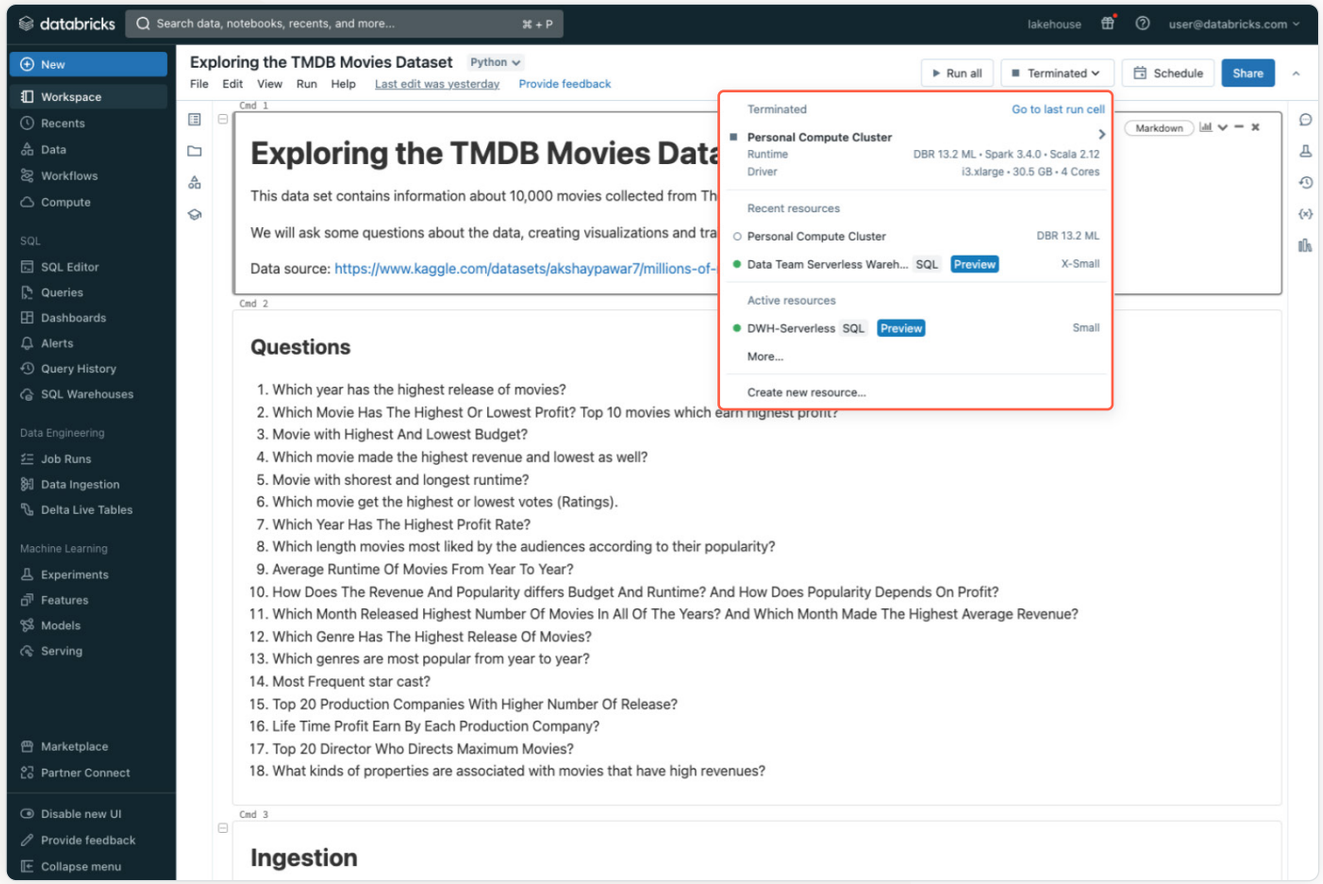
Personal compute is a Databricks-managed default cluster policy available on all Databricks workspaces. The policy allows you to easily create single-machine compute resources for individual use so you can start running workloads immediately.

Compute clusters

A Databricks cluster is an optimized Apache Spark™ compute resource that can be configured to fit the specific needs for your workloads. Multiple users can share a cluster. Databricks has all-purpose clusters and job clusters to fit the needed task. All-purpose clusters are for analyzing data, and job clusters are for running automated jobs.

