aws

User Guide

Amazon SageMaker

Amazon SageMaker: User Guide

Table of Contents

What is Amazon SageMaker?

Bringing together widely adopted artificial intelligence (AI) and analytics capabilities, the next generation of Amazon SageMaker delivers an integrated experience for analytics and AI with unified access to all your data. Collaborate and build in Amazon SageMaker Unified Studio using familiar AWS tools for SQL analytics, data processing, model development, and generative AI, accelerated by Amazon Q Developer. Access all your data whether it's stored in data lakes, data warehouses, or third-party or federated data sources, with governance built in to meet enterprise security needs.

The next generation of Amazon SageMaker overview

Guide to SageMaker

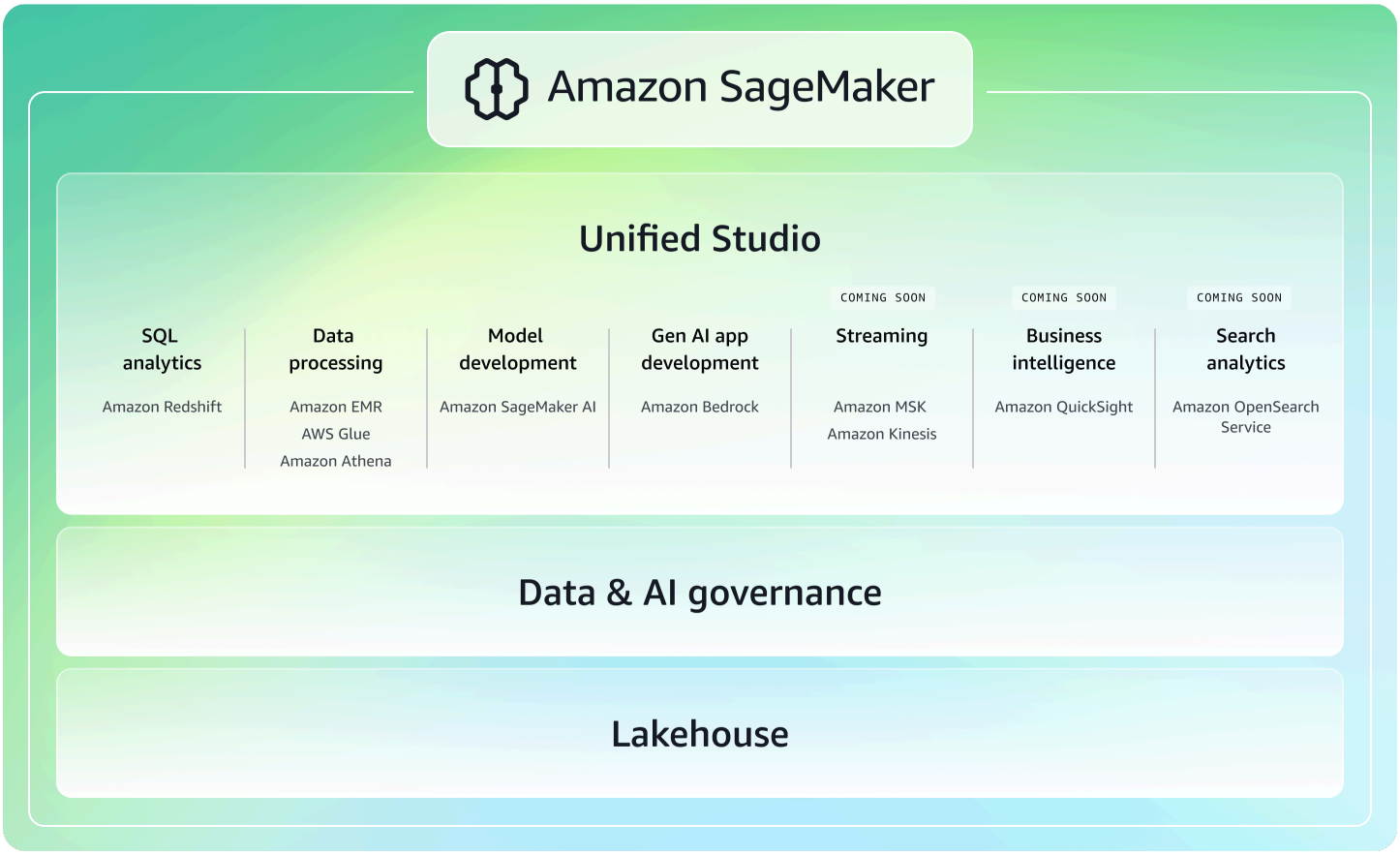
The next generation of Amazon SageMaker was announced at re:Invent 2024 serves as the center for all data, analytics, and AI. Analytics and AI workflows are converging, with organizations now using the same data sources for traditional analytics, machine learning, and generative AI. In response, AWS has created the next generation of SageMaker to serve as a unified platform for these workflows. The next generation of SageMaker brings together the purpose-built components needed for data exploration, preparation and integration, big data processing, SQL analytics, machine learning (ML) model development and training, and generative AI application development.

> **Note**
>
> The original Amazon SageMaker has been renamed SageMaker AI. It is available in the next generation Amazon SageMaker for those who wish to use it alongside additional capabilities, or as a standalone service for those who wish to focus specifically on building, training, and deploying AI and ML models at scale.

The next generation of Amazon SageMaker consists of two primary components:

1. Amazon SageMaker Unified Studio, which provides an integrated experience to use all your data and tools for analytics and AI
2. Data and AI governance, which applies enterprise-level security and data management with built-in governance throughout the entire data and AI lifecycle

Additionally, SageMaker is built upon an open lakehouse architecture that unifies access to all your data across Amazon Simple Storage Service (Amazon S3) data lakes, Amazon Redshift data warehouses, and other external sources



Unified Studio

Amazon SageMaker Unified Studio is a single data and AI development environment that brings together functionality and tools that AWS offers in Amazon EMR, AWS Glue, Amazon Athena, Amazon Redshift, Amazon MWAA, Amazon Bedrock, and Amazon SageMaker AI. From within the unified studio, you can discover, access, and query data and AI assets, then collaborate to build and share analytics and AI artifacts, including data, models, and generative AI applications.

Data & AI governance

The next generation of Amazon SageMaker simplifies the discovery, governance, and collaboration for data and AI. With Amazon SageMaker Catalog, users can securely discover and access approved data and assets using semantic search with generative AI–created metadata, or you could just ask Amazon Q Developer with natural language to find your data. Seamlessly share and collaborate on data and AI assets through publishing and subscribing workflows. With SageMaker, you can apply

Amazon Bedrock Guardrails to protect and filter your model outputs, helping ensure responsible gen AI application development. Build trust throughout your organization with data quality monitoring, sensitive data detection, and data and machine learning (ML) lineage.

Lakehouse architecture

The next generation of Amazon SageMaker is built on an open lakehouse architecture, fully compatible with Apache Iceberg. Unify all your data across Amazon S3 data lakes and Amazon Redshift data warehouses to build analytics and AI/ML applications on a single copy of data. The lakehouse gives you the flexibility to access and query your data with Apache Iceberg–compatible tools and engines. You can also connect to federated data sources such as Amazon DynamoDB, Google BigQuery, and Snowflake and query your data in-place. With zero-ETL integrations, you can bring data from operational databases and 3rd party applications into your lakehouse in near real-time. Integrated fine-grained access controls help you secure your data to ensure only the right people have access to the right data.

Capabilities of Amazon SageMaker Unified Studio

The next generation of Amazon SageMaker and its unified studio provide an integrated experience to use all your data and tools for analytics and AI. Discover your data and put it to work using familiar AWS tools for model development, generative AI, data processing, and SQL analytics. Work across compute resources using unified notebooks, discover and query diverse data sources with a built-in SQL editor, train and deploy AI models at scale, and rapidly build custom generative AI applications. Create and securely share analytics and AI artifacts such as data, models, and generative AI applications to bring data products to market faster.

Some common capabilities of Amazon SageMaker Unified Studio include the following:

SQL analytics

Leverage SageMaker's SQL analytic capabilities across all of your unified data through Amazon SageMaker's lakehouse architecture. Users have the flexibility to use Athena or Redshift query engines to support their analytical workloads. Query your data in open formats stored on Amazon S3 with high performance through Amazon Athena, eliminating the need to move or duplicate data between your data lakes and data warehouse. Include your Redshift data as part of the lakehouse architecture, leveraging the Redshift query engine for SQL workloads on structured data.

Data processing

Prepare, orchestrate, and process your data with capabilities in SageMaker, allowing you to run Apache Spark, Trino, and other open-source analytics frameworks in a unified data and AI development environment. Process your data, wherever it lives, with connectivity to hundreds of data sources with Amazon Athena, Amazon EMR, and AWS Glue.

Data integration

You can use data integration capabilities in Amazon SageMaker to connect to and act on all your data. With AWS data integration capabilities, you can bring together data from multiple sources, operationalize it, and manage to deliver high quality data to your lakehouse architecture, across your data lakes and data warehouses.

> **Note**
>
> *What data sources am I able to integrate with Amazon SageMaker?*
> You are able to unify all your data across Amazon Redshift data warehouses and Amazon S3 data lakes, including S3 Tables, with SageMaker's lakehouse architecture. Bring your operational databases and 3rd party application data like Salesforce and SAP to the lakehouse in near real time through zero-ETL integrations. You can use hundreds of connectors to integrate data from various sources. Additionally, you can access and query data in-place with federated query capabilities across third-party data sources.

Machine learning and model development

Amazon SageMaker AI is a fully managed service that brings together a broad set of tools to enable high-performance, low-cost machine learning (ML). Most capabilities of SageMaker AI are available as part of Amazon SageMaker Unified Studio, in addition to being available in Amazon SageMaker Studio. With SageMaker AI, you can build, train and deploy ML models at scale using tools like notebooks, debuggers, profilers, pipelines, MLOps, and more—all in one integrated development environment (IDE).

> **Note**
>
> *When should I use SageMaker Unified Studio instead of SageMaker AI studio?*

Currently, SageMaker Unified Studio should be used when you are looking to unify and share your data as a single integrated experience across analytics, ML, and gen AI workloads. You are able to eliminate data silos with an open lakehouse architecture to unify access to data lakes, data warehouses, third-party or federated data sources, and meet all enterprise security needs with built-in data and AI governance.
If you want to solely focus on the purpose-built tools to perform all machine learning (ML) development steps, from preparing data to building, training, deploying, and managing your ML and gen AI models, SageMaker Studio remains a great choice. Additionally, use SageMaker Studio when there are requirements for RStudio, Canvas, real-time collaboration via shared spaces, and Feature Store.

Generative AI application development

Access Amazon Bedrock's capabilities through SageMaker Unified Studio to quickly build and customize your generative AI applications. This intuitive interface lets you work with high-performing foundation models (FMs) from leading companies like Anthropic, Mistral, Meta, and Amazon, and use advanced features like Amazon Bedrock Knowledge Bases, Amazon Bedrock Guardrails, Amazon Bedrock Agents, and Amazon Bedrock Flows. You can develop generative AI applications faster within SageMaker Unified Studio's secure environment, ensuring alignment with your requirements and responsible AI guidelines.

Note

When should I use Bedrock in SageMaker Unified Studio versus the standalone Amazon Bedrock service?
Amazon Bedrock's capabilities in Amazon SageMaker Unified Studio are ideal for enterprise teams who need a governed low-code/no-code environment for collaboratively building and deploying generative AI applications, alongside unified analytics and machine learning capabilities.
Customers can use the standalone Bedrock service from the AWS Management Console or Bedrock API when they want to leverage the full feature set of Bedrock including the latest agents, flow and guardrail enhancements, and the Bedrock SDK.

Get started with Amazon SageMaker

You can view demos of Amazon SageMaker and get started by setting up a domain and project.

View demos of Amazon SageMaker

To see Amazon SageMaker before using it yourself, you can review the following clickthrough demos:

- For an end-to-end demo, see the Amazon SageMaker detailed clickthrough experience. This demo includes SageMaker Lakehouse, Amazon SageMaker Catalog, and more in Amazon SageMaker Unified Studio.
- For a demo of SageMaker Lakehouse, see Amazon SageMaker: Access data in your lakehouse. This demo includes SageMaker Lakehouse in Amazon SageMaker Unified Studio, including adding a data source and querying data.
- For a demo of the Amazon SageMaker Catalog, see Amazon SageMaker: Catalog. This demo includes Amazon SageMaker Catalog in Amazon SageMaker Unified Studio, including browsing assets and subscribing to an asset.
- For a demo of generative AI, see Amazon SageMaker: Generative AI playground and Gen AI app development.

Get started with setting up Amazon SageMaker

To get started using Amazon SageMaker, go to *Setting up Amazon SageMaker* in this guide to set up a domain and create a project. This domain setup and project creation is a prerequisite for all other tasks in Amazon SageMaker.

Prerequisites for Amazon SageMaker

Complete the following prerequisite tasks before you can set up Amazon SageMaker and proceed with the use cases in this guide.

Topics

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call or text message and entering a verification code on the phone keypad.

   When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform tasks that require root user access.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing **My Account**.

Create a user with administrative access

After you sign up for an AWS account, secure your AWS account root user, enable AWS IAM Identity Center, and create an administrative user so that you don't use the root user for everyday tasks.

Secure your AWS account root user

1. Sign in to the AWS Management Console as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

   For help signing in by using root user, see Signing in as the root user in the *AWS Sign-In User Guide*.

