

# Azure Databricks: Building a Chatbot Companion using AI/BI Genie

Wilson Mok

Jun 14, 2025



## Wilson Mok

Sr. Data architect & Consultant

 /wilson-mok

 /wilson-mok

 @the-analytics-lab



My experiences includes:

- Avanade
- CAE
- Air Canada



# Expected Audience and Presentation Structure



## What We'll Cover:

- What is Genie and how it helps users get answers from data using natural language.
- How to prepare your data for Genie to understand the business context.
- How to guide Genie's responses using custom instructions, examples, and trusted assets.
- How to monitor results and make Genie smarter over time.



## Fundamental knowledge: Basic SQL and Databricks.

- Experienced with AI/BI Genie would be beneficial but not required.

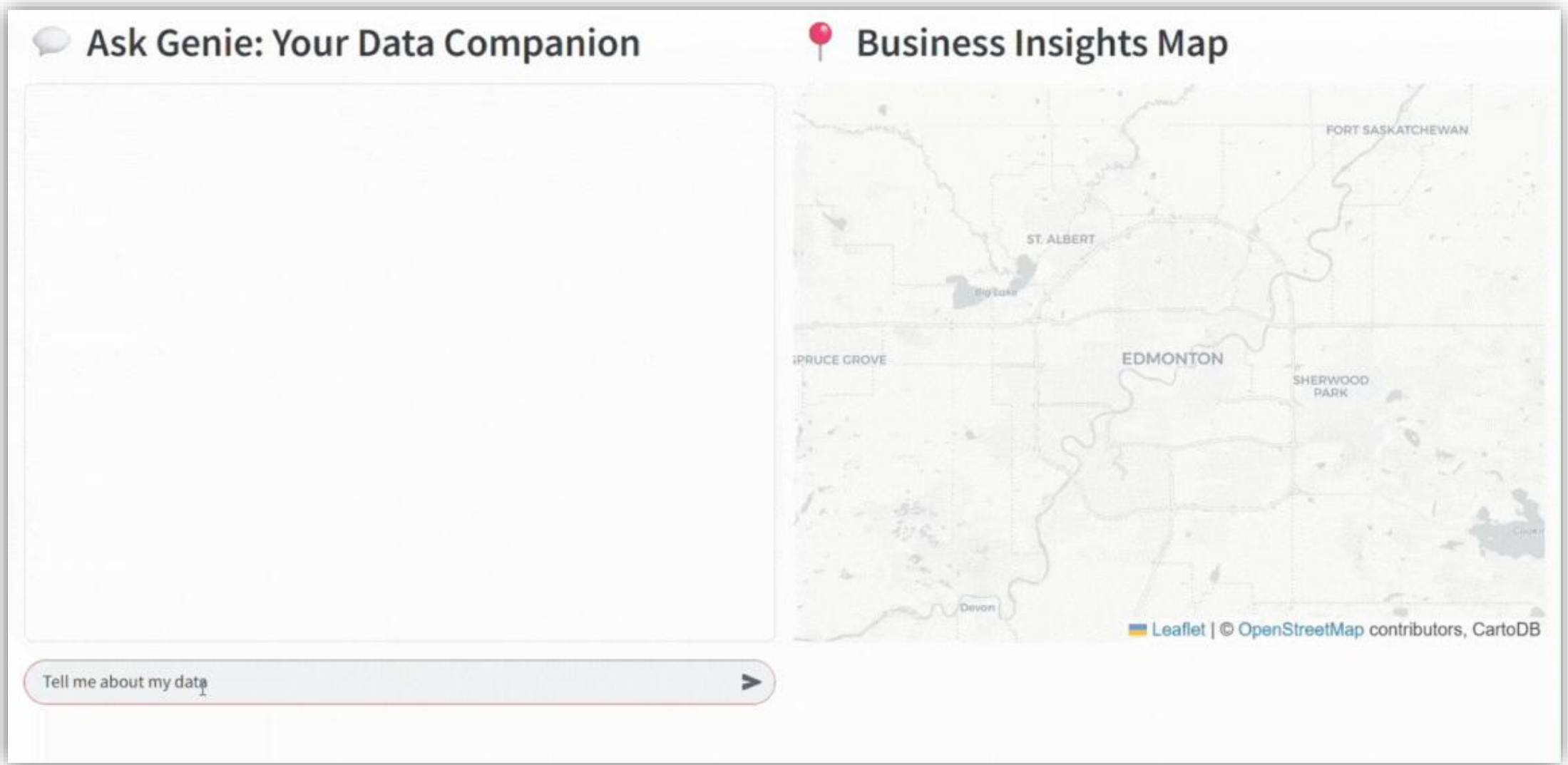