SQL warehouses

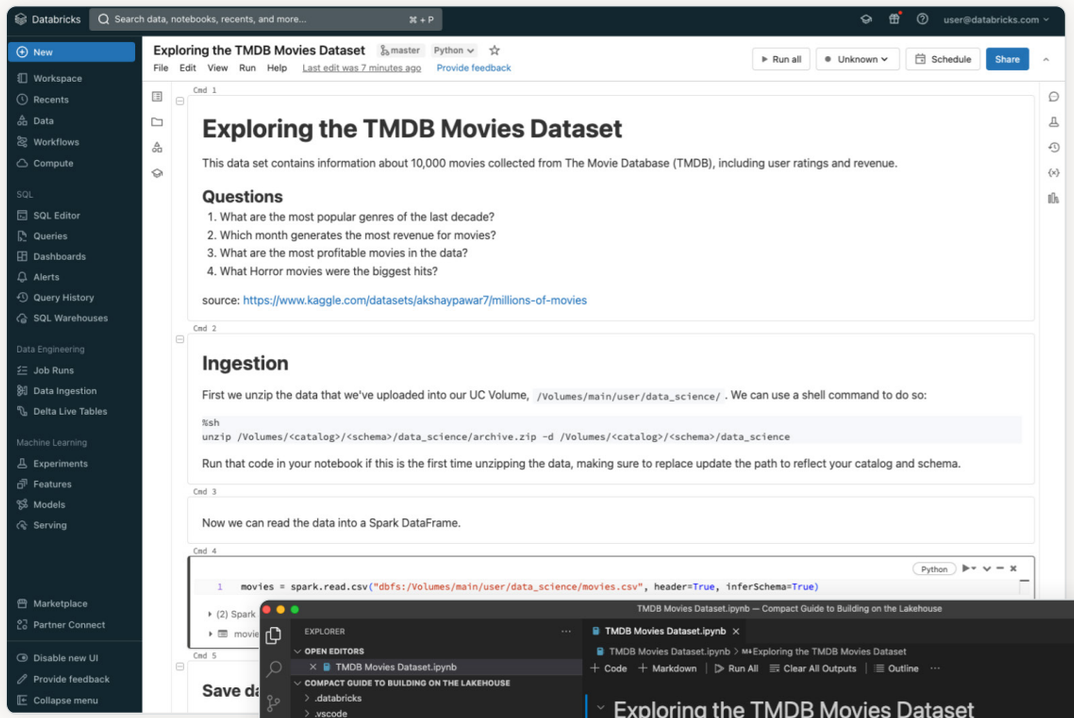
A SQL warehouse lets you run SQL commands on data within the SQL editor and Notebooks. This compute resource delivers better price/performance for SQL compared with all-purpose clusters. This type of compute only executes SQL commands. Cells using other languages (like Python or Scala) will be skipped. Markdown cells will continue to be rendered.



Select from available compute resources to run your data workloads.

Writing Code

There are several ways you can author code and build data assets on Databricks.
Users often use the editor they are most familiar with and that has the integrations they need.



Databricks Notebooks

Notebooks are interactive documents that combine code, visualizations and narrative text. Databricks Notebooks offer lakehouse-aware tools like Databricks Assistant and facilitate collaboration across teams. They have productivity features from popular IDEs like debugging and autocomplete-as-you-type.

File editor

You can use the workspace UI to perform tasks like creating, importing and editing files that enable you to develop modular software with Notebooks. File editor has a similar editing experience to Notebooks.

SQL editor

The SQL editor is a simple yet powerful editor to write, run and visualize SQL queries. Many users choose to develop with the SQL editor due to its familiar and streamlined user experience.

Integrated Development Environments (IDEs)

Databricks offers a native extension for VS Code. Other IDEs like PyCharm and RStudio are enabled through Databricks Connect. IDEs give you the scale of Databricks with a powerful toolset, including commonly used tools for refactoring, code navigation, local unit testing and more, that teams use to be more productive.

Exploring Your Data

You can analyze data and build reports to answer questions about your data. Databricks offers native features for data exploration and integrates with third-party services.

Databricks Assistant

Databricks Assistant is a context-aware AI assistant, available in Databricks Notebooks, SQL editor and file editor. Describe your task in English and the Assistant generates SQL queries, explains complex code and automatically fixes errors. It leverages Unity Catalog metadata to understand your data to provide personalized responses.

Run Code

In any editor, you can write and execute commands and see results alongside your code. Notebooks and IDEs support multiple languages. SQL editor executes only SQL queries.

Visualizations

Databricks comes with built-in support for charts and visualizations in Notebooks and the SQL editor. It supports open source libraries like matplotlib and ggplot. You can display the visualizations right in the editor.