2. Turn on multi-factor authentication (MFA) for your root user.

   For instructions, see Enable a virtual MFA device for your AWS account root user (console) in the *IAM User Guide*.

Create a user with administrative access

1. Enable IAM Identity Center.

   For instructions, see Enabling AWS IAM Identity Center in the *AWS IAM Identity Center User Guide*.

2. In IAM Identity Center, grant administrative access to a user.

   For a tutorial about using the IAM Identity Center directory as your identity source, see Configure user access with the default IAM Identity Center directory in the *AWS IAM Identity Center User Guide*.

Sign in as the user with administrative access

- To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.

  For help signing in using an IAM Identity Center user, see Signing in to the AWS access portal in the *AWS Sign-In User Guide*.

Assign access to additional users

1. In IAM Identity Center, create a permission set that follows the best practice of applying least-privilege permissions.

   For instructions, see Create a permission set in the *AWS IAM Identity Center User Guide*.

2. Assign users to a group, and then assign single sign-on access to the group.

   For instructions, see Add groups in the *AWS IAM Identity Center User Guide*.

Setting up Amazon SageMaker

Complete the following tasks to set up Amazon SageMaker.

Topics

Step 1 - Create an Amazon SageMaker unified domain

Complete the following procedure to create an Amazon SageMaker unified domain with the Quick setup option.

> **Important**
>
> Note that there is an additional charge for any VPC or resources that AWS sets up if you chose the Quick setup option for domain creation.

1. Navigate to the Amazon SageMaker management console at https://console.aws.amazon.com/datazone and use the region selector in the top navigation bar to choose the appropriate AWS Region.
2. Choose **Create a Unified Studio domain** and then choose **Quick setup**.

   With this option, you're choosing to create an Amazon SageMaker unified domain and you're letting Amazon SageMaker configure your domain with the following default capabilities that you can customize later:

   - Data analytics, machine learning, SQL, and generative AI
   - Data and AI governance
   - Generative AI app development using Amazon Bedrock serverless models
   - Amazon Q - Free tier
   - Authentication via AWS IAM or AWS IAM Identity Center

3. If you see the following note **No VPC has been specifically set up for use with Amazon SageMaker Unified Studio**, you can use the **Choose VPC** or **Create VPC** buttons to **Create a new VPC (recommended)** or choose an existing properly-configured VPC.

   If you plan to choose your own VPC, Amazon SageMaker Unified Studio enables you to choose VPCs within the same account as well as shared VPCs from other member accounts of the AWS organization. For more information, see Share your VPC subnets with other accounts.

   > **Note**
   >
   > If you choose to create a new VPC, note that the VPC template with which it is created is not intended for production use. You can use this template as a start and modify it for your organization's purposes.

4. If you see the following note **No models accessible**, you can use the **Grant model access** button to grant access to Amazon Bedrock serverless models for use in Amazon SageMaker.
5. Expand the **Quick setup settings** section and review the specified configurations for the domain. Leave these defaults and then choose **Continue** to proceed with creating your domain.

   > **Note**
   >
   > For more information, see IAM roles for Amazon SageMaker Unified Studio.

6. On the **Create IAM Identity Center user** page, create a new or select an existing SSO user that you want to enable to log in to Amazon SageMaker Unified Studio. This is done because IAM roles that are used to create Amazon SageMaker unified domains cannot log in to Amazon SageMaker Unified Studio. The SSO user specified here is used as the administrator in Amazon SageMaker Unified Studio.
7. Choose **Create domain**.

After some time, an email will be sent to the address you provided as part of the IAM Identity Center user setup. The email will prompt you to set a password that you can use to access the domain.

Step 2 - Create a new project

In Amazon SageMaker, projects enable a group of users to collaborate on various business use cases. Within projects, you can manage data assets in the Amazon SageMaker catalog, perform data analysis, organize workflows, develop machine learning models, build generative AI apps, and more.

Navigate to Amazon SageMaker Unified Studio

To begin creating a project, navigate to Amazon SageMaker Unified Studio. You can do this by using the link in your email that you used to set an IAM Identity Center password, or by selecting the domain in the Amazon SageMaker management console and choosing **Open unified studio**.

Sign in using your SSO credentials that you configured using the email from IAM Identity Center.

If your IAM Identity Center is configured to require multi-factor authentication (MFA), set up and use an MFA device. Follow the instructions on the screen to register or use an MFA device as needed, or contact your admin for support. For more information about configuring MFA device enforcement, see Configure MFA device enforcement in the IAM Identity Center User Guide.

Project name and description

After navigating to Amazon SageMaker Unified Studio, choose **Create project**.

The project name and description includes the following fields:

- Project name - the name of your project. Enter a name here. The name of the project can not be edited after the project is created.
- Description - an optional description of your project. You can edit this later.
- Project profile - project profiles define which resources and tools should be provisioned in the project. These include tools and compute resources for SQL, data science, data engineering, and machine learning development. Project profiles can include resources and tools from Amazon Redshift, Amazon SageMaker AI, and other AWS services. To complete the use cases in this getting started guide, choose the **All capabilities** project profile.

Choose **Continue** to review parameters.

Review parameters

On the next page of project creation, you can review and optionally edit the names and values for different resources that are created when the project is created. You can leave all the defaults and then choose **Continue**.

Review

Use the last page of project creation to review the configurations you have selected. When everything is configured as desired on the project creation review page, choose **Create project**.

You are then redirected to the project home page. The project will start building and a progress bar will appear with the status.

Get started with uploading and querying data in Amazon SageMaker Unified Studio

You can use the query editor to perform analysis using SQL. The query editor tool provides a place to write and run queries, view results, and share your work with your team.

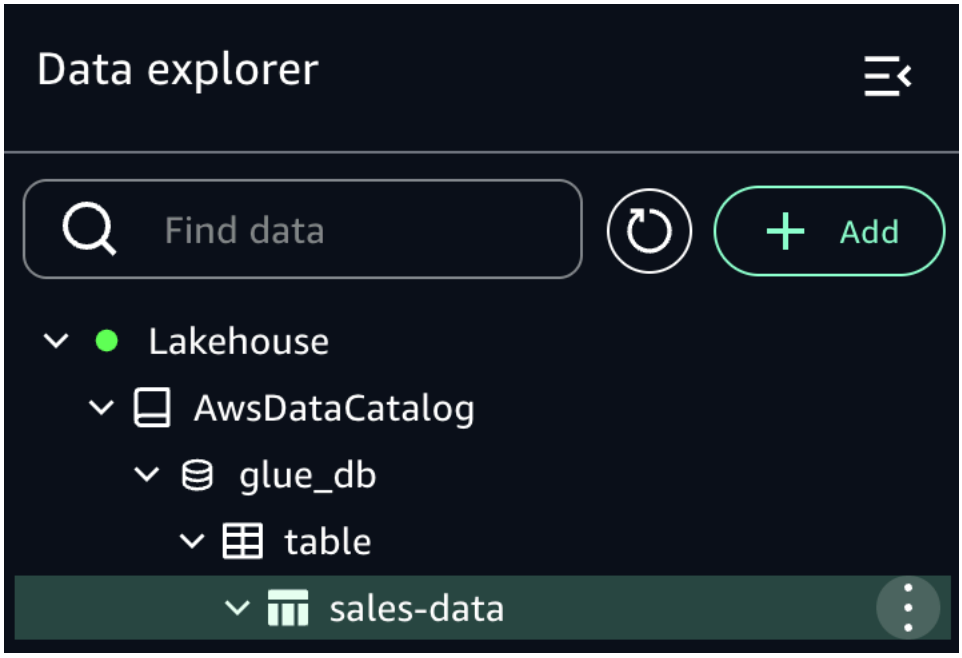
Prerequisites

Before you get started with the query editor, access Amazon SageMaker and create a project with the **SQL analytics** or **All Capabilities** project profile. For more information, see *Setting up Amazon SageMaker*.

Download the file sales-data.zip.

Query sample data using Amazon Athena in Amazon SageMaker

1. Navigate to Amazon SageMaker Unified Studio using the URL from the Amazon SageMaker management console and log in using your SSO or AWS credentials.
2. Use the top center menu of the Amazon SageMaker home page to navigate to the project you want to use to query data.
3. Expand the **Build** menu in the top navigation bar, then choose **Query editor**.
4. In the left data explorer navigation, choose the three-dot action menu next to a database and choose **Create table**.
5. Upload the sales-data.csv file from the prerequisites section.
6. Choose **Next**.
7. Choose **Create table**.
8. Refresh the Data explorer navigation pane and navigate to the sales-data table in the explorer.
9. Choose the three-dot action menu next to the table, then choose **Preview data**. A SQL command to select the first 10 rows from the table runs, and the results are then displayed in the query editor window.


Data explorer
Find data
Add
Lakehouse
AwsDataCatalog
glue_db
table
sales-data

Get started with importing and querying data sets for AWS Glue Data Catalog and Amazon S3 in Amazon SageMaker Unified Studio

In this Getting Started tutorial for the next generation of Amazon SageMaker, you will use Amazon SageMaker Unified Studio, Amazon SageMaker Catalog, and Amazon SageMaker Lakehouse to import and query data sets. You will learn how to access and leverage your existing AWS Glue Data Catalog resources within Amazon SageMaker Unified Studio, allowing you to query and analyze your data without moving or duplicating it.

You will need to have administrator access to a domain or create a domain.

A summary of the tasks in this getting started are as follows.

- Prerequisites and permissions setup
- Setting up AWS Glue Data Catalog resources
- Configuring S3 access and data import
- Multiple query options (Spark and Athena) and data visualization/analysis capabilities:
  - Use Spark in Jupyter notebooks
  - Use Athena in the query editor
  - Create and modify tables using SQL
  - Visualize results using charts

This getting started uses a .parquet file as sample S3 Raw file data to import that you can retrieve from the public bucket. There are other formats of data you can import into Lake Formation tables for AWS Glue Data Catalog, such as RDS tables, DynamoDB tables, or RedShift tables.

Topics

Prerequisites

The following prerequisities are required for this getting started procedure.

- Create a project with an **All capabilities** project profile. This project profile sets up your project with access to S3 and Athena resources. There is more information about how to create a new project in the topic Setting up Amazon SageMaker AI.
- A project role is created automatically when the project is created in SageMaker Unified Studio. You will make a note of the project role as detailed in the prerequisities below.
- You can either use an existing AWS Glue database or create a new one. The Glue database must be Lake Formation managed.
- You can either use an existing AWS Glue table or create a new one. The Glue table must be Lake Formation managed.
- You also set up the Data lake administrator and revoke specified permissions.

Subsequent sections go into more detail regarding configuring each of these prerequisites.

To set up the Lake Formation Data Lake administrator

You must set up a user or role as the Lake Formation Data Lake administrator for your catalog data. This administrator grants access to data-lake resources.

1. In Create a data lake administrator in the *AWS Lake Formation Developer Guide*, follow the instructions to add the `AWSLakeFormationDataAdmin` managed policy to the user in IAM.
2. After you add the IAM permission, follow the steps in Create a data lake administrator to add the inline policy granting permission to create the service-linked role.

To add the Lake Formation Data Lake administrator in the Lake Formation console

After updating the policies in the previous step for the user or role you want to make the Data lake administrator, use the Lake Formation console to add that user or role to the list under Data lake administrators. Use the following steps to add the Data lake administrator on the console.

1. Open the AWS Lake Formation console.
2. Under **Administration**, choose **Administrative roles and tasks**.
3. Under **Data lake administrators**, choose **Add**.

4. For **Access type**, choose **Data lake administrator**.
5. For **IAM users and roles**, choose the user or role that you want to make the Data lake administrator. Make sure it is the same user or role for which you updated the IAM permissions in the Prerequisites.
6. Choose **Confirm**.

Revoke the `IAMAllowedPrincipals` group permission

You must revoke the `IAMAllowedPrincipals` group permission on both database and table to enforce AWS Lake Formation permission for access. For more information, see Revoking permission using the AWS Lake Formation console in the *AWS Lake Formation Developer Guide*.

> **Note**
>
> For the purposes of this topic, revoke the group permission as provided. This makes it so Lake Formation is the central point for managing fine-grained access control to your data lake resources. You can also use hybrid permissions in Lake Formation. For more information about hybrid permissions, see Hybrid access mode in the *AWS Lake Formation Developer Guide*.

1. Open the AWS Lake Formation console.
2. Under **Permissions**, choose **Data permissions**.
3. Choose the selector next to the `IAMAllowedPrincipals` group designated for **Database**.
4. Choose **Revoke**.

Step 1: Connect to an AWS Glue Data Catalog

Complete the steps in this section to set up your resources and permissions for accessing AWS Glue Data Catalog and preparing to import data.

