

Claude Code + Databricks MCP Setup Guide

Get Claude Code investigating your `(novatech)` bugs in under 30 minutes.

Your Options

Option	Best For	Setup Time
A. Databricks Managed MCP	Production, in-workspace	15 min
B. Community MCP Server	Claude Code local dev	30 min
C. Direct SDK (no MCP)	Simplest approach	20 min

I recommend **Option B** for your demo since you want Claude Code investigating locally.

Prerequisites

- Databricks workspace with `(novatech)` catalog created (you've done this)
 - Claude Code installed
 - Databricks Personal Access Token
 - Node.js 18+ (for MCP servers)
 - Python 3.10+ (for community MCP server)
-

Step 1: Install Claude Code

```
bash
```

```
# Install Claude Code globally  
npm install -g @anthropic-ai/claude-code
```

```
# Verify installation  
claude --version
```

```
# Authenticate (opens browser)  
claude auth login
```

Step 2: Get Your Databricks Credentials

You need two things from your Databricks workspace:

2a. Workspace URL

```
https://<your-workspace>.cloud.databricks.com
```

2b. Personal Access Token

1. Go to Databricks → User Settings → Developer → Access Tokens
2. Click "Generate New Token"
3. Name: `claude-code-mcp`
4. Lifetime: 90 days
5. Copy the token (you won't see it again)

2c. SQL Warehouse ID

1. Go to SQL Warehouses in Databricks
2. Click on your warehouse
3. Copy the ID from the URL: `/sql/warehouses/<THIS-IS-YOUR-ID>`

Step 3: Install Community Databricks MCP Server

The best community MCP server for your use case is **RafaelCartenet's mcp-databricks-server** - it has Unity

Catalog metadata, lineage, and SQL execution.

```
bash

# Create a directory for MCP servers
mkdir -p ~/mcp-servers
cd ~/mcp-servers

# Clone the Databricks MCP server
git clone https://github.com/RafaelCartenet/mcp-databricks-server.git
cd mcp-databricks-server

# Install with uv (recommended) or pip
pip install uv
uv venv
source .venv/bin/activate # On Windows: .venv\Scripts\activate
uv pip install -e .

# Or with pip
pip install -e .
```

What this MCP server provides:

Tool	What It Does
list_catalogs	List all Unity Catalogs
list_schemas	List schemas in a catalog
list_tables	List tables in a schema
describe_table	Get table schema, columns, comments
get_table_lineage	Get upstream/downstream tables
execute_sql_query	Run SQL on your warehouse

Step 3b: Set Environment Variables

```
bash
```

```
# Add to your ~/.bashrc or ~/.zshrc
export DATABRICKS_HOST="https://YOUR-WORKSPACE.cloud.databricks.com"
export DATABRICKS_TOKEN="YOUR-PERSONAL-ACCESS-TOKEN"
export DATABRICKS_SQL_WAREHOUSE_ID="YOUR-SQL-WAREHOUSE-ID"

# Reload
source ~/.bashrc
```

Test the connection:

```
bash

cd ~/mcp-servers/mcp-databricks-server
python -c "from databricks.sdk import WorkspaceClient; w = WorkspaceClient(); print(w.current_user.me())"
```

You should see your user info if the connection works.

Step 4: Configure Claude Code MCP Settings

Claude Code uses a config file to know which MCP servers to connect to.

Find your config file location:

```
bash

# macOS
~/Library/Application Support/Claude/clause_desktop_config.json

# Linux
~/.config/Claude/clause_desktop_config.json

# Windows
%APPDATA%\Claude\clause_desktop_config.json
```

Create/edit the config:

```
json
```

```
{
  "mcpServers": {
    "databricks": {
      "command": "python",
      "args": ["-m", "mcp_databricks_server"],
      "cwd": "/Users/YOUR_USERNAME/mcp-servers/mcp-databricks-server",
      "env": {
        "DATABRICKS_HOST": "https://YOUR-WORKSPACE.cloud.databricks.com",
        "DATABRICKS_TOKEN": "YOUR-PERSONAL-ACCESS-TOKEN",
        "DATABRICKS_SQL_WAREHOUSE_ID": "YOUR-SQL-WAREHOUSE-ID"
      }
    }
  }
}
```

Replace:

- **YOUR_USERNAME** → your system username
- **YOUR-WORKSPACE** → your Databricks workspace (e.g., `adb-1234567890.12.azuredatabricks.net`)
- **YOUR-PERSONAL-ACCESS-TOKEN** → token from Step 2b
- **YOUR-SQL-WAREHOUSE-ID** → warehouse ID from Step 2c

Alternative: Use uvx (simpler)