Generative AI

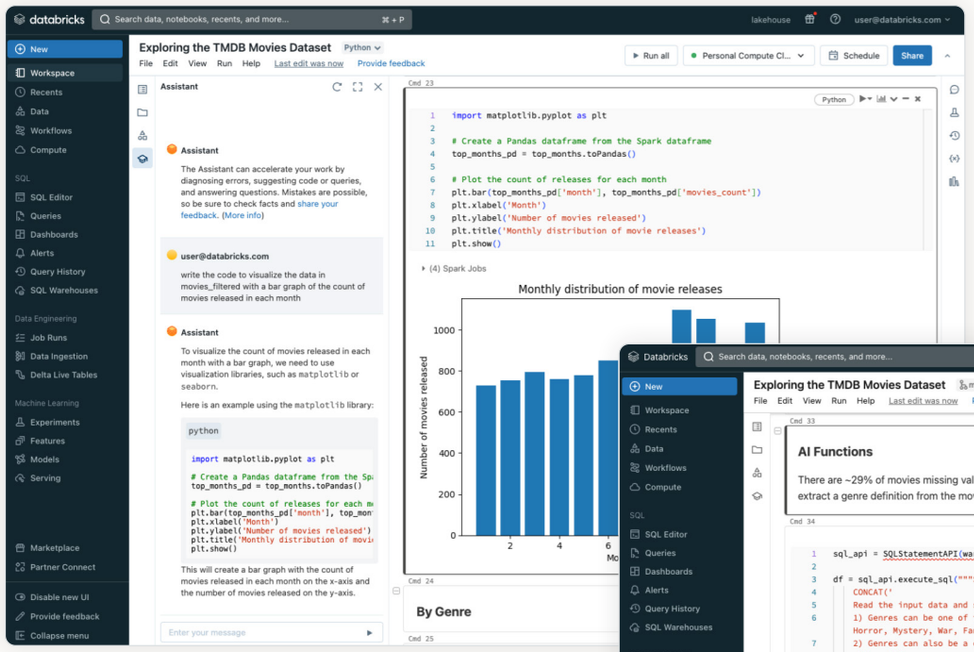
You can call large language models (LLMs) with AI Functions to help analyze and understand your data. LLMs excel at interpreting written language and can be used to assign scores or categories to rows to simplify analysis.

Transforming Your Data

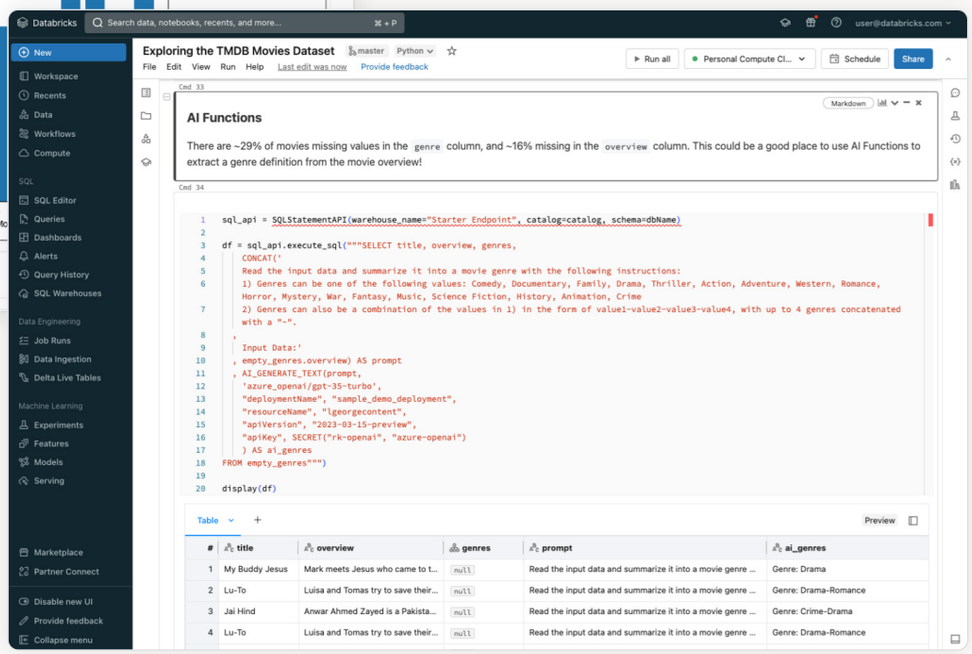
To keep your analysis up to date, you can create data pipelines to maintain data quality and freshness to match your use case.