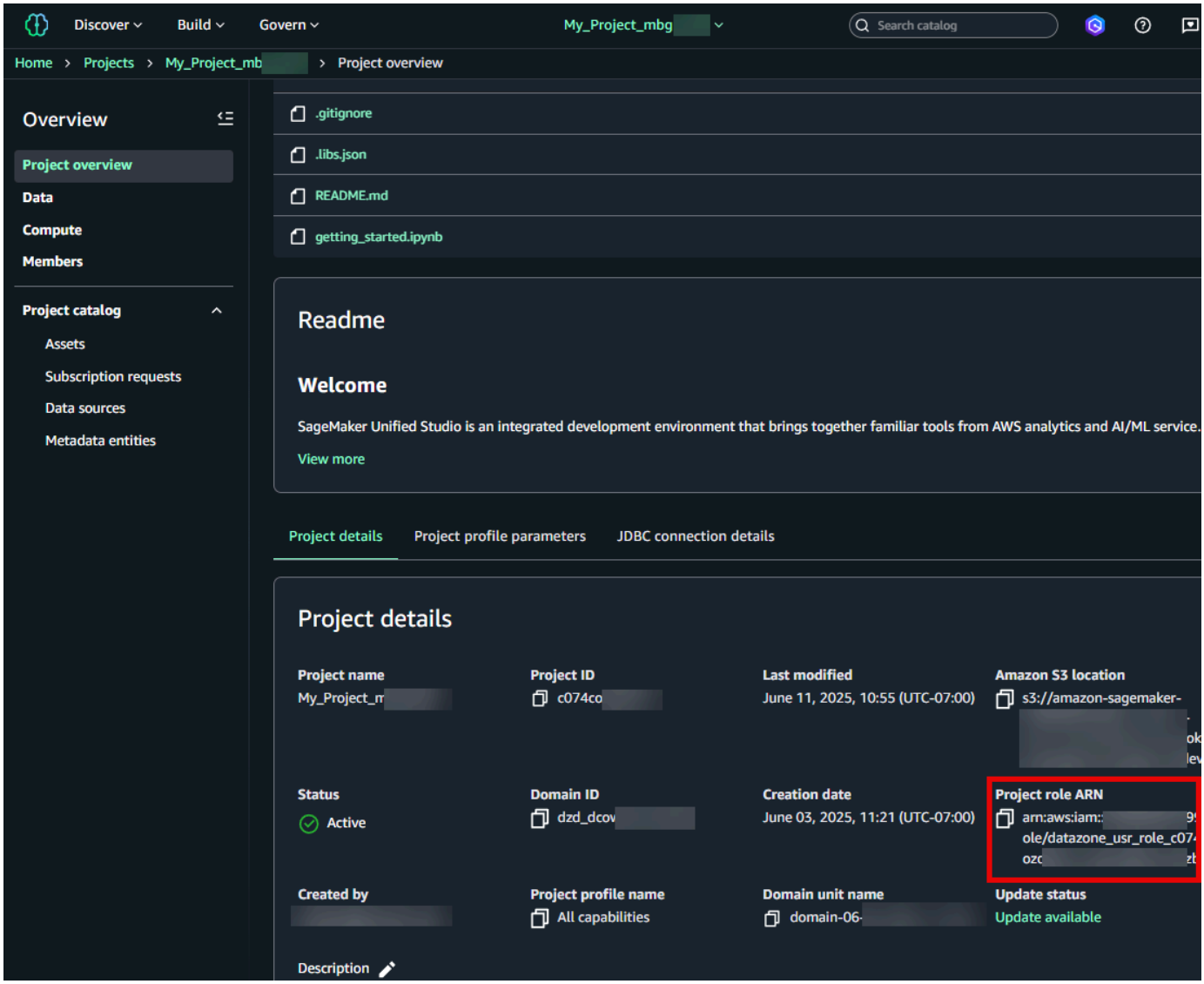
Make a note of your IAM project role

In the following sections of this topic, you will configure permissions using the project role in IAM that was created when you created your SageMaker Unified Studio project. The project role is an IAM role that is created and associated with a new project. This role grants the necessary

permissions for users working on the project to use AWS resources, such as Amazon S3, for instance. You will attach a resource-based bucket policy and configure permissions in the lakehouse. Use the following steps to make a note of the IAM project role for your SageMaker Unified Studio project. You will use the role in a procedure that follows when you configure and grant Lake Formation permissions.

1. Navigate to Amazon SageMaker Unified Studio using the URL from the Amazon SageMaker management console and log in using your SSO or AWS credentials.
2. Use the top center menu of the Amazon SageMaker home page to navigate to the project you want to use.
3. Under the **Overview**, choose **Project overview**.
4. Choose the **Project details** tab.
5. Choose the project role that is associated with your Amazon SageMaker Unified Studio project. This role was created in IAM upon project creation and was copied in the steps above. In **Project role ARN**, copy the project role ARN.

   The Project IAM role will have the following format: `arn:aws:iam::ACCOUNT_ID:role/datazone_usr_role_xxxxxxxxxxxxxx_yyyyyyyyyyyyyy`



Register the S3 location for AWS Glue Data Catalog tables in Amazon SageMaker Unified Studio

To access existing AWS Glue Data Catalog tables in Amazon SageMaker Unified Studio, complete the following steps to configure permissions.

To register the S3 location and configure access

1. Open the AWS Lake Formation console using the data lake administrator. Choose **Data lake locations** in the navigation pane, and then choose **Register location**.

2. Enter the S3 prefix for Amazon S3 path. For this topic, you must register the following S3 location in order to allow it to be queried: `s3://aws-bigdata-blog/generated_synthetic_reviews/data/product_category=Video_Games`.
3. For **IAM role**, choose your Lake Formation data access IAM role, which is not a service linked role.
4. Select **Lake Formation** for **Permission mode**, and then choose **Register location**.
5. For Database permissions, choose **Describe**, and then choose **Grant**.

Grant permission on the databases to the project role

You will grant database access to the IAM role that is associated with your Amazon SageMaker Unified Studio project. This role is called the project role, and it was created in IAM upon project creation. To access existing AWS Glue Data Catalog databases in Amazon SageMaker Unified Studio, complete the following steps to configure permissions.

1. On the Lake Formation console, under **Data Catalog** in the navigation pane, choose **Databases**.
2. Select the existing AWS Glue Data Catalog database.
3. From the **Actions** menu, choose **Grant** to grant permissions to the project role.
4. For **IAM users and roles**, choose the **project role**. This is the SageMaker Unified Studio project role that you noted previously in Make a note of your IAM project role.
5. Select **Named Data Catalog resources**, and for **Catalogs**, choose the default catalog or a catalog you want to use.
6. For **Databases**, choose the default database or a database you want to use.
7. For **Database permissions**, select **Describe** and choose **Grant**.

Granting these permissions provides the means to query the Lake Formation data in later steps.

Grant permission on the tables to the project role

You will grant table access to the IAM role that is associated with your Amazon SageMaker Unified Studio project. This role is called the project role, and it was created in IAM upon project creation. To grant permission on the tables to the project role, complete the following steps.

1. On the Lake Formation console, under **Data Catalog** in the navigation pane, choose **Databases**.
2. Select the existing Data Catalog database.
3. From the **Actions** menu, choose Grant to grant permissions to the project role.
4. For **IAM users and roles**, choose the project role. This is the SageMaker Unified Studio project role that you noted previously in Make a note of your IAM project role.
5. Select **Named Data Catalog resources**, and for **Catalogs**, choose the default catalog.
6. For **Databases**, choose your **Data Catalog** database.
7. For **Tables**, select the tables that you need to provide permission to the project role.
8. For **Table permissions**, select **Select** and **Describe**.
9. For **Grantable permissions**, choose **Select** and **Describe**.
10. Choose **Grant**.

> **Important**
>
> You should revoke any existing permissions of `IAMAllowedPrincipals` on the databases and tables within Lake Formation as detailed in the prerequisites.

Create a new Lakehouse catalog

In your project, create a new Lakehouse catalog. If you plan to use the default catalog, you can skip these steps.

To create a Lakehouse catalog

1. In the Amazon SageMaker Unified Studio, navigate to your project.
2. On the project page, under **Data**, choose **Lakehouse**.
3. Choose the **+** button.
4. In the **Add data** section, choose **Create Lakehouse catalog**.
5. Choose **Next**.
6. In the **Add catalog** section, enter a name for your catalog.
7. (Optional) Enter a description for the catalog.
8. Choose **Add catalog**.

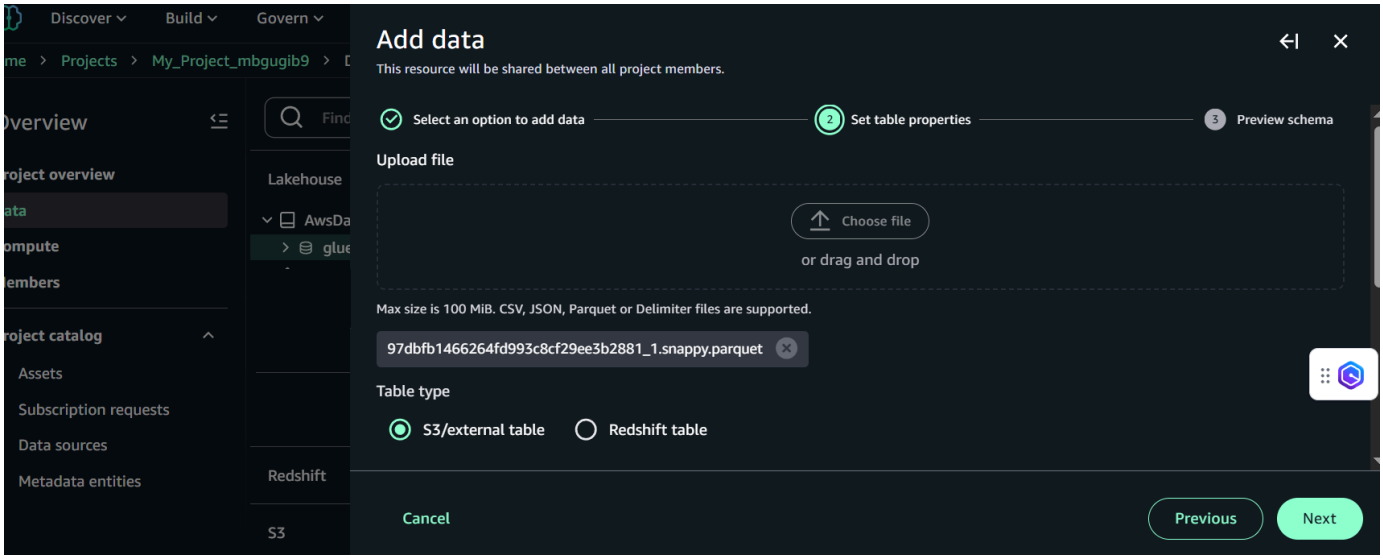
After completing these steps, your database will appear under the catalog that you've created.

Add data and create an AWS Glue table

In your project, create an AWS Glue table using sample data. To create a Glue table in Amazon SageMaker Unified Studio, complete the following steps.

To add data and create a Glue table

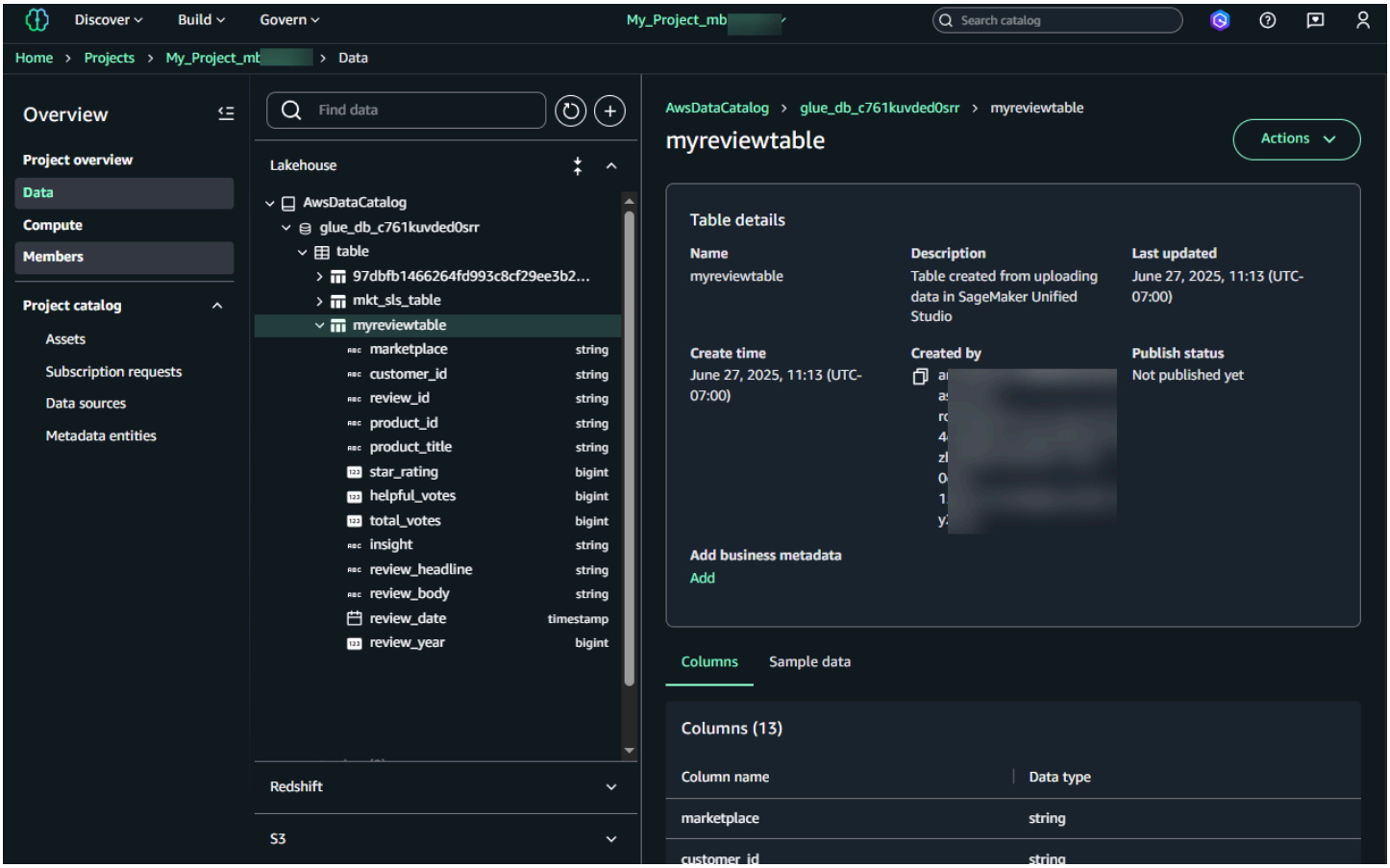
1. Access the public S3 bucket to download the sample data. Download the .parquet file named `97dbfb1466264fd993c8cf29ee3b2881_1.snappy.parquet` to your local drive.
2. In the Amazon SageMaker Unified Studio console, navigate to your project.
3. On the project page, under **Overview**, choose **Data**. Choose **Lakehouse**.
4. Next to your Glue database, choose the options menu (three dots), and choose **Create table**.
5. Next, upload the file in .CSV, JSON, Parquet, or Delimiter formats. For this example, upload the Parquet file you downloaded from the public sample bucket.
6. For **Table type**, **External/S3** is selected by default as the type of source.
7. Choose **Add data**. For **Catalog name**, choose the name from the drop-down menu.