# The Reality of Self-Service Analytics



Image Credit: <https://imgflip.com/l/9wxm75>



# What if you could talk to your data like this?



# Meet Genie: Your Conversational Data Companion

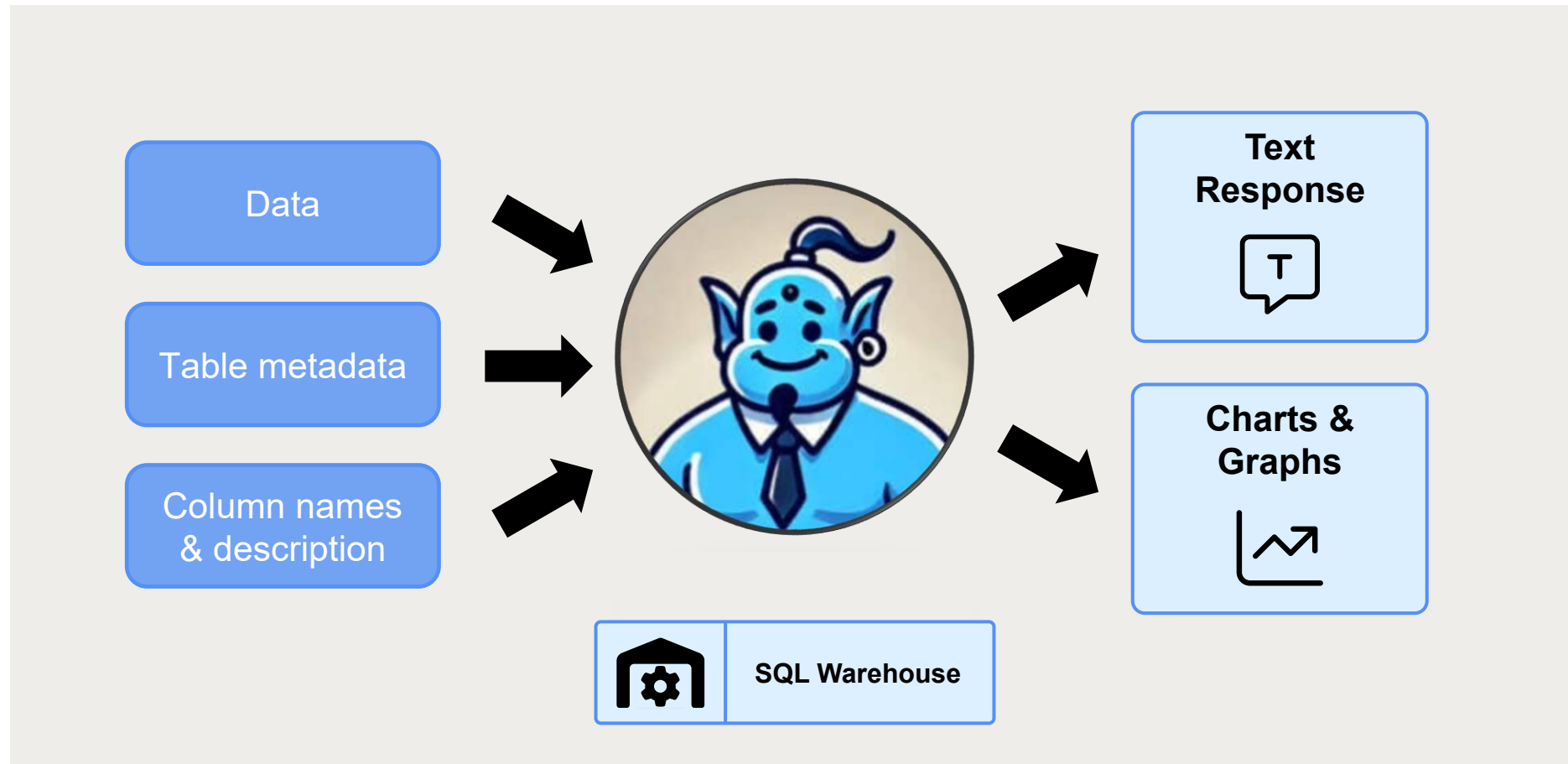


Image Credit: <https://www.databricks.com/blog/onboarding-your-new-ai-bi-genie>

# Now... How do we get there?

## Genie Lifecycle

### 1. Define & Prepare

Lay the foundation with business-aligned data

- Define the purpose and audience
- Select the right tables
- Improve Metadata Quality



### 2. Configure

Tailor Genie's logic, context and trusted assets

- Review Value Dictionaries
- Add Custom Instructions
- Create Trusted Assets



### 3. Validate

Quality control and validate results

- Improve accuracy through Benchmark testing.
- Validate results to measure accuracy.