Delta Live Tables

Once you have manipulated the data to what you want, turn that logic into a data pipeline with Delta Live Tables (DLT). You specify the data source, transformation logic and destination state of the data, and Delta Live Tables manages task orchestration, cluster management, monitoring, data quality and error handling. You can run DLT for batch or streaming data.



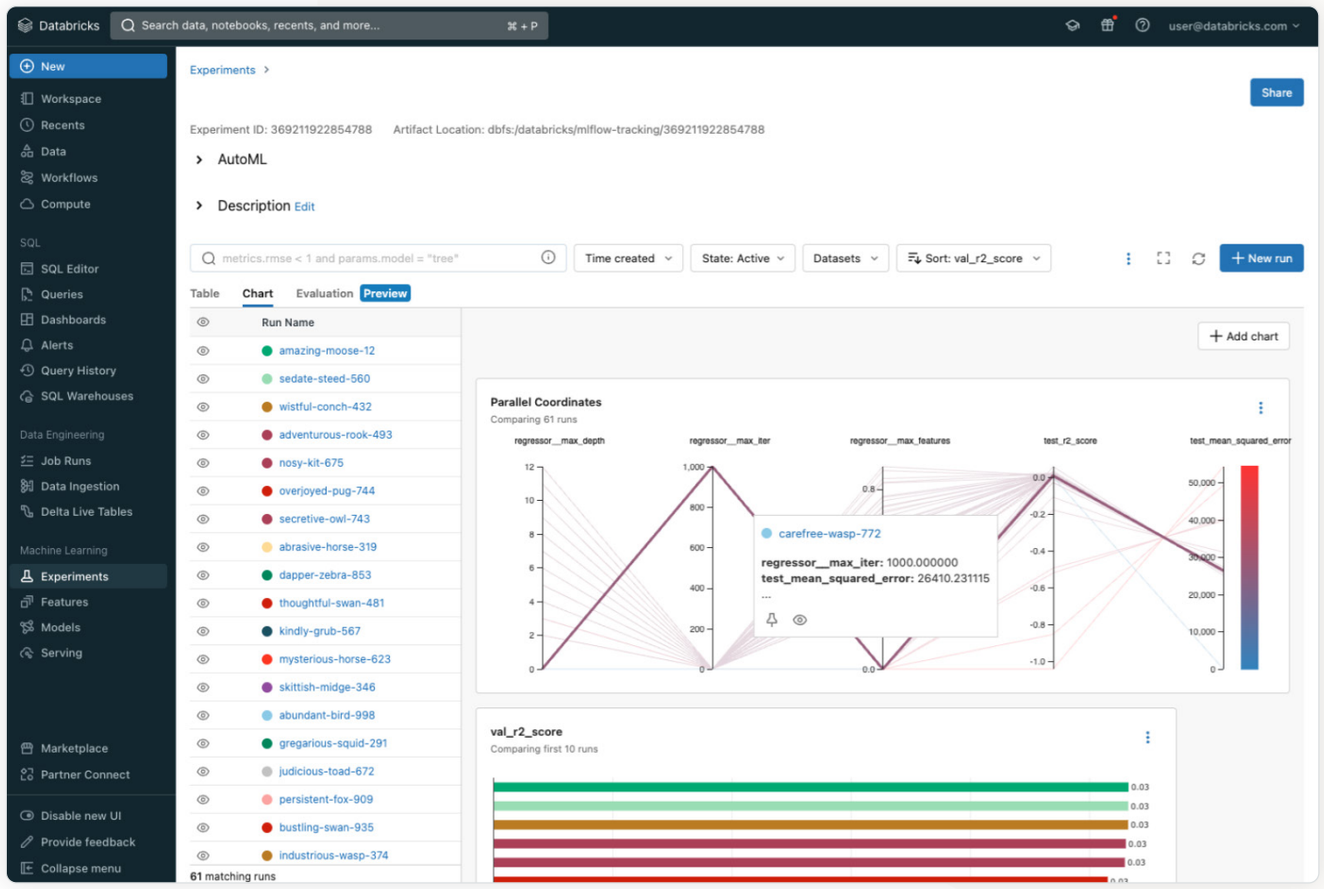
> Databricks Assistant generates code to graph the number of movies released in each month.



> With the movie data, a third of movies are missing "genre." AI Functions can categorize these movies into preset genres based on the movie overview.