8. For **Database**, choose the database that you created in the **Use or create a Glue database** section from the drop-down menu.
9. For **Table name**, enter a table name of your choice.
10. For **Data format**, choose the data format from the drop-down menu. The format updates automatically when you upload a file.
11. Choose **Next**. Allow a few minutes for the schema creation to display.

12. Choose **Create table**.

The table appears under your database, such as in this example showing the new table `myreviewtable` added.



Verify access to your AWS Glue table from the Amazon SageMaker Unified Studio query editor

To verify that you can access the existing AWS Glue table from the Amazon SageMaker Unified Studio query editor, complete the following steps:

To verify that the Athena query can be accessed for the table

1. Navigate to Amazon SageMaker Unified Studio using the URL from the Amazon SageMaker management console and log in using your SSO or AWS credentials.
2. Use the top center menu of the Amazon SageMaker home page to navigate to the project you want to use.
3. On the project page, under **Overview**, choose **Data**, and then choose **Lakehouse**.

4. Next to the new table, choose the options menu (three dots), and choose **Query with Athena**. You can also choose to preview the data.

Step 2: Get started with importing S3 data

Create or use an S3 bucket

In S3, create or have a bucket and note the bucket path, such as `s3://amzn-s3-demo-bucket`. You will upload your sample data to the existing bucket.

(Optional) Use sample data in your existing S3 bucket

Configure your S3 data using an existing bucket and sample data to upload and import.

Alternately, you can use the public bucket with the sample data location and skip this step.

To upload sample S3 data

Use the S3 console to upload the sample .parquet file from your local drive to your S3 source bucket.

1. Sign in to Amazon Simple Storage Service.
2. Navigate to the .parquet file that you downloaded from the public sample bucket.
3. Navigate to your existing S3 bucket and choose **Upload**. Upload the file to your S3 bucket.
4. Choose **Save**.

Edit your IAM project role and attach the S3 bucket policy

Configure your IAM role with a policy for S3 bucket permissions to allow the SageMaker project role to access your S3 source bucket. Use these steps to create and attach a resource-based bucket policy and configure permissions in Lakehouse.

To attach the S3 bucket policy to the project role

1. Using the account that is associated with the SageMaker domain, navigate to the IAM console, and choose **Roles**.
2. Choose the project role that is associated with your Amazon SageMaker Unified Studio project. This role was created automatically when you created your project in Amazon SageMaker

Unified Studio. This is the project role that you made a note of previously in Make a note of your IAM project role. Open the project role, such as `arn:aws:iam::ACCOUNT_ID:role/<datazone_usr_role_xxxxxxxxxxxxxx_yyyyyyyyyyyyyy>`.

3. Choose **Add permissions**, and then choose **Create inline policy**.
4. Choose **JSON**, and then paste the following policy statement into the Policy editor. The following is an example bucket policy. Replace ACCOUNT_ID with the account ID where the domain resides, <s3_bucket> with the name of the S3 bucket that you intend to query in SageMaker Unified Studio, and <datazone_usr_role_xxxxxxxxxxxxxx_yyyyyyyyyyyyyy> with the project role in SageMaker Unified Studio. For the purposes of this topic, replace the `amzn-s3-demo-bucket` value with the bucket name for the sample data (`s3://aws-bigdata-blog/generated_synthetic_reviews/*`).

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
            "Condition": {
                "ArnEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::ACCOUNT_ID:role/
<datazone_usr_role_xxxxxxxxxxxxxx_yyyyyyyyyyyyyy>"
                }
            }
        },
        {
            "Sid": "Statement2",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
                "ArnEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::ACCOUNT_ID:role/
<datazone_usr_role_xxxxxxxxxxxxxx_yyyyyyyyyyyyyy>"
                }
            }
```

```
        }
    ]
}
```

Open a new notebook and start an Apache Spark session to import the data

Configure your SageMaker spark session to import and query the S3 data using a Jupyter notebook in the console. To access the data through the unified JupyterLab experience with a spark session, complete the following steps:

1. Sign in to your SageMaker project.
2. Navigate to the **Project overview** page.
3. Choose **New**, and then choose **Notebook**.
4. Choose the default notebook titled `Untitled.jpynb`. Click the file name and type in the field to rename the file to `mynotebok.jpynb`.
5. On the SageMaker Unified Studio project page, on the top menu, choose **Build**. Under **IDE & APPLICATIONS**, choose **JupyterLab**.
6. Wait for the space to be ready.
7. Choose the plus sign and for **Notebook**, choose **Python3**.
8. In the notebook, switch the connection type to `PySpark` and choose `spark.fineGrained`.
9. Use the following command to initialize a Spark session.

```
from pyspark.sql import SparkSession
spark = SparkSession.builder.getOrCreate()
```

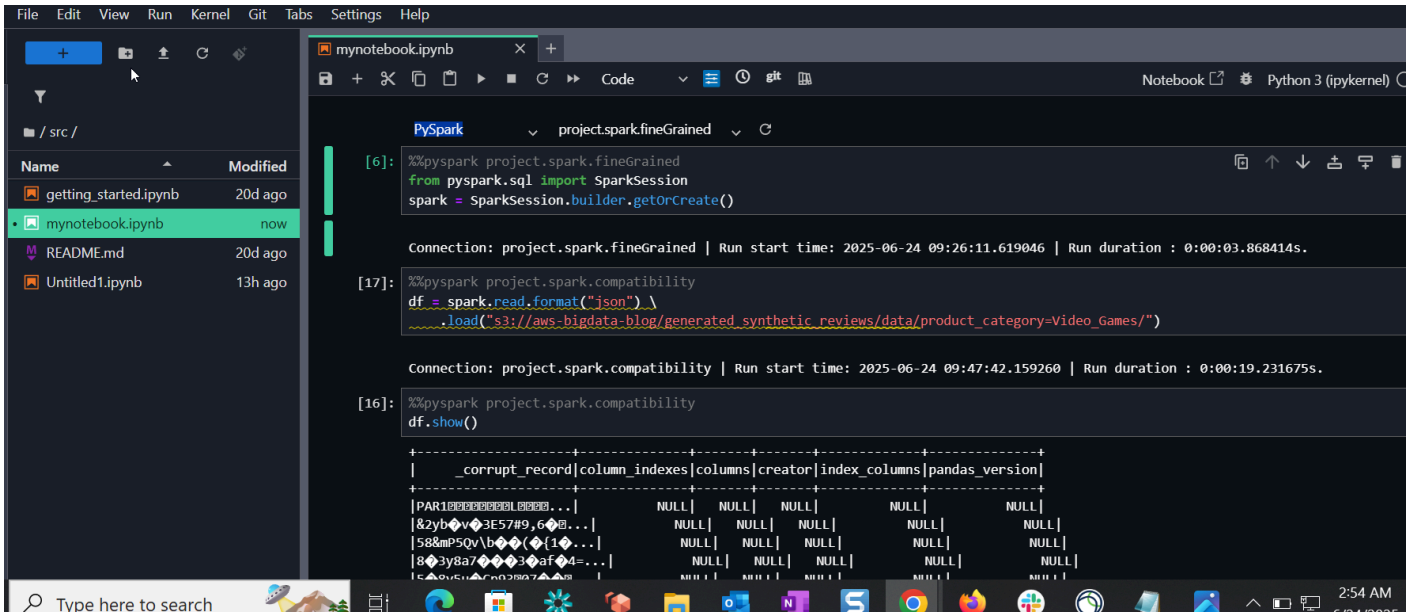
10. In the notebook, keep the connection type at `PySpark` and choose `spark.compatibility`.

    Use the following command in a cell to load the S3 source with the %%pyspark cell magic. This imports the S3 data. Make sure the second line is indented as shown.

```
%%pyspark
df = spark.read.format("json") \
    .load("s3://s3://aws-bigdata-blog/generated_synthetic_reviews/data/
product_category=Video_Games/")
```

11. Use the following command in a cell to query the S3 source with the %%pyspark cell magic. This queries the data.

```
%%pyspark
df.show()
```



Step 3: Get started with the query editor

You can use the query editor to perform analysis using SQL. The query editor tool provides a place to write and run queries, view results, and share your work with your team.

Prerequisites to access your project

Before you get started with the query editor, you must have access to Amazon SageMaker Unified Studio and create a project.

- Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or configure credentials with IAM Identity Center.

Query AWS Glue sample data using Amazon Athena in Amazon SageMaker Unified Studio

After you create a project, you can use the query editor to write and run queries. Use the following steps to create a table using a SQL query with Athena, query the table, and visualize the results.

1. In the Amazon SageMaker Unified Studio, navigate to your project.
2. On the project page, under **Overview**, choose **Data**.
3. Choose **Lakehouse**, Expand **AwsDataCatalog**, and then choose the three-dot action menu next to your database.

4. Choose **Query with Athena**.
5. Copy and paste the following SQL query into the editor. The following query will create a table `synthetic_reviews_video_games` and query it.

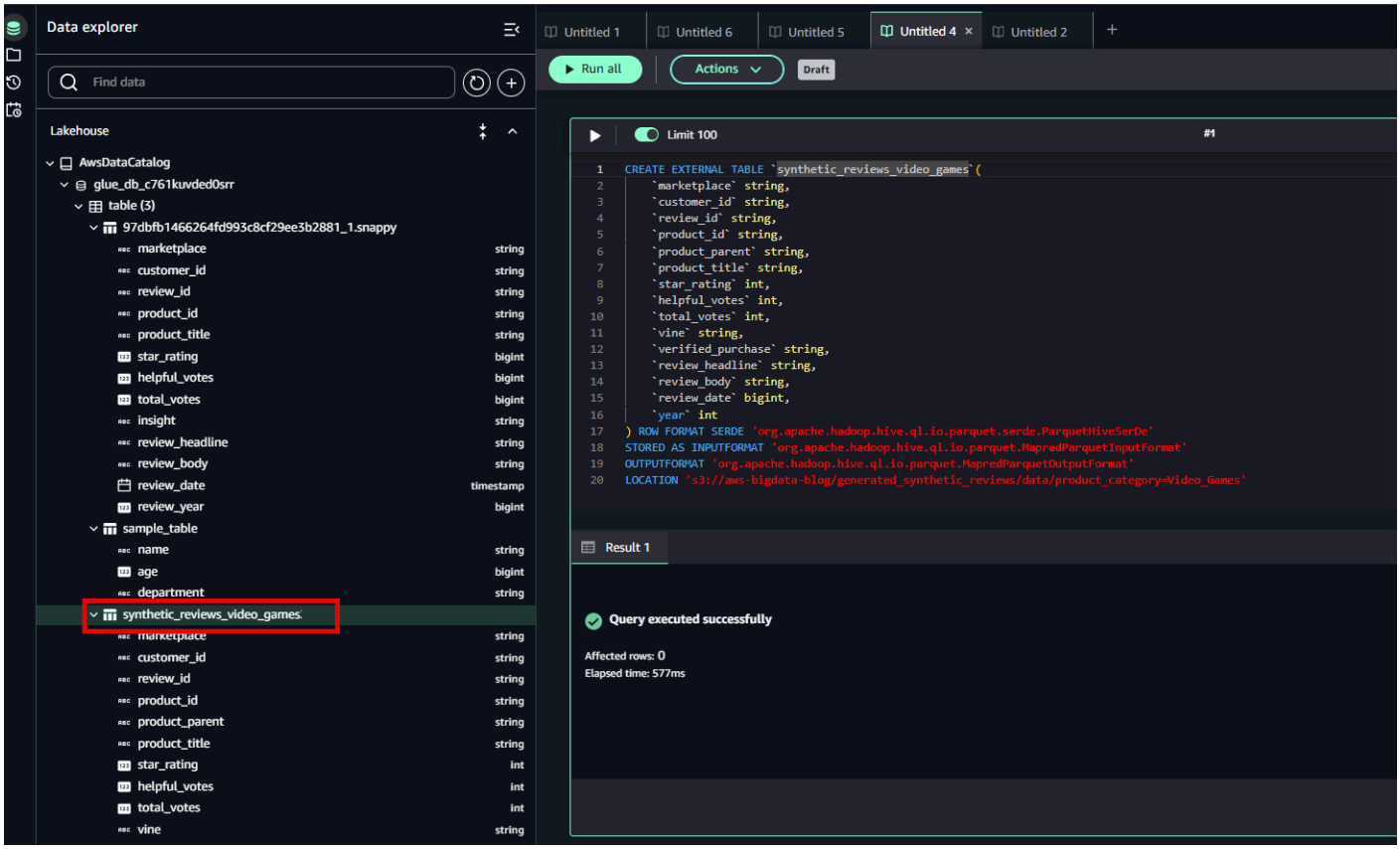
```
CREATE EXTERNAL TABLE `synthetic_reviews_video_games`(
    `marketplace` string,
    `customer_id` string,
    `review_id` string,
    `product_id` string,
    `product_parent` string,
    `product_title` string,
    `star_rating` int,
    `helpful_votes` int,
    `total_votes` int,
    `vine` string,
    `verified_purchase` string,
    `review_headline` string,
    `review_body` string,
    `review_date` bigint,
    `year` int
) ROW FORMAT SERDE 'org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe'
STORED AS INPUTFORMAT 'org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputFormat'
OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutputFormat'
LOCATION 's3://aws-bigdata-blog/generated_synthetic_reviews/data/
product_category=Video_Games'
```

> **Note**
>
> For simplicity, in this topic, these steps create a table under a specific partition folder instead of creating a top level table that includes all the partition folders. As a gene3ral recommendation, create tables at the top level.

The SQL query creates an external table named "synthetic_reviews_video_games" that maps to Amazon product review data stored in Parquet format. The table defines columns for marketplace, customer information, product details, ratings, and review content.

6. Choose the **Run cell** icon.

   When the query finishes running, a Result tab appears below the cell to display the outcome.



7. Refresh the **Data explorer** navigation pane, and view the table you created in the **Lakehouse** section.
8. Choose **Add SQL** to add another cell to the querybook. Then enter the following script:

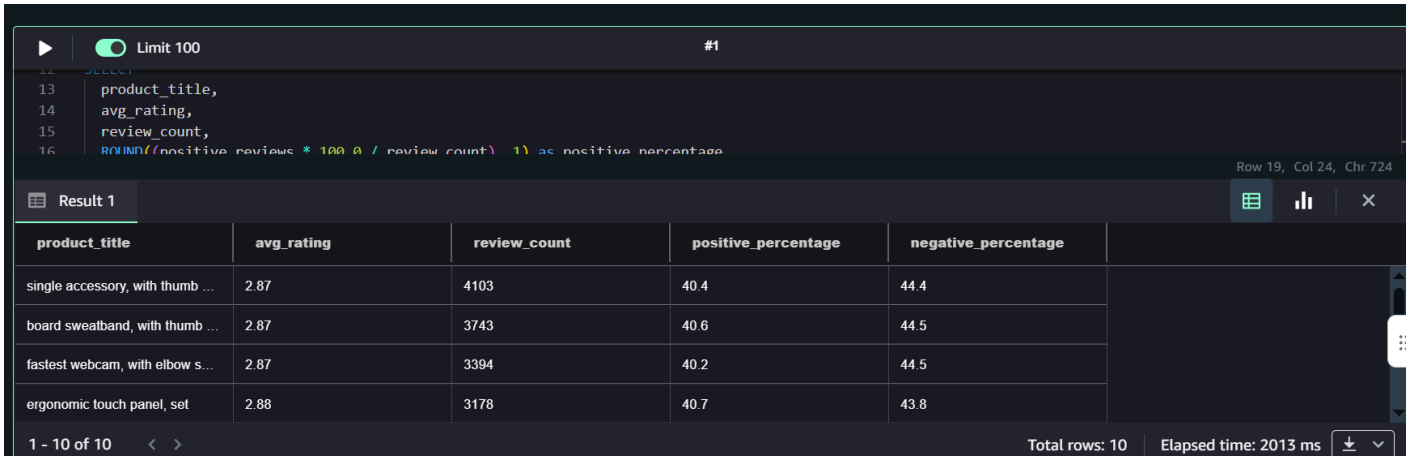
```
WITH review_stats AS (
```