### 4. Monitor & Refine

Monitor user responses and apply feedback

- Monitor user feedback
- Continuous Improvement through refining configurations.



# Define & Prepare: Laying the Foundation

## Metadata: Help Genie Understand How Your Data Connects

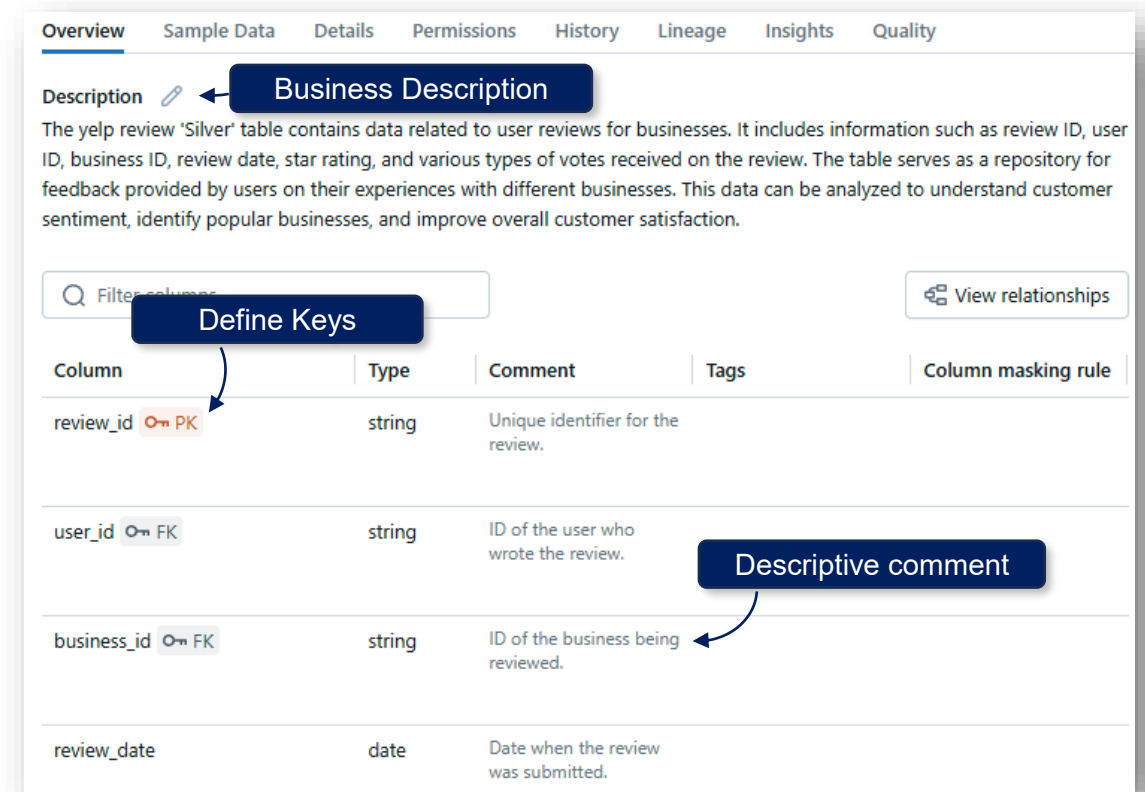


**Purpose:** Improve Genie's ability to generate accurate queries by clarifying data relationships and meaning through business-friendly metadata.

**How:** Leverage table description, column naming standards, column comments and table relationships (PK/FK) hints.

### Tips

- ✓ Use descriptive, consistent column names (e.g. user\_id, product\_id, review\_date).
- ✓ Add clear, business-friendly column comments. (e.g. Date when the review was submitted)
- ✓ Define PK/FK relationships. They serve as hints/guidance and not enforcement.



The screenshot shows a data catalog interface with tabs: Overview, Sample Data, Details, Permissions, History, Lineage, Insights, and Quality. The 'Overview' tab is active, displaying a 'Description' section with a 'Business Description' callout. Below the description is a 'Filter columns' search bar and a 'View relationships' button. A table lists columns with their types and comments. A 'Define Keys' callout points to the 'review\_id' column, which is marked as a Primary Key (PK). A 'Descriptive comment' callout points to the comment for the 'business\_id' column.