Training Machine Learning (ML) Models

Depending on your use case, you may use machine learning. Databricks makes it simple to train different models on your data. The platform provides an integrated environment for the complete ML lifecycle from experimentation to production, including for generative AI and large language models.



Here we train a machine learning model to predict the Popularity Score for a movie, based on everything else we know about it.

AutoML

Quickly generate baseline models and notebooks to tackle a variety of machine learning problems, including classification, regression and forecasting. AutoML uses multiple algorithms from a variety of machine learning libraries for each problem type, and lets you pick the best for your problem.

Experiments

Each experiment tracks the data used, model parameters and model performance so you can evaluate and improve the model. Models are managed with Unity Catalog for fast and secure sharing.

Feature Engineering

Typically, raw data needs to be processed into useful features to train ML models. Developing features for ML training is complex and time-consuming. Often, different teams will have similar feature needs but are not aware of work that other teams have done. Any table in Unity Catalog can be used as a feature table. Tables with features for ML have all Unity Catalog capabilities, such as security, lineage, tagging and cross-workspace access.

Model Serving

Model Serving is the simplest way to deploy ML models. It integrates with your lakehouse data and offers automatic lineage, governance and monitoring across the model lifecycle. Model Serving automates infrastructure configuration and maintenance to reduce overhead. It also scales from zero all the way up to your most critical needs in real time, and down as your needs change, so you only pay for the compute you use.

Overcoming Errors

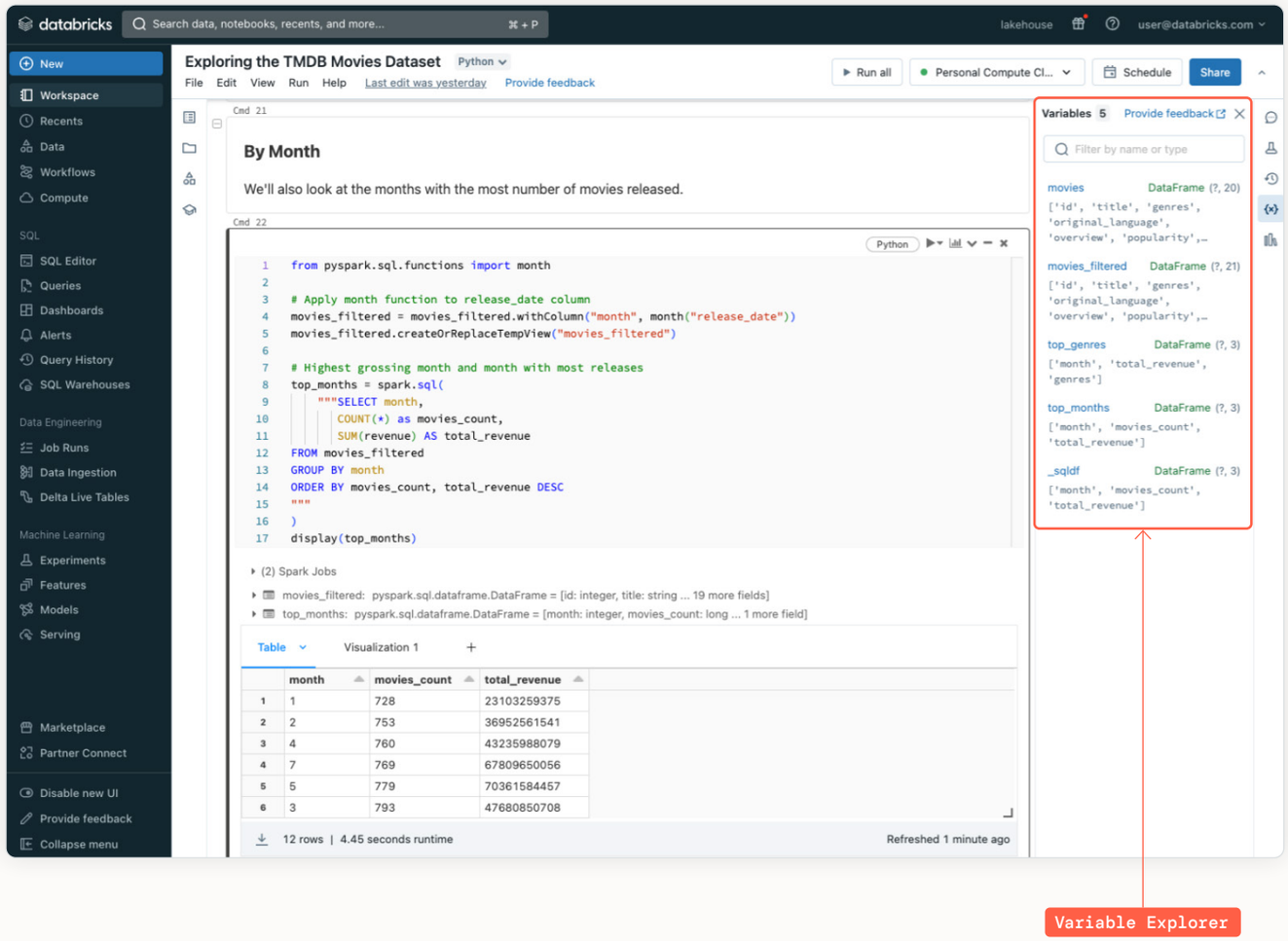
Not everything runs correctly the first time. Databricks comes with built-in tools to quickly identify issues and fix them.