```
  SELECT
    product_title,
    ROUND(AVG(star_rating), 2) as avg_rating,
    COUNT(*) as review_count,
    COUNT(CASE WHEN star_rating >= 4 then 1 END) as positive_reviews,
    COUNT(CASE WHEN star_rating <= 2 then 1 END) as negative_reviews
  FROM "awsdatacatalog"."glue_db_<database-ID>"."synthetic_reviews_video_games"
  GROUP BY product_title
  HAVING COUNT(*) >= 5
)
SELECT
  product_title,
  avg_rating,
  review_count,
  ROUND((positive_reviews * 100.0 / review_count), 1) as positive_percentage,
  ROUND((negative_reviews * 100.0 / review_count), 1) as negative_percentage
FROM review_stats
WHERE avg_rating >= 2.5
ORDER BY review_count DESC, avg_rating DESC
LIMIT 10;
```

This query completes the following tasks:

- Creates a CTE (Common Table Expression) to calculate review statistics
- Calculates average ratings, total review count, and counts of positive/negative reviews per game
- Filters for games with at least 5 reviews
- Computes the percentage of positive and negative reviews
- Shows only games with an average rating of 2.5 or higher
- Orders results by review count and average rating
- Returns the top 10 most reviewed, highly-rated games

The results will show you the most popular well-rated games in your dataset, along with meaningful metrics about their review distribution.



9. Choose the **Run cell** icon.

   In the **Results** tab, the first ten rows of the table you created are displayed.

10. In the **Results** tab, you can choose the **Chart view** icon. This opens up a chart view with a line graph as a default.

11. Set up the chart to display a pie chart. Choose **Trace**.

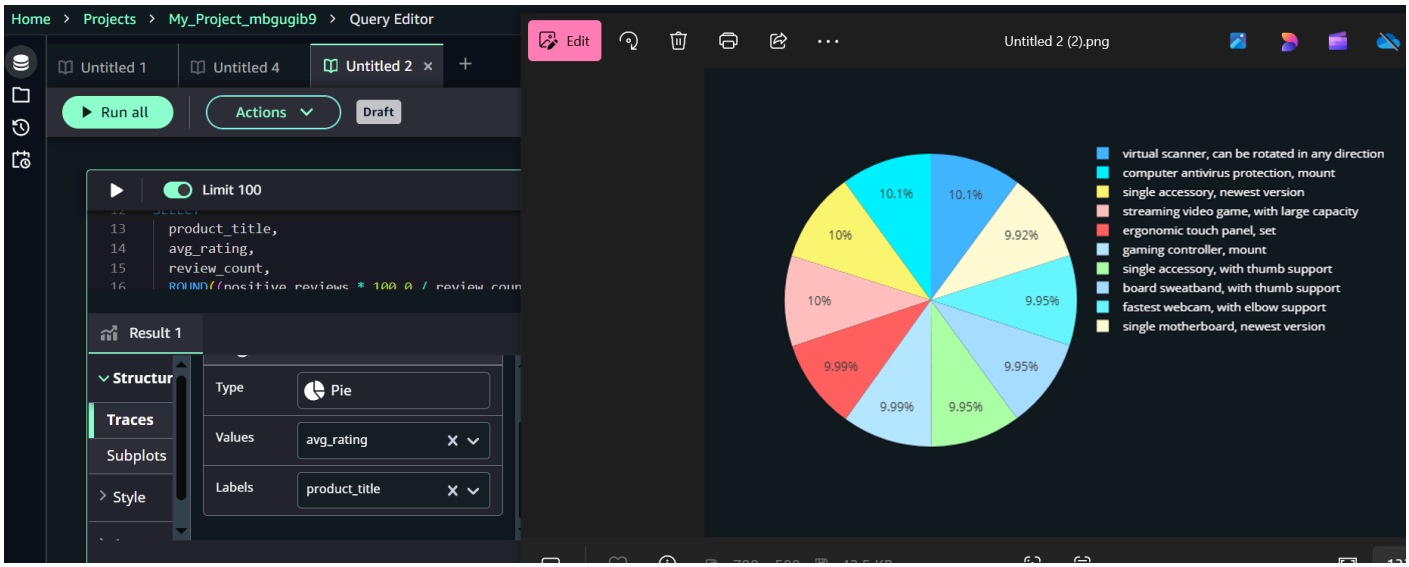
    a. For **Type**, choose **Pie**.

    b. For **Values**, choose **avg_rating**.

    c. For **Labels**, choose **product_title**.

    d. Choose the download arrow to view the chart.

    This displays a pie chart so you can visualize results.



After you've finished querying the data, you can choose to view the queries in your query history and save them to share with other project members.

Get started using EMR Serverless in Amazon SageMaker Unified Studio

Overview

Amazon EMR Serverless provide a powerful way to process data at scale without managing infrastructure. In addition to Amazon EMR on EC2 clusters, you can create and delete EMR Serverless applications directly from SageMaker Unified Studio. EMR Serverless applications operate similarly to traditional notebooks, letting you run queries and code while actively observing the output simultaneously.

Unlike traditional notebooks, the contents of an EMR notebook run in a client and are executed by a kernel in your EMR Serverless Application. This means you don't need to configure a cluster to run applications, and helps you avoid over or under provisioning resources for your jobs. EMR Serverless is ideal for applications that need responses quickly, such as interactive data analysis.

This architecture allows you to use a single EMR Serverless application on multiple clusters and run clusters on demand as it fits your use case and needs. These are seperate from Spark applications For more general information about EMR Serverless Notebooks and Applications, see the EMR Management Guide.

Getting started with EMR serverless applications

SageMaker Unified Studio provides a straightforward interface for creating EMR Serverless applications. In order to create a new EMR Serverless Application your admin needs to enable blueprints. For more information about the blueprint setup process see Enable or disable blueprints in the *Amazon Sagemaker Unified Studio Guide*. Once blueprints are enabled:

1. From the SageMaker Unified Studio UI, navigate to the Project Management view and then select your project from the project list.
2. Select **Compute** from the navigation bar, then select Data processing. Select the Add Compute button. You'll be prompted to connect to an existing compute resource or create new compute resources. From there, select EMR Serverless.
3. On the **Add Compute** screen, you'll add your compute resource's name, description, and release label. You will also be prompted to select a permission mode. Your options are compatibility and fine-grained.

- Compatibility mode. This permission mode allows your project to be compatible with data managed using full-table access, meaning the compute engine can access all rows and columns in the data. Choosing this option configures your compute to work with data assets from AWS and from external systems that you connect to from your project.
- Fine-grained mode. This option is for data managed using fine-grained access, meaning the compute engine can only access specific rows and columns from the full dataset. Choosing this option configures your compute to work with data asset subscriptions from Amazon SageMaker catalog.

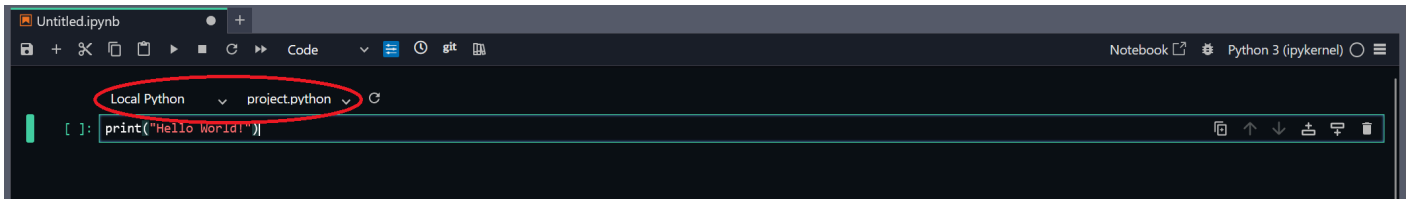
Your EMR Serverless compute will now be listed in your Data processing list. From here, you can connect to your EMR Serverless compute while running a notebook.

Connecting to an EMR Serverless compute

Once you have an EMR serverless compute added, you can connect to the compute directly from the Sagemaker Unified Studio notebook workspace.

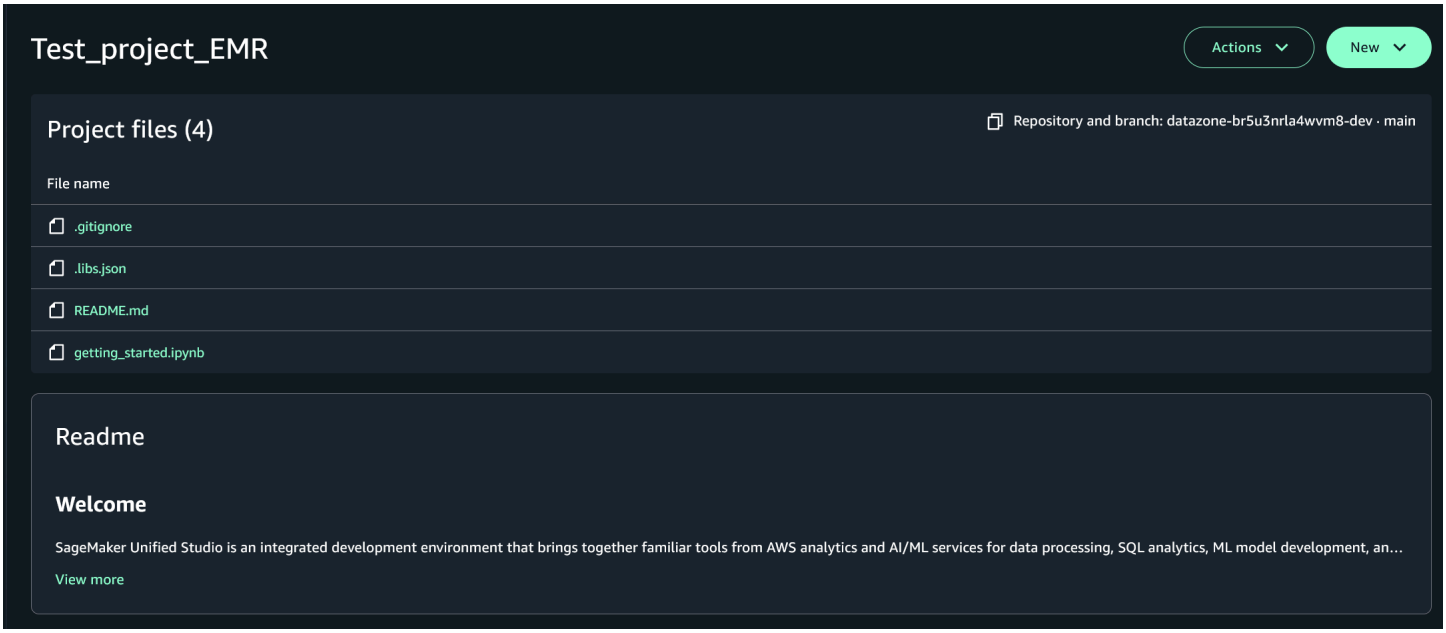
To connect to an EMR Serverless compute:

1. Above a code block in your Jupyter Notebook, there will be two drop down boxes. One lets you select your connection type, the other your compute.



2. Select the connection type "PySpark" and then click on the drop down for your Compute. From here you can select your EMR Serverless compute from the second drop down box.
3. Run the code in your code block. The first time you run code, it will connect to the compute and start a session for your connection. This means that you are connected to the serverless compute, and all codeblocks using this EMR compute this session will use this connection.

For first-time users, we recommend starting with the EMR example notebook provided (getting_started.ipynb), which demonstrates basic operations and best practices. You can access this notebook from the Examples tab in the Unified Studio file browser, pictured below:



Due to the nature of EMR Serverless applications, you can have multiple computes available at once. This allows you to maintain your notebook code while connecting to different EMR applications as needed for various workloads. You can switch between applications without modifying your notebook code, allowing you to test different configurations or work with different data processing requirements.

(Optional) Remove or stop an EMR application

While using EMR serverless compute, you may need to stop using a particular compute either for a period of time or permanently. You can remove or stop EMR serverless computes in those cases. Stopping a compute lets you pause a compute until you want to reactiveate it Applications in this paused state can be reactivated whenver you want, with all definition remaining. You only incure storage costs for stopped applications

For applications you don't intend to use again, you can delete, or remove them. This will permanently delete the application, and cannot be undone. To access the application again you will need to recreate it. Deleting an application removes all costs associated with it including storage cost.

EMR Serverless computes can be removed from projects via the Data processing tab in your project view. Simply click the menu to the right of the compute's name and click Remove. Removed EMR Serverless computes are deleted. You can also manually stop an EMR Serverless compute by using the EMR Studio page on the AWS Console. For more information see Manage applications from the EMR Studio console in the EMR Serverless user guide.

Get started using Amazon Bedrock in SageMaker Unified Studio

Amazon Bedrock in SageMaker Unified Studio offers multiple playgrounds that allow you to easily access and experiment with Amazon Bedrock models. With the chat playground, you can chat with a model through text and image prompts. With the image and video playground, you can use a compatible model to generate and edit images and videos.

In addition to the playgrounds, you can also use Amazon Bedrock in SageMaker Unified Studio to create chat agent apps and flows apps. A chat agent app allows users to create a custom app that interacts with a Amazon Bedrock model through a conversational interface. You can enhance chat agent apps with Amazon Bedrock features such as data sources and guardrails and share the app with other users. A flows app allows users to link together prompts, foundation models, and other components to create a visual, end-to-end generative AI workflow.

The following section will walk you through the basic functionalities of Amazon Bedrock in SageMaker Unified Studio. First, you will select a model from the model catalog and chat with it in the chat playground. Then, you will create a chat agent app that can create playlists for a rock and pop radio station. For more in-depth information on other Amazon Bedrock features you can use with Amazon Bedrock in SageMaker Unified Studio, see Amazon Bedrock in SageMaker Unified Studio.

Step 1: Explore Amazon Bedrock foundation models

The following section shows how to select a model from the model catalog in the Amazon Bedrock in SageMaker Unified Studio playground. You can also access the model catalog from inside your projects. The models you have access to in your projects might be different from those you can access in the playground, based on your administrator's settings. To check which models you can access in a project, open or create a project, and then select **Models** in the navigation pane to open the model catalog.

To open the model catalog in the playground

1. Navigate to the Amazon SageMaker landing page by using the URL from your admininstrator.
2. Access Amazon SageMaker using your IAM or single sign-on (SSO) credentials. For more information, see Access Amazon SageMaker Unified Studio.
3. At the top of the page, choose **Discover**.

4. Under **Data and model catalog**, choose **Amazon Bedrock models**. This opens the model catalog in the Amazon Bedrock in SageMaker Unified Studio playground.
5. (Optional) Choose **Group by: Modality** and select **Provider** to sort the list by model provider.
6. Choose a model from the list of models that you have access to. For information about a model, choose **View full model details** in the information panel. If you don't have access to an appropriate model, contact your administrator. Some features may not be supported by all models.

If you are ready to begin chatting with the model you chose, proceed to the following step.