If you installed with uv, you can use:

```
json
{
  "mcpServers": {
    "databricks": {
      "command": "uvx",
      "args": ["--from", "git+https://github.com/RafaelCartenet/mcp-databricks-server", "mcp-databricks-server"],
      "env": {
        "DATABRICKS_HOST": "https://YOUR-WORKSPACE.cloud.databricks.com",
        "DATABRICKS_TOKEN": "YOUR-PERSONAL-ACCESS-TOKEN",
        "DATABRICKS_SQL_WAREHOUSE_ID": "YOUR-SQL-WAREHOUSE-ID"
      }
    }
  }
}
```

Alternative: Use Databricks Managed MCP (In-Workspace)

Databricks now has **managed MCP servers** built into the platform. If you want to use these from AI Playground or Agent Bricks (not Claude Code), here's how:

Available Managed MCP Servers:

Server	URL Pattern	What It Does
DBSQL	{host}/api/2.0/mcp/dbsql	Execute SQL queries
Vector Search	{host}/api/2.0/mcp/vector-search/{catalog}/{schema}	Query vector indexes
UC Functions	{host}/api/2.0/mcp/functions/{catalog}/{schema}	Run UC functions
Genie	{host}/api/2.0/mcp/genie/{space_id}	Query Genie spaces

To Use in AI Playground:

1. Go to **AI Playground** in your workspace
2. Select a model with **Tools enabled**
3. Click **Tools → + Add tool → MCP Servers**
4. Select a managed server (DBSQL, Vector Search, etc.)
5. Start chatting!

This is great for quick testing but doesn't give you the Claude Code terminal experience.

Step 5: Create a Project Directory

```
bash
```

```
# Create project folder
mkdir ~/datascope-debug
cd ~/datascope-debug

# Initialize for Claude Code
claude init

# Create a context file for Claude
cat > CLAUDE.md << 'EOF'
# DataScope Data Debugger
```

You are investigating data quality issues in the NovaTech B2B SaaS data platform.

Environment

- **Catalog**: `novatech`
- **Schemas**: `bronze`, `silver`, `gold`
- **Warehouse**: Connected via MCP

Available Tables

Bronze (Raw)

- `novatech.bronze.salesforce_accounts_raw`
- `novatech.bronze.stripe_payments_raw`
- `novatech.bronze.product_events_raw`
- `novatech.bronze.zendesk_tickets_raw`

Silver (Cleaned)

- `novatech.silver.dim_customers`
- `novatech.silver.fct_subscriptions`
- `novatech.silver.fct_payments`
- `novatech.silver.fct_product_usage`

Gold (Business Metrics)

- `novatech.gold.arr_by_customer`
- `novatech.gold.churn_predictions`
- `novatech.gold.customer_health_scores`
- `novatech.gold.payment_status_summary`
- `novatech.gold.revenue_recognition`

Investigation Approach

When investigating data issues:

1. **Quantify** - How many records affected? What percentage?
2. **Trace lineage** - Where does this data come from?
3. **Compare layers** - Do bronze/silver/gold match?
4. **Check transformations** - What SQL creates this table?
5. **Find root cause** - WHERE clause? JOIN? NULL handling?

Known Bug Patterns

- Timezone mismatches (PST vs UTC)
- Missing ELSE in CASE statements
- Duplicate records not deduplicated
- Addon products excluded from aggregations
- JOIN fanout (1:N creating extra rows)
- Late-arriving data (payment_date vs processed_at)

EOF

Step 6: Upload Your Transformation SQL

Put your transformation SQL files in the project so Claude can search them:

```
bash
```

```
# Create sql folder
mkdir -p ~/datascope-debug/sql/gold

# Copy your transformation files (or create them)
cat > ~/datascope-debug/sql/gold/churn_predictions.sql << 'EOF'
-- Transformation: Churn Predictions
-- Owner: datascience@novatech.com
```

```
CREATE OR REPLACE TABLE novatech.gold.churn_predictions AS
```

```
WITH recent_activity AS (
    SELECT
        customer_id,
        MAX(usage_timestamp) as last_activity,
        AVG(logins) as avg_logins
    FROM novatech.silver.fct_product_usage
    WHERE usage_timestamp > current_timestamp() - INTERVAL 30 DAYS
    GROUP BY customer_id
)
```

```
SELECT
    c.customer_id,
    c.company_name,
    c.segment,
    ra.last_activity,
    ra.avg_logins,
```

```
-- BUG: Missing ELSE clause!
CASE
    WHEN ra.avg_logins > 20 THEN 'Low Risk'
    WHEN ra.avg_logins > 5 THEN 'Medium Risk'
    WHEN ra.avg_logins <= 5 THEN 'High Risk'
    -- Missing: ELSE 'High Risk' for NULL cases
END as churn_risk
```

```
FROM novatech.silver.dim_customers c
LEFT JOIN recent_activity ra ON c.customer_id = ra.customer_id;
EOF
```

```
cat > ~/datascope-debug/sql/gold/arr_by_customer.sql << 'EOF'
-- Transformation: ARR by Customer
-- Owner: finance@novatech.com
```

```
CREATE OR REPLACE TABLE novatech.gold.arr_by_customer AS
```

```
SELECT
    customer_id,
    SUM(mrr) * 12 as arr,
    COUNT(*) as subscription_count
FROM novatech.silver.fct_subscriptions
WHERE
    status = 'active'
    -- BUG: Only Core Platform, excludes addons!
    AND product_type = 'Core Platform'
GROUP BY customer_id;
EOF
```