Variable Explorer

Display all variables available in a notebook session in one place. The name, type and value are surfaced for all simple variable types. Variable Explorer also surfaces additional metadata for Spark and pandas DataFrames. The shape and column names are available at a glance, and full view of the schema is available on hover.

Python debugger

You can do step-through debugging of Python code with support for pdb in Databricks Notebooks. You set breakpoints with breakpoint() or pdb.set_trace(). When you run the cell, the execution will pause at the breakpoint and the Variable Explorer will automatically update with the state of the notebook.



Applying Software Development Best Practices

When working with large code bases and large teams, it is helpful to follow common practices to make developing code more efficient and reliable. The Databricks Data Intelligence Platform offers tools for creating code while applying these principles. See other Notebook best practices here.

Version control with Databricks Repos

Databricks Repos is a visual Git client and API in Databricks. It supports common Git operations such as cloning a repository, committing and pushing, pulling, branch management and visual comparison of diffs. Within Repos, you can develop code in Notebooks or other files and use Git for version control, collaboration and deployment.

Modular code with workspace files

A best practice for code development is to modularize code so it can be easily reused. You can create custom Python files within your workspace and make the code in those files available in a notebook with an import statement. You can create and manage source code files with Databricks Repos.

Unit testing with functions

You can set up unit tests outside your notebook with functions and call these as part of a standard testing procedure. IDEs also have native integrations for writing and running unit tests.

Going to Production

In production environments that are always on, you may want to use CI/CD to make changes and push them to production in a way that ensures that everything will work as expected.

Databricks Asset Bundles (DABs)

DABs standardize and unify the deployment strategy for all data products developed on the Databricks Data Intelligence Platform. DABs are a collection of Databricks artifacts such as Notebooks, ML models, DLT pipelines, clusters and other assets that are co-versioned in the same repository using a configuration (yaml) file. DABs simplify production changes by ensuring all pieces work together as tested and are modified together with clear versioning.

Terraform

HashiCorp Terraform is a popular open source tool for creating safe and predictable cloud infrastructure across several cloud providers. The Databricks Terraform provider supports all Databricks REST APIs, and you can use it to provision Databricks workspaces, to deploy and manage clusters and jobs, and to configure data access.