Step 2: Chat with a model in the chat playground

In this section you will chat with your selected model in the chat playground. You chat by sending a prompt to the model and receiving a response. For more information, see Experiment with the Amazon Bedrock playgrounds.

> **Warning**
>
> Generative AI may give inaccurate responses. Avoid sharing sensitive information. Chats may be visible to others in your organization.

To chat with a model

1. In the chat playground, enter **`What is Avebury stone circle?`** in the **Enter prompt** text box.
2. (Optional) If the model you chose is a reasoning model, you can choose **Reason** to have the model include its reasoning in the reponse. For more information, see Enhance model responses with model reasoning in the *Amazon Bedrock user guide*.
3. Press Enter on your keyboard, or choose the run button, to send the prompt to the model. The response from the model will be generated in the playground.
4. Continue chatting with the model by entering the prompt **`Is there a museum there?`**.

   The model will use the previous prompt as context for generating its response to this question.
5. (Optional) Compare the output from multiple models, or shared apps.

   a. In the playground, turn on **Compare mode**. This will open two panes side-by-side.

b. In each panes, select a model that you want to compare. If you want to use a shared app, select **App** in **Type** and then select the app in **App**.

c. Enter a prompt in the text box and run the prompt. The output from each model is shown in their respective panes. You can choose the copy icon to copy the prompt or model response to the clipboard.

d. (Optional) Choose **Add chat window** to add a third window. You can compare up to 3 models or apps.

e. Turn off **Compare mode** to stop comparing models.

6. Choose **Reset** to start a new chat with the model.

Step 3: Create a chat agent app

In this section you will learn how to create a simple Amazon Bedrock in SageMaker Unified Studio chat agent app that creates playlists for a radio station and shares the dates and locations of upcoming shows.

To create an Amazon Bedrock chat agent app

1. On the Amazon SageMaker home page, choose **Build chat agent app** to create a new chat agent app. The **Select or create a new project to continue** dialog box opens.
2. In the **Select or create a new project to continue** dialog box, do one of the following:

   - If you want to use a new project, follow the instructions at Step 2 - Create a new project. For the **Project profile** in step 1, choose **Generative AI application development**.
   - If you want to use an existing project, select the project that you want to use and then choose **Continue**.
3. On the app creation page, an untitled app will automatically be created for you. In **Untitled App - nnnn**, enter **`Radio show`** as the name for your app.
4. In the **Configs** pane, do the following:

   a. For **Model**, select a model that supports Guardrails, Data, and Function components. The description of the model tells you the components that a model supports. For full information about the model, choose **View full model details** in the information panel. For more information, see Find serverless models with the model catalog. If you don't have access to an appropriate model, contact your administrator. Different models might not support all features.

b. For **Enter a system instruction** in **Instructions for chat agent & examples**, enter **`You are a chat agent app that creates 2 hour long playlists for a radio station that plays rock and pop music.`**.

c. In the **UI** section, update the user interface for the app by doing the following:

i. In **Hint text for empty chat** enter **`Hi! I'm your radio show playlist creator.`**.

ii. In **Hint text for user input** enter **`Enter a prompt that describes the playlist that you want.`**.

iii. In **Quick start prompts** choose **Edit**.

iv. Choose **Reset** to clear the list of quick start prompts

v. For **Quick-start prompt 1**, enter **`Create a playlist of pop music songs.`**.

vi. (Optional). Enter quick start prompts of your choice in the remaining quick start prompt text boxes.

vii. Choose **Back to configs**.

5. Choose **Save** to save the current working draft of your app.

6. In the **Quick start prompts** section of the **Preview** pane, run the quick start prompt that you just created by choosing the prompt.

   The app shows the prompt and the response from the model in the **Preview** pane.

7. In the prompt text box (the text should read **Enter a prompt that describes the playlist that you want**), enter **`Create a playlist of songs where each song on the list is related to the next song, by musician, bands, or other connections. Be sure to explain the connection from one song to the next.`** .

8. Choose the run button (or press Enter on your keyboard) to send the prompt to the model.

You have now created a basic chat agent app that can create playlists for a rock and pop radio station. You can experiment with sending prompts and receiving responses from your chat agent app.

Additional capabilities

Amazon Bedrock in SageMaker Unified Studio offers many additional capabilities to the ones covered in this walkthrough, including the following.

- You can customize and influence model behavior using inference parameters and system prompts. For more information, see What is a prompt?.
- You can enhance your chat agent app by adding data sources and guardrails. For more information, see Build a chat agent app.
- You can share your chat agent app with other users and use it as a component in a flows app. For more information, see Share a chat agent app and Deploy a chat agent app.
- You can create a flows app to link together different components such as knowledge bases and reusable prompts. For more information, see Build a flow app.

Get started with Amazon S3 Tables in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio provides integrated support for S3 Tables, allowing you to create S3 table buckets and Apache Iceberg tables in those buckets.

Amazon S3 Tables provide S3 storage that's optimized for analytics workloads, with built-in Apache Iceberg support and features designed to continuously improve query performance and reduce storage costs for tables. Data in S3 Tables is stored in table buckets, which are specialized buckets for storing tabular data. For more information, see Working with Amazon S3 Tables and table buckets.

You can begin working with S3 Tables directly by creating an S3 table bucket as a new data source within Amazon SageMaker Unified Studio.

Integrating S3 with AWS analytics services through Amazon SageMaker Unified Studio

Amazon S3 table buckets integrate with AWS Glue Data Catalog and AWS Lake Formation to allow AWS analytics services to automatically discover and access your table data. For more information, see Integrating Amazon S3 Tables with AWS analytics services.

If you've never used S3 Tables before in the current Region, you can allow Amazon SageMaker to enable the S3 Tables analytics integration when you create a new S3 Tables catalog in the Amazon SageMaker Unified Studio console.

When you allow Amazon SageMaker Unified Studio to perform the integration, Amazon SageMaker takes the following actions on your behalf in your account:

- Creates a new AWS AWS Identity and Access Management (IAM) service role that gives Lake Formation access to all your tables and table buckets in your current Region. This allows Lake Formation to manage access, permissions, and governance for all current and future table buckets in that Region.
- Creates the `S3tablescatalog` in the AWS Glue Data Catalog in your current Region without privileged access.

- Adds the Amazon Redshift service role (`AWSServiceRoleForRedshift`) as a Lake Formation Read-only administrator. This allows Amazon Redshift to automatically mount all tables in S3 table buckets in the Region.

> **Note**
>
> Integration will be performed in the current Region only.

Prerequisites

- Create a Amazon SageMaker domain and project. For more information, see Setting up Amazon SageMaker.

Creating S3 Tables catalogs in Amazon SageMaker Unified Studio

To get started using S3 Tables in Amazon SageMaker Unified Studio you create a new Lakehouse catalog with S3 table bucket source using the following steps.

1. Open the Amazon SageMaker at https://console.aws.amazon.com/sagemaker/ and use the Region selector in the top navigation bar to choose the appropriate AWS Region.
2. Select your Amazon SageMaker domain.
3. Select the project you want to create a table bucket in.
4. In the navigation menu select **Data**, then select **+** to add a new data source.
5. select **Create Lakehouse catalog**.
6. In the add catalog menu, choose **S3 Tables** as the source.
7. Enter a name for the catalog, and a database name.
8. Choose **Create catalog**. This creates the following resources in your account:
   a. A new S3 Table bucket and the corresponding AWS Glue child catalog under the parent catalog `s3tablescatalog`.
   b. A new database within that AWS Glue child catalog. The database name will match the database name you provided. In S3 tables, this is the table namespace.

9. Begin creating tables in your database and querying them using query editor or Jupyter notebook.

Creating and Querying S3 Tables

After you add an S3 Tables catalog it can be queried as `s3tablescatalog/`*`your-bucket-name`*. You can begin creating S3 tables in the catalog and querying them in Amazon SageMaker Unified Studio with the Query editor and Jupyterlab.

> **Note**
>
> You can only create S3 tables in Amazon SageMaker Unified Studio with Athena engine or Spark. Once created, you can query tables with Athena, Amazon Redshift, or Spark.

Using the Query Editor

1. Navigate to the project you created in the top center menu of the Amazon SageMaker Unified Studio home page.
2. Expand the **Build** menu in the top navigation bar, then choose **Query editor**.
3. Create a new querybook tab. A querybook is a kind of SQL notebook where you can draw from multiple engines to design and visualize data analytics solutions.
4. Select a data source for your queries by using the menu in the upper-right corner of the querybook.
   a. Under **Connections**, choose **Lakehouse (Athena)** to connect to your Lakehouse resources.
   b. Under **Catalogs**, choose `s3tablescatalog/{your-table-bucket}`
   c. Under **Databases**, choose the name of the database for your S3 tables.
5. Select **Choose** to connect to the database and query engine.
6. Enter SQL to create your first table, the following is an example SQL query:

```
CREATE TABLE daily_sales (
    sale_date date,
    product_category string,
    sales_price double
)
```

```
PARTITIONED BY (month(sale_date))
TBLPROPERTIES ('table_type' = 'iceberg')
```

- After you create the table you can browse to it in the **Data explorer** by choosing **S3tablescatalog** → *your-bucket-name* → *example_database* → *example_table*

7. Insert data into a table with the following query.

```
INSERT INTO daily_sales
VALUES (DATE '2024-01-15', 'Monitor', 900.00),
(DATE '2024-01-14', 'Keyboard', 250.00),
(DATE '2024-01-16', 'CPU', 1350.00)
;
```

8. Select data from a table with the following query.

```
SELECT *
FROM daily_sales
WHERE sale_date BETWEEN DATE '2024-01-14' AND DATE '2024-01-16' ORDER BY
 sale_date;
```

To learn more about the query editor and see more SQL examples, see: Get started with the query editor in Amazon SageMaker Unified Studio

Using JupyterLab

1. Navigate to the project you created in the top center menu of the Amazon SageMaker Unified Studio home page.
2. Expand the **Build** menu in the top navigation bar, then choose **JupyterLab**.
3. Create a new notebook.
4. Select engine you want to use
5. Select your table bucket and namespace as the data source for your queries:

   a. For Spark engine, execute query `USE S3tablescatalog_`*example-table-bucket*

   b. For Athena or Amazon Redshift engine, use the following configure magic. For more information, see Configure compute resources in JupyterLab in the *SageMaker AI Unified Studio User Guide*.

```
%%configure -n project.athena -f
```

```
{
    "catalog_name": "s3tablescatalog/examples-table-bucket",
    "schema_name": "example-namespace"
}
```

6. Enter SQL queries into the notebook cell to create a table in the database.

> **Important**
>
> When using the Spark engine through a Spark connection, the `S3TableFullAccess` permission is required for table creation. For more information, refer to Considerations for enabling Lake Formation permissions in the *AWS Glue Developer Guide*.

The following are examples of basic SQL queries you can use to start working with tables.

Create a new table

```
CREATE TABLE daily_sales (
    sale_date date,
    product_category string,
    sales_price double
)
PARTITIONED BY (month(sale_date))
TBLPROPERTIES ('table_type' = 'iceberg')
```

After you create the table you can browse to it in the **Data explorer** by choosing **S3tablescatalog** → ***your-bucket-name*** → ***your-database-name*** → ***daily_sales***

Insert data into a table

```
INSERT INTO daily_sales
VALUES
(DATE '2024-01-15', 'Monitor', 900.00),
(DATE '2024-01-14', 'Keyboard', 250.00),
(DATE '2024-01-16', 'CPU', 1350.00)
;
```

Select data from a table

```
SELECT * FROM daily_sales
```

```
WHERE sale_date BETWEEN DATE '2024-01-14' AND DATE '2024-01-16'
ORDER BY sale_date;
```

Drop a table

```
DROP TABLE IF EXISTS sample_table;
```

Get started with SageMaker Lakehouse integrated access controls for Athena federated queries in Amazon SageMaker Unified Studio

Scaling data infrastructure creates challenges with data silos, fragmented access controls, and complex connectivity requirements. Data analysts need to access information across multiple storage systems but are frequently hindered by:

- **Complex connectivity setup** - Configuring connections to various data sources requires technical expertise and access to configuration details that analysts may not have.
- **Fragmented governance** - Different data sources have their own access control mechanisms, making consistent security policies difficult to implement.
- **Data duplication** - Copying data between systems for analysis increases costs and creates data consistency risks.

To address the challenges of data silos and fragmented access, SageMaker Lakehouse with integrated access controls for Amazon Athena (Athena) federated queries offers:

- Streamlining the creation of connections to diverse data sources through a unified interface
- Centralizing access control management through AWS Lake Formation
- Enabling in-place querying through federated catalogs without data movement
- Providing fine-grained permissions at the catalog, database, table, and column levels
- Exploring data for ad hoc reporting and proof of concept before setting up new zero-ETL pipelines

SageMaker Lakehouse provides a unified environment for accessing, discovering, preparing, and analyzing data from various sources for machine learning (ML) and analytics workloads. Athena complements this as a serverless query service that analyzes data lake and federated data sources such as Amazon DynamoDB and PostgreSQL, through using SQL without extract, transform, and load (ETL) scripts. Federated connections in SageMaker Lakehouse establish secure links to external data sources, enabling access without data movement. Federated catalogs organize metadata about these connected data sources, making them discoverable and queryable through the

SageMaker Lakehouse interface. Federated queries use these connections to run SQL statements across multiple data sources simultaneously, breaking down data silos for comprehensive analysis.

What you'll learn

This guide shows you how to use SageMaker Lakehouse with integrated access controls for Athena federated queries. In this guide, you create an environment where data analysts can discover and query data across sources while administrators maintain consistent governance and appropriate security controls. This guide includes the following steps:

1. Set up federated connections between SageMaker Lakehouse and DynamoDB.
   - Create connections that serve as bridges between your SageMaker Lakehouse and external data sources.
   - Enable seamless data access while maintaining security boundaries.
   - Learn how connections eliminate the need for data movement or duplication.
2. Create federated catalogs for data discovery.
   - Establish catalogs that contains metadata and views about tables from your connected data sources.
   - Access data from the connected data source within your SageMaker Lakehouse environment.
   - Make external tables queryable through the Lakehouse interface.
   - Use catalogs as directories of available data assets to simplify discovery and access.
3. Implement column-level security using AWS Lake Formation
   - Configure fine-grained permissions for sensitive data.
   - Apply data access controls based on user roles and responsibilities.
   - Ensure consistent security policies across all data sources.
4. Validate security controls through Athena queries
   - Test access permissions with different user personas.
   - Verify that you properly protect sensitive data.
   - Confirm that authorized users can access appropriate data.

Prerequisites

Before you begin, make sure you have the following:

- An AWS account with permission to create IAM roles and IAM policies.
- An AWS Identity and Access Management (IAM) user with an access key and secret key to configure the AWS Command Line Interface (AWS CLI).
- Your administrator role added as a data lake administrator in AWS Lake Formation. For more information about how to create and add a data lake administrator, see Create a data lake administrator in *AWS Lake Formation Developer Guide*.
- Administrator access to Amazon SageMaker Unified Studio. For more information about permissions of the administrator role, see Lake Formation personas and IAM permissions reference in the *AWS Lake Formation Developer Guide*. For more information about using the IAM Identity Center directory as your identity source, see Configure user access with the default IAM Identity Center directory in the *AWS IAM Identity Center User Guide*. For more information about how to access SageMaker, see Accessing Amazon SageMaker Unified Studio in the *Amazon SageMaker Unified Studio Administrator Guide*.
- A SageMaker Unified Studio domain with the SQL Analytics profile enabled. For more information about creating an Amazon SageMaker Unified Studio domain and a project, see Setting up Amazon SageMaker in the *Amazon SageMaker User Guide*. For more information about SQL analytics project profile, see SAQL analytics project profile in the *Amazon SageMaker Unified Studio Administrator Guide*.

> **Note**
>
> Add your administrator as an SSO user to your domain. For more information about how to add an SSO user as a root domain owner, see Step 1 - Create an Amazon SageMaker unified domain in the *Amazon SageMaker User Guide* and Managing users in Amazon SageMaker Unified Studio in the *Amazon SageMaker Unified Studio Administrator Guide*.

- Two SageMaker Unified Studio projects set up for this guide:
  - An Admin project for creating connections. This project has a SQL analytics project profile.
  - A Data Analyst project for analyzing data, which includes both administrator and analysts as members. This project has a SQL analytics project profile.

  For more information about how to create a project in SageMaker Unified Studio, see Setting up Amazon SageMaker in the *Amazon SageMaker User Guide*.

Note

- To find the project role ARN for each project, in the SageMaker Unified Studio, choose the name of the project, choose **Project overview**, and find **Project role ARN** under **Project details**. For more information, see Get project details in the *Amazon SageMaker Unified Studio User Guide*.
- For more information about how to add members to your projects, see Add project members in the *Amazon SageMaker Unified Studio User Guide*.

- Administrator access to a data source. SageMaker Lakehouse connections support several popular data sources, such as Amazon DynamoDB, PostgreSQL, and Amazon DocumentDB. In this guide, we use DynamoDB as the data source.
  - To set up data sources in DynamoDB:
    - You can create a new table in DynamoDB with the partition key `cust_id` and the sort key `zipcode` and another column `mobile` through AWS CloudShell by using the following command:

```
aws dynamodb create-table \
  --table-name customer_ddb \
  --attribute-definitions \
AttributeName=cust_id,AttributeType=N \
AttributeName=zipcode,AttributeType=N \
  --key-schema \
AttributeName=cust_id,KeyType=HASH \
AttributeName=zipcode,KeyType=RANGE \
  --provisioned-throughput \
ReadCapacityUnits=5,WriteCapacityUnits=5 \
  --table-class STANDARD
```

    - You can populate the DynamoDB table with sample data by using the following commands:

```
# First item
aws dynamodb put-item \
  --table-name customer_ddb \
```

```
  --item '{"cust_id": {"N": "11"}, "zipcode": {"N": "2000"}, "mobile": {"N":
 "11113333"}}'

# Second item
aws dynamodb put-item \
  --table-name customer_ddb \
  --item '{"cust_id": {"N": "12"}, "zipcode": {"N": "2000"}, "mobile": {"N":
 "22224444"}}'

# Third item
aws dynamodb put-item \
  --table-name customer_ddb \
  --item '{"cust_id": {"N": "13"}, "zipcode": {"N": "3000"}, "mobile": {"N":
 "33335555"}}'

# Fourth item
aws dynamodb put-item \
  --table-name customer_ddb \
  --item '{"cust_id": {"N": "14"}, "zipcode": {"N": "4000"}, "mobile": {"N":
 "55556666"}}'
```

For more information about setting up a DynamoDB data source by using AWS CloudShell, see Amazon DynamoDB tutorial for AWS Cloud9 in the *AWS Cloud9 User Guide*.

- To allow the appropriate actions for the SageMaker Unified Studio projects to take on your DynamoDB data source, add a resource-based policy to your DynamoDB data source. Attach the following policy for the table `customer_ddb`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
              "AWS": [

 "arn:aws:iam::111122223333:role/
datazone_usr_role_xxxxxxxxxxxxxx_yyyyyyyyyyyyyy",

 "arn:aws:iam::111122223333:role/datazone_usr_role_zzzzzzzzzzzzz_aaaaaaaaaaaaa"
             ]
```

```
            },
            "Action": [
                "dynamodb:Query",
                "dynamodb:Scan",
                "dynamodb:DescribeTable",
                "dynamodb:PartiQLSelect",
                "dynamodb:BatchWriteItem"
            ],
            "Resource": "arn:aws:dynamodb:us-
west-2:111122223333:table/customer_ddb"
        }
    ]
}
```

This example policy allows connecting to DynamoDB tables as a federated source. Replace *us-west-2* with your AWS Region, *111122223333* with the AWS account ID where DynamoDB is deployed, *customer_ddb* with the DynamoDB table that you intend to query from SageMaker Unified Studio, *datazone_usr_role_xxxxxxxxxxxxxx_yyyyyyyyyyyyyyy* with the admin project role, and *datazone_usr_role_zzzzzzzzzzzzzz_aaaaaaaaaaaaaa* with the data analyst project role in SageMaker Unified Studio. For more information about how to attach a policy to a DynamoDB data source, see Attach a policy to a DynamoDB existing table in the *Amazon DynamoDB Developer Guide*.

Step 1: Set up federated catalogs

The first step is to set up federated catalogs for our data sources using an administrator account.

To set up federated catalogs

1. On the SageMaker Unified Studio console, for the domain you created in the prerequisite, choose **Open unified studio**.
2. Choose your admin project name under **Your projects**.
3. Choose **Data** in the navigation pane.
4. In the **Data explorer**, choose the plus icon to add a data source.
5. Under **Add data**, choose **Add connection**, choose **Next**.
6. Choose **Amazon DynamoDB**, and choose **Next**.

7. For **Name**, enter the name for your data source of DynamoDB.
8. Choose **Add data**.

SageMaker Unified Studio connects to the DynamoDB data source that you created in the prerequisites, registers the data source as a federated catalog with SageMaker Lakehouse, and displays it in your data explorer. The catalog references your DynamoDB data source.

To explore and query your data

1. Choose your admin project from SageMaker Unified Studio.
2. Choose **Data** in the navigation pane.
3. Choose the SageMaker Lakehouse catalog that you just created to view its contents. Use the data explorer to drill down to a table and choose **Query with Athena**.
4. In the query editor, run a sample SQL query to understand your data.

   For example, run the following query. Replace *your_federated_catalog_name* with the name of the federated catalog that you just created, *default* with the name of your database, and *your_table_name* with the name of your DynamoDB table. To learn more, see SQL analytics in the *Amazon SageMaker Unified Studio User Guide*.