Column	Type	Comment	Tags	Column masking rule
review_id <span>PK</span>	string	Unique identifier for the review.		
user_id <span>FK</span>	string	ID of the user who wrote the review.		
business_id <span>FK</span>	string	ID of the business being reviewed.		
review_date	date	Date when the review was submitted.		



# Configure: Tailor Genie's logic, context and trusted assets

## Value Dictionaries: Help Genie Interpret Business Terms Accurately



**Purpose:** Map domain-specific terms to specific column values in string fields so Genie can understand (e.g. customer\_segment).

**How:** Genie automatically creates and manages list of values for eligible string columns.

### Tips

- ✓ Enable dictionaries only for meaningful, well-maintained values. Refresh them regularly to keep value sets up to date.
  - ⚠ Dictionaries are managed by Genie. It cannot be edited.
- ✓ For non-string columns, use Custom Instructions.
- ✓ Stay within limit of value dictionaries: Public Preview
  - 30 columns/Genie, 1024 distinct values, 127 max char.

Columns (14)			
Name	Description		
business_id A <sup>B</sup> <sub>C</sub> STRING	On PK	Unique business id...	Add value dictionary ⋮
name A <sup>B</sup> <sub>C</sub> STRING		Name of the busin...	Value dictionary added ⋮
city A <sup>B</sup> <sub>C</sub> STRING		City where the busi...	Value dictionary added ⋮
state A <sup>B</sup> <sub>C</sub> STRING		State where the bu...	Value dictionary added ⋮
latitude 1.2 DOUBLE		Latitude coordinate...	Add value dictionary ⋮

# Configure: Tailor Genie's logic, context and trusted assets

## Custom Instructions: Guide Genie's Behaviour with Context, and Tone



**Purpose:** Shape Genie's response by defining how it should behave, what to prioritize, how to response and what to avoid.

**How:** Use Markdown to define Genie's behavior in a structured way.

### Tips

- ✓ Organize using Markdown formatting: Heading, bullet lists.
- ✓ Be specific and concise – less is more when communicating intent (e.g. the fiscal year starts on May).
- ✓ Tailor language and expectations to your audience.
- ✓ Define any common terms and include reporting standards.

Context
Settings

Data
Instructions
SQL Queries

### General Instructions

Add general instructions on how you want Genie to behave.

Examples:

- \* MCA stands for My Company Abbreviation
- \* Countries in the country\_code column are stored with two characters (e.g. US, IT)
- \* The rating column is for parental guidance and not popularity
- \* When a user asks for performance they are interested in a products revenue

(You can use markdown text)

# Configure: Tailor Genie's logic, context and trusted assets

## Trusted Assets: Prioritize Curated Logic to Deliver Accurate Response

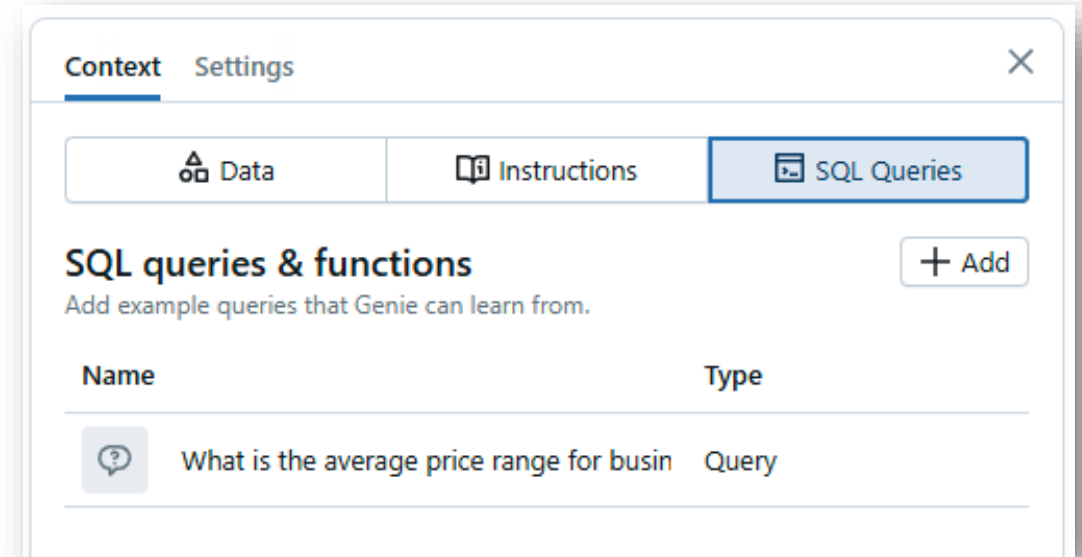


**Purpose:** Ensure Genie uses vetted queries and functions to produce responses users can rely on.

**How:** Add “Trusted” SQL Queries and Functions with parameters for Genie to prioritize when answering questions.

### Tips

- ✓ Trust only production-ready, business-approved assets.
- ✓ Use parameters for repeatable logic.
- ✓ Regularly review Trusted assets to ensure they are valid and relevant.