Sharing Results

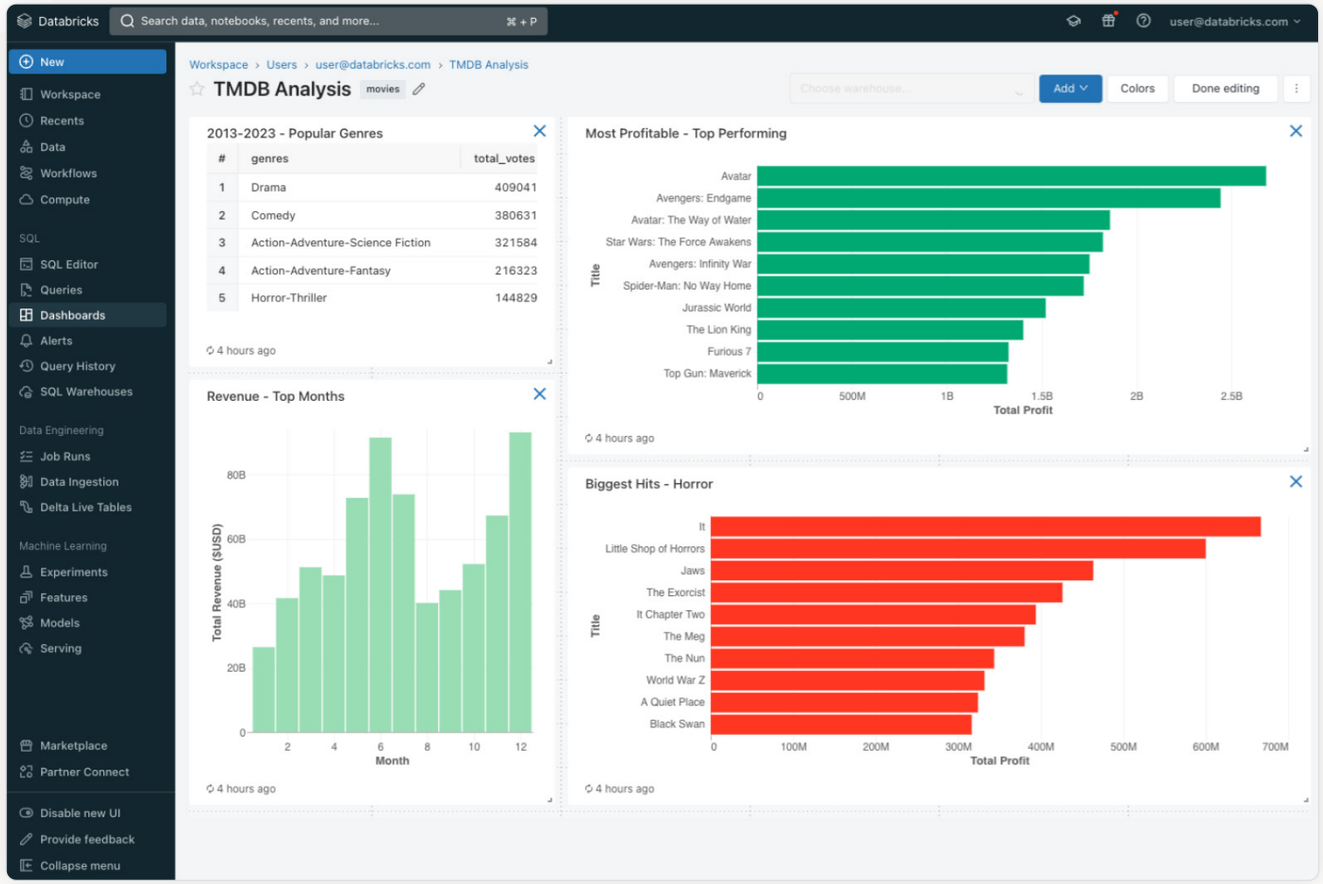
Once you have a data product that is built, tested and ready for viewing, it's time to share it with others. Databricks makes sharing assets simple.

Notebooks, queries, experiments and other assets

The simplest way is to share your code and results, and give others permission to view. You can share these assets or many others through the UI. Unity Catalog makes setting and governing permissions across assets simple.