   ```
   select * from your_federated_catalog_name.default.your_table_name limit 10;
   ```

Note

Access to the data source in the SageMaker Unified Studio project is governed by the policies for the project role. Users whoever become the member of this admin project use the same project role ARN and have the same full access level permissions to the data source. For more information about how to add members to your projects, see Add project members in the *Amazon SageMaker Unified Studio User Guide*. To grant fine-grained access permissions to different user personas, such as data analysts, create a separate data analyst project and add the data analyst users as project members of the data analyst project. Step 2 shows how to set up the fine-grained permissions.

For more information about creating connections in SageMaker Lakehouse, see Creating a connection in SageMaker Lakehouse in the *Amazon SageMaker Unified Studio User Guide*. For more information about creating catalogs, see Creating a catalog in the *Amazon SageMaker Unified Studio User Guide*.

Step 2: Set up fine-grained access permissions on federated catalogs

Security is a critical aspect of data access. SageMaker Lakehouse provides integrated access controls that work with federated queries in Athena to ensure proper governance. You can manage permissions at the catalog, database, and table levels. Administrators can apply access controls at different levels of granularity to ensure sensitive data remains protected while expanding data access.

This step is to delegate access permissions on your DynamoDB federated catalogs to other users. You grant permissions to the data analyst persona. To set up the fine-grained access permissions to the data analyst persona, you need to add permissions on your DynamoDB federated catalogs to the SageMaker Unified Studio data analyst project role that you created in the prerequisites section. This will ensure that access controls that you specify are enforced when the data is queried. For more information about the Lake Formation personas and IAM permissions, see Lake Formation personas and IAM permissions reference in the *AWS Lake Formation Developer Guide*.

To set up fine-grained access permissions on federated catalog and database

1. Navigate to Lake Formation in the AWS Management Console as an administrator.
2. In the Lake Formation console, under **Data Catalog** in the navigation pane, choose **Catalogs**.
3. Choose the federated catalog name that you set up in Step 1: Set up federated catalogs. You'll see the databases.
4. Choose the database name in the catalog. You can see details for the database and manage permissions.
5. To set up permissions for the federated catalog and database to your SageMaker Unified Studio data analyst project (the data analyst project that you set up in prerequisites), from the **Actions** menu, choose **Grant**.
6. For **Principal type**, choose **Principals**.
7. For **Principals**, choose **IAM users and roles**.

8. For **IAM users and roles**, choose the project role ARN that you got from your data analyst project in the prerequisites section.
9. For **LF-Tags or catalog resources**, choose **Named Data Catalog resources**.
10. For **Catalogs**, choose the federated catalog name for the source (the federated catalog that you set up in Step 1) to grant permissions on.
11. For **Databases**, the console populates the databases for your DynamoDB data source.
12. For **Database permissions - Database permissions**, select **Describe**.
13. Choose **Grant**.

To set up fine-grained access permissions on the tables

For example, if you wish to restrict access to a sensitive column containing the mobile phone number for each customer, the steps are as follows.

1. Navigate to Lake Formation in the AWS Management Console as an administrator.
2. In the Lake Formation console, under **Data Catalog** in the navigation pane, choose **Tables**.
3. Under **Choose catalog**, choose the federated catalog name that you set up in Step 1.
4. Choose the table name in the catalog. You can see details for the table and manage permissions.
5. From the **Actions** menu, choose **Grant**.
6. For **Principal type**, choose **Principals**.
7. For **Principals**, choose **IAM users and roles**.
8. For **IAM users and roles**, choose the project role ARN that you got from your data analyst project in the prerequisites section.
9. For **LF-Tags or catalog resources**, choose **Named Data Catalog resources**.
10. For **Catalogs**, choose the federated catalog name for the source (the federated catalog that you set up in Step 1) to grant permissions on.
11. For **Databases**, the console populates the databases for our DynamoDB data source.
12. For **Tables**, the console populates the tables for your DynamoDB data source.
13. For **Table permissions - Table permissions**, select **Select**.
14. For **Data permissions**, choose **Column-based access**.
15. For **Choose permission filter**, choose **Include columns**.
16. For **Select columns**, choose columns `zipcode` and `cust_id`.

17. Choose **Grant**.

In this example, we demonstrate how to set up a basic column-level filter to restrict access to sensitive data. However, SageMaker Lakehouse supports a broad range of fine-grained access control scenarios beyond column filters that allow you to meet complex security and compliance requirements across diverse data sources. For more information about managing permissions on catalogs, see Adding existing databases and catalogs using AWS Lake Formation permissions in the *Amazon SageMaker Unified Studio User Guide* and Managing Lake Formation Permissions in the *AWS Lake Formation Developer Guide*.

By implementing these fine-grained access controls, you can ensure that users only access data they're authorized to see, maintaining compliance with your organization's security policies. This creates a consistent security model across your data sources. Now, you have successfully set up fine-grained access permissions on your DynamoDB federated catalog.

Step 3: Validate fine-grained access permissions on federated catalogs

After you set up federated catalogs with fine-grained access permissions in Step 2, run queries to confirm access permissions are working as expected.

To validate fine-grained access permissions on federated catalogs

1. On the SageMaker Unified Studio console, for the domain you created in the prerequisite, choose **Open unified studio**.
2. Choose your data analyst project name under **Your projects**.
3. From the **Build** menu, choose **Query Editor**.
4. In the **Data explorer**, expand **Lakehouse**, choose the DynamoDB catalog that you created in Step 1.
5. Drill down to the table that you set up fine-grained access permissions in Step 2, and choose **Query with Athena** to run a sample query.

   For example, run the following query. Replace *`your_federated_catalog_name`* with the name of your catalog, *`default`* with the name of your database, and *`your_table_name`* with the name of your DynamoDB table. To learn more, see SQL analytics in the *Amazon SageMaker Unified Studio User Guide*.

```
select * from your_federated_catalog_name.default.your_table_name limit 10;
```

Note how permissions are working as expected because the query result doesn't include the mobile phone number column that was visible in the admin project view.

To have other users under the data analyst persona get the fine-grained access permissions

1. Create data analyst SSO users or groups. For more information about how to add an SSO user to your domain, see Managing users in Amazon SageMaker Unified Studio in the *Amazon SageMaker Unified Studio Administrator Guide*.
2. Add these SSO users to your SageMaker Unified Studio domain. For more information about how to add an SSO user to your domain, see Managing users in Amazon SageMaker Unified Studio in the *Amazon SageMaker Unified Studio Administrator Guide*.
3. Add these users as members (**"Contributor"**) to your SageMaker Unified Studio data analyst project. The data analyst users can have access to this data analyst project and will only have access to a subset of data that's defined by the data lake administrator in Step 2. For more information about how to add members to your projects, see Add project members in the *Amazon SageMaker Unified Studio User Guide*.

Step 4: Clean up

Make sure you remove the SageMaker Lakehouse resources to mitigate any unexpected costs. Delete the following resources:

- The connections and catalogs that you created in Step 1.

  Specifically, choose your project from SageMaker Unified Studio. Choose **Data** in the navigation pane. Choose the SageMaker Lakehouse catalog that you created in Step 1. Choose the **Actions** menu and choose **Remove**. Type "**Confirm**" and choose **Remove connection**.
- The underlying DynamoDB data sources that you created in the prerequisites. For more information about deleting a DynamoDB table, see Delete your DynamoDB table to clean up resources in the *Amazon DynamoDB Developer Guide*.

- The SageMaker Unified Studio admin and data analyst projects that you created in the prerequisites. For more information about deleting projects, see Delete a project in the Amazon SageMaker Unified Studio User Guide.
- The SageMaker Unified Studio domain that you created in the prerequisites.

Next steps

Now that you've successfully set up SageMaker Lakehouse integrated access controls for Athena federated queries, consider these next steps to further enhance your data governance and analytics capabilities:

- **Expand your data sources** - Connect supported data sources such as PostgreSQL, MySQL, or Amazon DocumentDB, to create a unified data ecosystem with consistent access controls.
- **Implement advanced security patterns** - Explore row-level security, cell-level filtering, and attribute-based access control to meet complex compliance requirements across your organization. For more information, see Managing Lake Formation Permissions in the *AWS Lake Formation Developer Guide*.
- **Build analytics workflows** - Create end-to-end analytics pipelines that leverage federated queries for data preparation and ML model training.
- **Integrate with visualization tools** - Connect Amazon QuickSight to your federated catalogs to create dashboards and visualizations with the same security controls.
- **Automate governance processes** - Use the Amazon Athena REST API (CreateDataCatalog), AWS CloudFormation (`AWS::Athena::DataCatalog`) or the AWS CDK (CfnDataCatalog) to automate the creation and management of federated connections and access controls. After creating a data catalog, you need to create a data source connection and register your connection as a Glue Data Catalog.

This integration between SageMaker Lakehouse and Athena federated queries provides significant benefits for organizations with diverse data ecosystems. Data scientists can now analyze customer behavior by combining transaction data from PostgreSQL with clickstream data in Amazon S3. Financial analysts can query historical market data alongside real-time trading information without complex ETL processes. Healthcare researchers can analyze patient records stored in different systems while maintaining compliance with privacy regulations.

For more information about federated queries in Athena and the data sources that support fine-grained access controls, see Register your connection as a Glue Data Catalog in the *Athena User*

Guide. For more information about extending your SageMaker Lakehouse environment, see Add Data to SageMaker Lakehouse and Publishing Data in the *Amazon SageMaker Unified Studio User Guide*. For more information about specific use cases and implementation examples, see Simplify data access for your enterprise using SageMaker Lakehouse, Simplify analytics and AI/ML with new SageMaker Lakehouse, and Catalog and govern Amazon Athena federated queries with SageMaker Lakehouse in the *AWS Blog posts*.

Get started fine-tuning foundation models in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio provides a large collection of state-of-the-art foundation models. These models support use cases such as content writing, code generation, question answering, copywriting, summarization, classification, information retrieval, and more. You can find, customize, and deploy these foundation models in the JumpStart model catalog. You can use the foundation models to build your own generative AI solutions for a wide range of applications.

A foundation model is a large pre-trained model that is adaptable to many downstream tasks and often serves as the starting point for developing more specialized models. Examples of foundation models include Meta Llama 4 Maverick 17B, DeepSeek-R1, or Stable Diffusion 3.5 Large. These models are pre-trained on massive amounts of data.

Model customization

You might need to customize a base foundation model to better align it with your specific use cases. The recommended way to first customize a foundation model is through prompt engineering. Providing your foundation model with well-engineered, context-rich prompts can help achieve desired results without any fine-tuning or changing of model weights. For more information, see Prompt engineering for foundation models in the *Amazon SageMaker AI Developer Guide*.

If prompt engineering alone is not enough to customize your foundation model to a specific task, you can fine-tune a foundation model on additional domain-specific data. The fine-tuning process involves changing model weights.

To help you learn how to fine-tune foundation models, Amazon SageMaker Unified Studio provides an example training dataset for each model that's eligible for training. You can also choose to fine-tune the model with your own data set. Before you can do that, you must prepare your data set and store it in an Amazon S3 bucket. The required format for the data set varies between models. You can learn about the required format in the model details page in Amazon SageMaker Unified Studio.

Fine-tuning a foundation model

One way to fine-tune a model in Amazon SageMaker Unified Studio is to use JumpStart. First, you choose a foundation model from the catalog. Then, you train the model with a training data set. Follow these steps to learn how to fine-tune with this approach.

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. Choose a model to train.

    a. From the main menu, choose **Build**.

    b. From the drop-down menu, choose **Jumpstart Models**.

    The JumpStart page lists the model providers.

    c. Choose a model provider. The page displays the models for that provider.

    d. Under **Action**, choose **Trainable**. The page displays the trainable models for that provider.

    e. From the provider's list of models, choose the model you want to train.

    Amazon SageMaker Unified Studio shows the model details page, which provides information from the model provider. If you want to prepare a custom fine-tuning data set, use this page to learn the required format.
3. From the model details page, choose **Train** to create a training job.
4. On the **Fine-tune model** page, under **Data**, do one of the following:

    a. Keep the default selection of **Example training dataset**. This data set is useful when you want to learn how to fine-tune with Amazon SageMaker Unified Studio. However, it won't be effective for customizing the model for your specific needs.

    b. If you've prepared a custom training data set, choose **Enter training dataset**, and provide the URI that locates it in Amazon S3.
5. (Optional) Under **Hyperparameters**, update the hyperparameters you want to change.

    The hyperparameters available for each fine-tunable model differ depending on the model. Review the help text and additional information in the model details pages in Amazon SageMaker Unified Studio to learn more about hyperparameters specific to the model of your choice.

    For more information on available hyperparameters, see Commonly supported fine-tuning hyperparameters in the *Amazon SageMaker AI Developer Guide*.

6. Under **Deployment**, for **Training Instance**, specify the training instance type for your training job. You can only choose from instances that are compatible with the model that you chose.

   For **Output artifact location (S3 URI)**, specify where Amazon SageMaker uploads the fine-tuned model. You can choose to use the default bucket, or you can specify a custom location in Amazon S3.
7. (Optional) Under **Additional Information**, for **Training Job Name**, you can edit the default name.
8. (Optional) For **Tags**, you can add and remove tags in the form of key-value pairs to help organize and categorize your fine-tuning training jobs.
9. Enter **Submit** to submit the training job. You can view the training job from the **Training jobs** page.

Document history for the Amazon SageMaker User Guide

The following table describes the documentation releases for Amazon SageMaker.

Change	Description	Date
Initial release	Initial release of the Amazon SageMaker User Guide	June 13, 2025