Step 1: Create and authorize a service principal

7AI connects to Databricks as a service principal using OAuth machine-to-machine (M2M) credentials. Create a dedicated service principal so its access can be scoped to only the data 7AI should query.

The steps below use the account console. A workspace admin can instead create and assign the service principal from the workspace's Settings > Identity and access page.

ADD A SERVICE PRINCIPAL TO YOUR ACCOUNT

[01]	As an account admin, log in to the Databricks account console.
[02]	In the sidebar, click User management.
[03]	On the Service principals tab, click Add service principal.
[04]	Enter a name for the service principal.
[05]	Click Add.
[06]	Open the service principal and copy its Application ID. This is the client ID used by 7AI and in the SQL grants in Step 2.

ASSIGN THE SERVICE PRINCIPAL TO A WORKSPACE

[01]	In the account console sidebar, click Workspaces.
[02]	Click the name of the workspace that hosts your SQL warehouse.
[03]	On the Permissions tab, click Add permissions.
[04]	Search for and select the service principal.
[05]	Assign the Databricks SQL access workspace entitlement, then click Save.

GRANT ACCESS TO THE SQL WAREHOUSE

[01]	Open the target Databricks workspace. In the workspace sidebar, select SQL Warehouses, then select the warehouse 7AI will query.
[02]	On the Permissions tab, add the service principal with CAN USE.
[03]	Open the warehouse's Connection details tab and copy the SQL Warehouse ID — the value after `/sql/1.0/warehouses/` in the HTTP path. Save it; you'll need it when configuring 7AI in Step 4.

Step 2: Grant read-only data access

Grant the service principal access only to the Unity Catalog objects that 7AI should query. For read-only queries, the service principal needs three privileges:

- USE CATALOG on the catalog
- USE SCHEMA on the schema
- SELECT on the tables or views

Use table-level grants where possible to preserve least privilege. Replace the placeholder names with your catalog, schema, and table. `<client-id>` is the service principal's Application ID copied in Step 1 — the same value Databricks uses to identify the principal in SQL grants and as its OAuth client ID.

GRANT ACCESS TO A SINGLE TABLE:

```
GRANT USE CATALOG ON CATALOG `<catalog-name>` TO `<client-id>`;
GRANT USE SCHEMA ON SCHEMA `<catalog-name>`.`<schema-name>`
  TO `<client-id>`;
GRANT SELECT ON TABLE `<catalog-name>`.`<schema-name>`.`<table-name>`
  TO `<client-id>`;
```

GRANT READ ACCESS TO ALL CURRENT AND FUTURE TABLES AND VIEWS IN A SCHEMA:

```
GRANT USE CATALOG ON CATALOG `<catalog-name>` TO `<client-id>`;
GRANT USE SCHEMA ON SCHEMA `<catalog-name>`.`<schema-name>`
  TO `<client-id>`;
GRANT SELECT ON SCHEMA `<catalog-name>`.`<schema-name>`
  TO `<client-id>`;
```

Step 3: Generate an OAuth secret

Generate an OAuth secret for the service principal. In the next step, you'll enter this secret together with the client ID copied in Step 1.

[01]	Click your username in the top bar and select Settings.
[02]	Click the Identity and access tab.
[03]	Next to Service principals, click Manage.
[04]	Select the service principal.
[05]	Click the Secrets tab.
[06]	Click Generate secret.
[07]	Set the secret's lifetime in days according to your policies.
[08]	Click Generate.
[09]	Copy the displayed secret and client ID, then click Done.

Databricks displays the secret only once. Store it securely — you'll need it in the next step.

Step 4: Connect Databricks to 7AI

Now that you have the service principal's client ID and OAuth secret, connect Databricks to 7AI:

[01] From the 7AI dashboard, select Configuration at the top of the screen.

[02] Navigate to the Connector Directory in the configuration page.

[03] Find and select Databricks from the list of available connectors.

[04] Enter the required configuration:

- `WORKSPACE URL`: Your workspace URL, for example `https://dbc-xxxx.cloud.databricks.com`
- `SQL WAREHOUSE ID`: The SQL Warehouse ID you copied in Step 1
- `SERVICE PRINCIPAL CLIENT ID`: The service principal's client ID (its application ID)
- `SERVICE PRINCIPAL CLIENT SECRET`: The OAuth secret generated in Step 3

[05] Click Save to create the connection.

[06] Verify the connector installation:

- The connector should appear in your Active Connectors list.
- Wait approximately 5 minutes for the first health check to run automatically.
- Check that the status shows as Operational or Healthy.