Dashboards

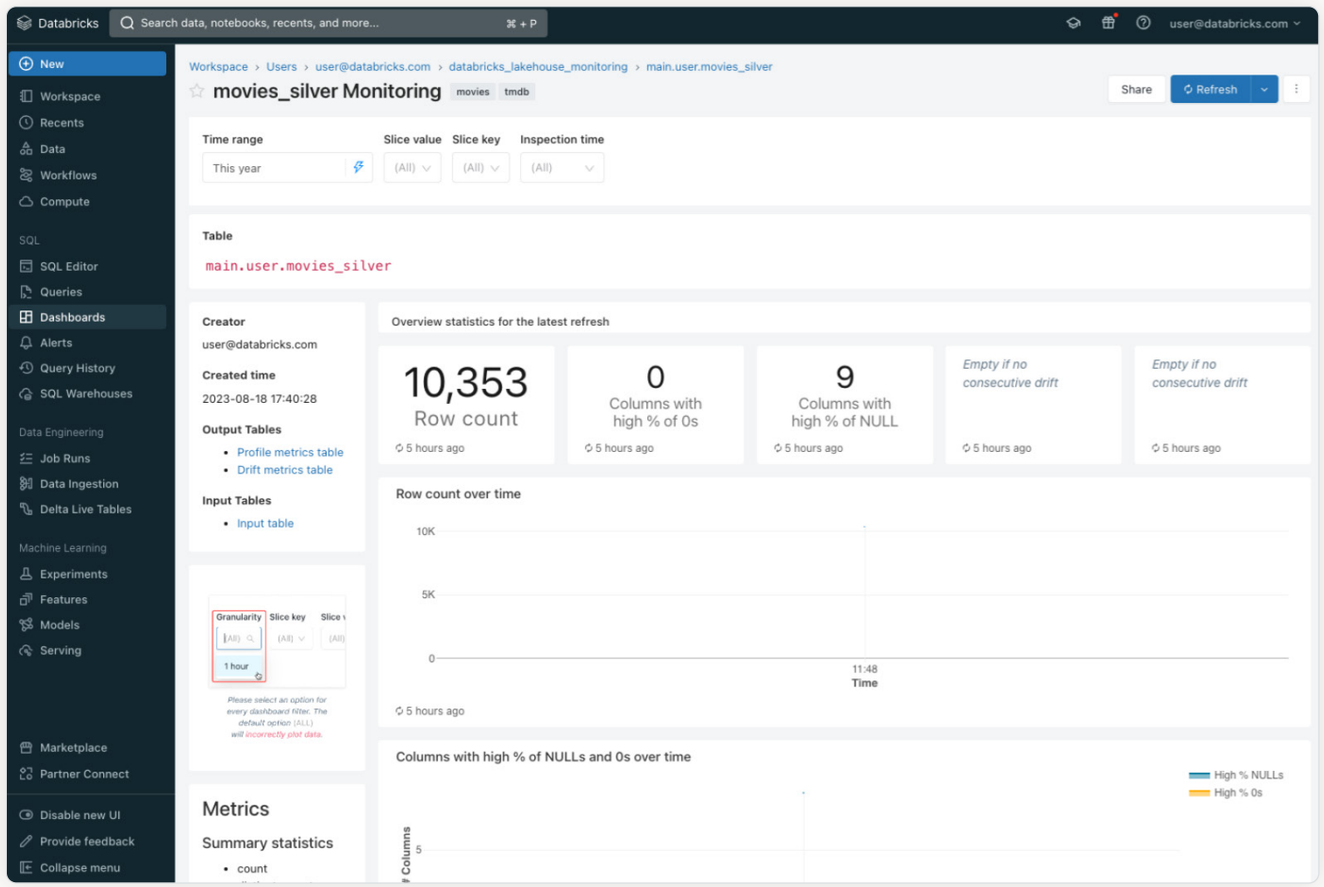
Creating tangible impact requires more than just finding the right answers — it also requires communicating the answers to relevant decision-makers. With Databricks, you can add your report or visualization to a dashboard and share it with others. Drop-down menus enable other users to filter for only the data they want to see.



See several analyses at a glance with dashboards. Shown here are different breakdowns of the movie dataset.

Automating and Monitoring

Automations for manual tasks help simplify the development experience. Numerous tools are available to reduce the need for intervention, allowing users to concentrate on their core tasks.



> You can start monitoring for any changes or new releases.

Workflows

Databricks Workflows enable you to orchestrate tasks on the Databricks Data Intelligence Platform. Workflows create recurring jobs to do batch ingestion at preset times or implement real-time data pipelines. Workflows can automate any lakehouse capability, such as Delta Live Table pipelines, Notebooks and SQL queries.

Software Development Kits (SDKs)

You can use SDKs to programmatically interact with Databricks workspaces. SDKs are available for Python, Go and Java and cover all public Databricks REST API operations. The SDKs include an internal HTTP client that handles different levels of failures by performing intelligent retries.

Lakehouse Monitoring

You can monitor entire data pipelines — from data and features to ML models — without additional tools or complexity. Powered by Unity Catalog, it lets you ensure your data and AI assets are high-quality, accurate and reliable through statistical analysis and insight into their lineage. The unified approach makes it simple to diagnose errors and find solutions.

Learn More

The world of data management and machine learning is always changing, and Databricks is changing with it. We're always working on new features and capabilities to make your life easier and help you on your journey.

Visit us at Databricks Learning to find out more about training and certification, documentation and upcoming events.

You can learn more about each of these features in the Databricks documentation.

About Databricks

Databricks is the data and AI company. More than 10,000 organizations worldwide — including Comcast, Condé Nast, Grammarly and over 50% of the Fortune 500 — rely on the Databricks Data Intelligence Platform to unify and democratize data, analytics and AI. Databricks is headquartered in San Francisco, with offices around the globe, and was founded by the original creators of Lakehouse, Apache Spark™, Delta Lake and MLflow. To learn more, follow Databricks on Twitter, LinkedIn and Facebook.

Sign up for a free trial

databricks