The screenshot shows a settings window titled 'Context' with a close button (X) in the top right corner. Below the title bar are two tabs: 'Context' (selected) and 'Settings'. Under the 'Context' tab, there are three sub-tabs: 'Data', 'Instructions', and 'SQL Queries' (selected). The 'SQL Queries' sub-tab contains the heading 'SQL queries & functions' and a sub-heading 'Add example queries that Genie can learn from.' To the right of this heading is a '+ Add' button. Below this is a table with two columns: 'Name' and 'Type'. The table contains one entry: a question mark icon in the 'Name' column and the text 'What is the average price range for busin' followed by 'Query' in the 'Type' column.

Name	Type
What is the average price range for busin	Query

# Validate: Quality Control and Validate Results

## Benchmarking: Measure Genie's response accuracy



**Purpose:** Test the overall response accuracy using a pre-defined set of test questions and answers.

**How:** Create realistic questions the corresponding expected SQL query. Genie will compare its response with the provided query output

### Tips

- ✓ Phrase each question 3 to 5 different ways to simulate realistic language variability from users.
- ✓ Include questions known to be ambiguous or frequently misunderstood.
- ✓ Repeat benchmarks regularly to monitor performance over time.

Evaluations Questions (2)	
Question	SQL Answer
What is the max, min and avg stars for	<pre>SELECT   max(stars) as max_stars,   min(stars) as min_stars,   avg(stars) as avg_stars FROM   `demo_catalog`.`demo_yelp_academic`.`silver_business` ... 3 more lines</pre>
What is the top 5 reviews for the best	<pre>WITH top_5_reviews AS (   SELECT     b.name,     r.text   FROM     `demo_catalog`.`demo_yelp_academic`.`silver_business` ... 16 more lines</pre>

# Monitor & Refine: Monitor User Responses and Apply Feedback

## Feedback & Monitoring: Capturing and Actioning on User's Review



**Purpose:** Creates a continuous improvement loop by tracking user's feedback to improve performance.

**How:** Users provide feedback on responses. These signals are tracked and monitored to help identify areas where Genie refinements are required.

### Tips

- ✓ Encourage users to give feedback (Yes, Fix it, or Request Review). They are crucial in helping improve the Genie.
- ✓ Establish a review process to "Close the loop".

What is the average stars for businesses in Sherwood Park, Alberta?

This query calculates the average star rating of businesses located in Sherwood Park, Alberta. It focuses on the ratings provided by users to assess the overall quality of businesses in that specific area. `silver_business`

1 row [Download](#) [Add as instruction](#) [Show code](#) [More](#)

	1.2 average_stars
1	3.5

Is this correct? Your response makes Genie better over time. [Yes](#) [Fix it](#) [Request review](#)

# Let's Build One – Configuring Genie in Azure Databricks



## Genie Lifecycle

### 1. Define & Prepare

Lay the foundation with business-aligned data

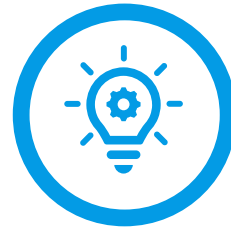
- Define the purpose and audience
- Select the right tables
- Improve Metadata Quality



### 2. Configure

Tailor Genie's logic, context and trusted assets

- Review Value Dictionaries
- Add Custom Instructions
- Create Trusted Assets



### 3. Validate

Quality control and validate results

- Improve accuracy through Benchmark testing.
- Validate results to measure accuracy.



### 4. Monitor & Refine

Monitor user responses and apply feedback

- Monitor user feedback
- Continuous Improvement through refining configurations.





# Key Takeaways: Building a Trusted, Self-Serve Genie Space

---

- To democratize data, users need to “talk to the data” using natural language.
  - Genie enables this – but will only shine with a thoughtful setup.
- Metadata, Value Dictionaries and Trusted Assets help Genie interpret questions and generate reliable answers.
- Custom instructions align Genie’s tone, logic, and priorities with business context.
- Benchmarking and Feedback loops are essential to continuously improve the data companion’s accuracy and usability.
- The code is in my Github repo:
  - <https://github.com/wilson-mok/demo/tree/main/2025/MSDEVMTL/2-Databricks-Chatbot-Companion>

# Thank you



LinkedIn