Step 7: Start Claude Code

```
bash
cd ~/datascope-debug

# Start Claude Code in interactive mode
claude

# Or start with a specific question
claude "Why do some customers have NULL churn_risk in the novatech.gold.churn_predictions table?"
```

Step 8: Test the Connection

Once Claude Code starts, test MCP is working:

You: List the tables in the novatech.gold schema

Claude: [Uses Databricks MCP to query Unity Catalog]

I can see the following tables in novatech.gold:

- arr_by_customer
- churn_predictions
- customer_health_scores
- payment_status_summary
- revenue_recognition

Step 9: Run Your First Investigation

Try these questions:

Question 1: NULL churn_risk (Easy)

Why do some customers have NULL churn_risk in novatech.gold.churn_predictions?

Expected Claude behavior:

1. Query the table to count NULLs
2. Check what other columns are NULL for those records
3. Look at the transformation SQL
4. Find the missing ELSE clause

Question 2: ARR Discrepancy (Easy)

Why does novatech.gold.arr_by_customer show lower ARR than if I sum all active subscriptions in silver.fct_subscriptions?

Question 3: Revenue Inflation (Medium)

Why might revenue in novatech.gold.revenue_recognition be higher than expected compared to source payment data?

Question 4: Churn False Positive (Medium)

Why would a customer be marked as high churn risk in novatech.gold.churn_predictions when they have recent activity?

Troubleshooting

MCP not connecting

```
bash

# Check if MCP server runs standalone
npx -y @anthropic-ai/mcp-server-databricks

# Should output: "MCP server running..."
# If error, check your env vars
```

Token issues

```
bash

# Test your token works
curl -X GET "https://YOUR-WORKSPACE.cloud.databricks.com/api/2.0/clusters/list" \
-H "Authorization: Bearer YOUR-TOKEN"
```

Warehouse not responding

- Make sure your SQL Warehouse is **running** (not stopped)
- Check warehouse ID is correct
- Verify your token has SQL access

Recording a Demo

Once working, record yourself:

1. **Open terminal** with Claude Code
2. **Ask a question:** "Why do some customers have NULL churn_risk?"
3. **Watch Claude** query tables, check lineage, find the bug

4. Show the fix it recommends

This becomes your portfolio video for Sierra interview!

Next: Add GitHub MCP (Optional)

If your transformation code is in a GitHub repo:

```
json
{
  "mcpServers": {
    "github": {
      "command": "npx",
      "args": ["-y", "@modelcontextprotocol/server-github"],
      "env": {
        "GITHUB_TOKEN": "ghp_your_github_token"
      }
    }
  }
}
```

Then Claude can search your actual codebase for transformations.

Summary

Step	Action	Time
1	Install Claude Code	2 min
2	Get Databricks credentials (token + warehouse ID)	5 min
3	Clone & install community MCP server	10 min
4	Configure MCP in claude_desktop_config.json	5 min
5	Create project with CLAUDE.md context	5 min
6	Add SQL transformation files	5 min

Step	Action	Time
7-9	Test and investigate	10 min

Total: ~40 minutes to first investigation

Quick Reference: Your Credentials

Fill these in and keep handy:

```
DATABRICKS_HOST=https://_____cloud.databricks.com
DATABRICKS_TOKEN=dapi_____
DATABRICKS_SQL_WAREHOUSE_ID=_____
```

What Success Looks Like

You: Why do some customers have NULL churn_risk?

Claude: I'll investigate this data quality issue.

[Calling tool: execute_sql_query]

```
SELECT COUNT(*) as total,
       SUM(CASE WHEN churn_risk IS NULL THEN 1 ELSE 0 END) as null_count
FROM novatech.gold.churn_predictions
```

Result: total=500, null_count=19 (3.8%)

[Calling tool: execute_sql_query]

```
SELECT customer_id, avg_logins, churn_risk
FROM novatech.gold.churn_predictions
WHERE churn_risk IS NULL LIMIT 5
```

Result: All have NULL avg_logins too!

[Reading file: sql/gold/churn_predictions.sql]

Found the bug! The CASE statement on line 47 is missing an ELSE clause:

CASE

```
WHEN ra.avg_logins > 20 THEN 'Low Risk'  
WHEN ra.avg_logins > 5 THEN 'Medium Risk'  
WHEN ra.avg_logins <= 5 THEN 'High Risk'  
-- Missing ELSE here!  
END as churn_risk
```

When avg_logins is NULL (customers with no activity in 30 days),
none of the conditions match, so churn_risk becomes NULL.

Fix: Add `ELSE 'High Risk'` to classify inactive customers.

That's exactly what Asana demonstrated!