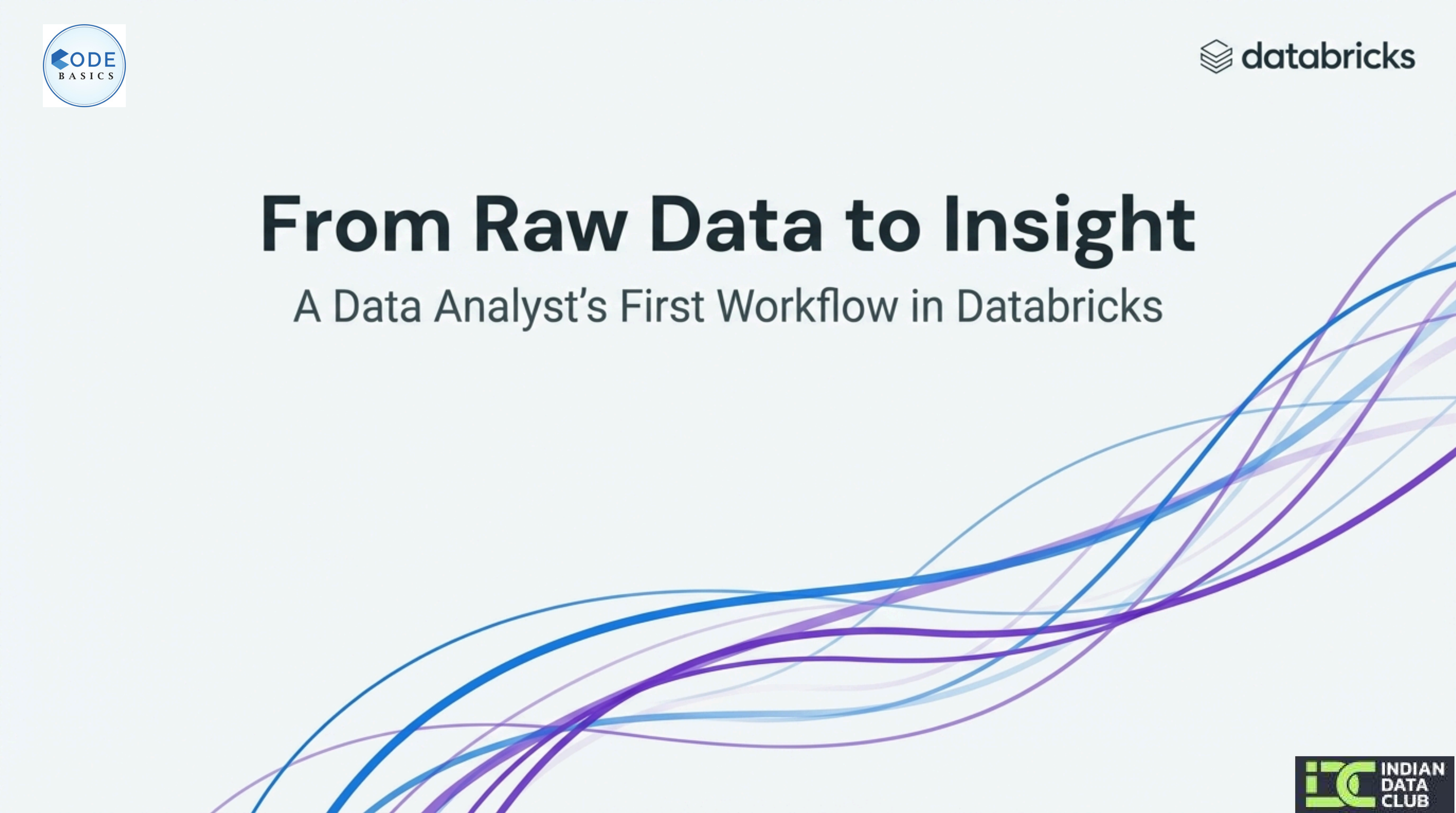
CODE BASICS

databricks

From Raw Data to Insight

A Data Analyst's First Workflow in Databricks



INDIAN DATA CLUB

The Objective: Analyze a New E-commerce Dataset

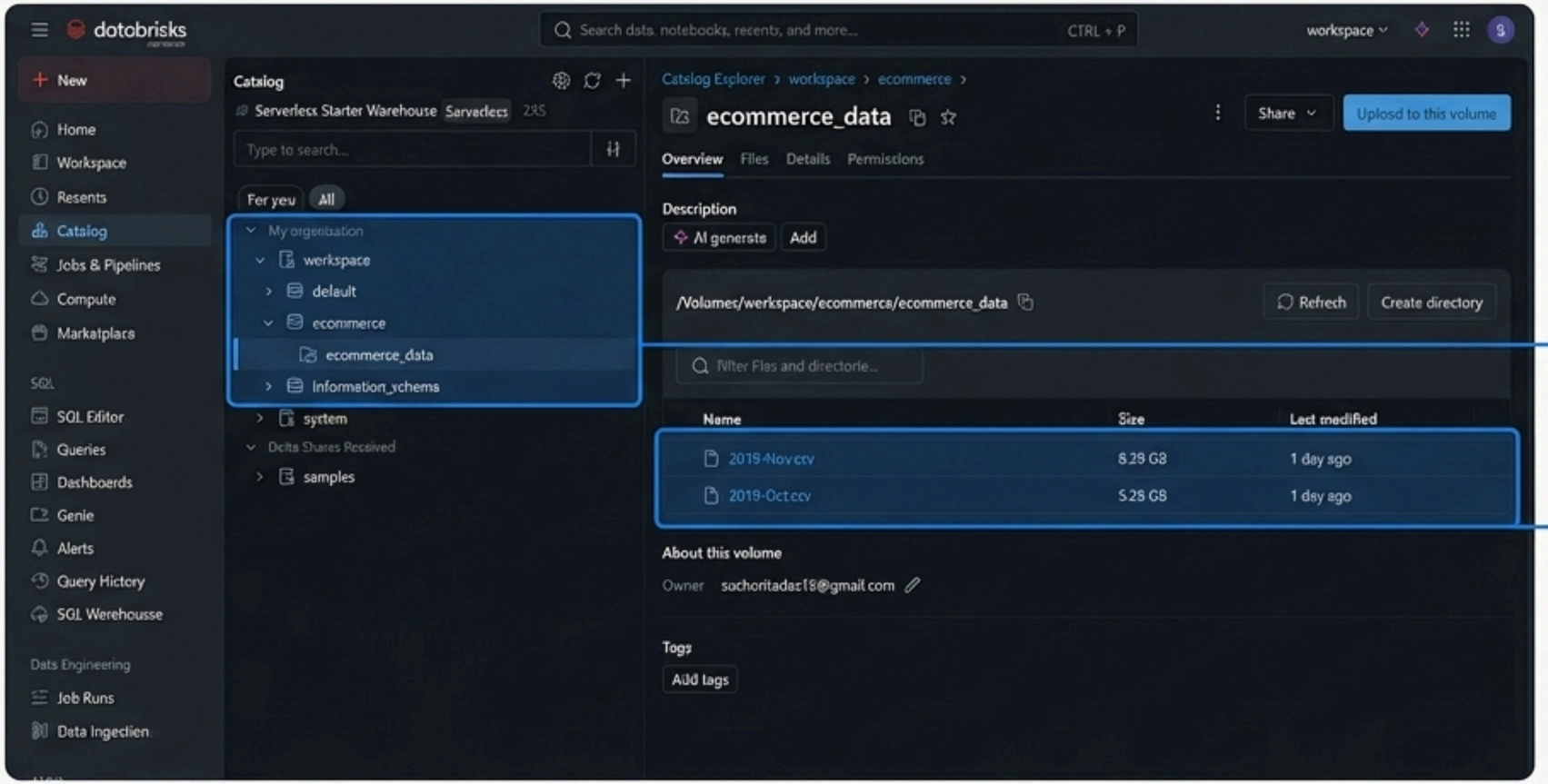
Our goal is to perform an initial exploration of a large e-commerce dataset. We need to locate the source files, load the data into a workable format, and ask a basic analytical question to verify our setup. The entire process will take place within the Databricks Data Intelligence Platform.

Discover Data

Load & Process

Query & Analyze

Step 1: Discovering Our Data in the Unity Catalog



Before we can analyze anything, we need to find our data. The Unity Catalog provides a single, organized view of all our data assets.

We navigate a familiar folder structure to find our project files.

Our source files are substantial, totaling over 13 GB. We need a tool that can handle this scale.

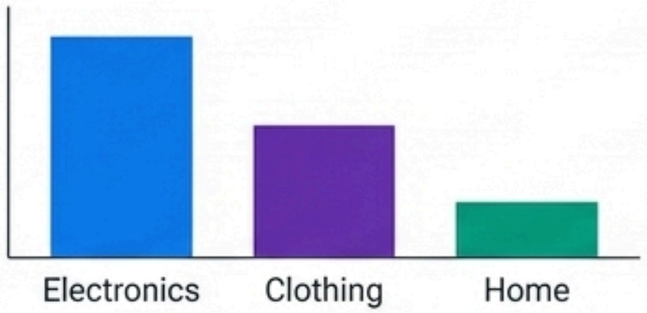
INDIAN DATA CLUB

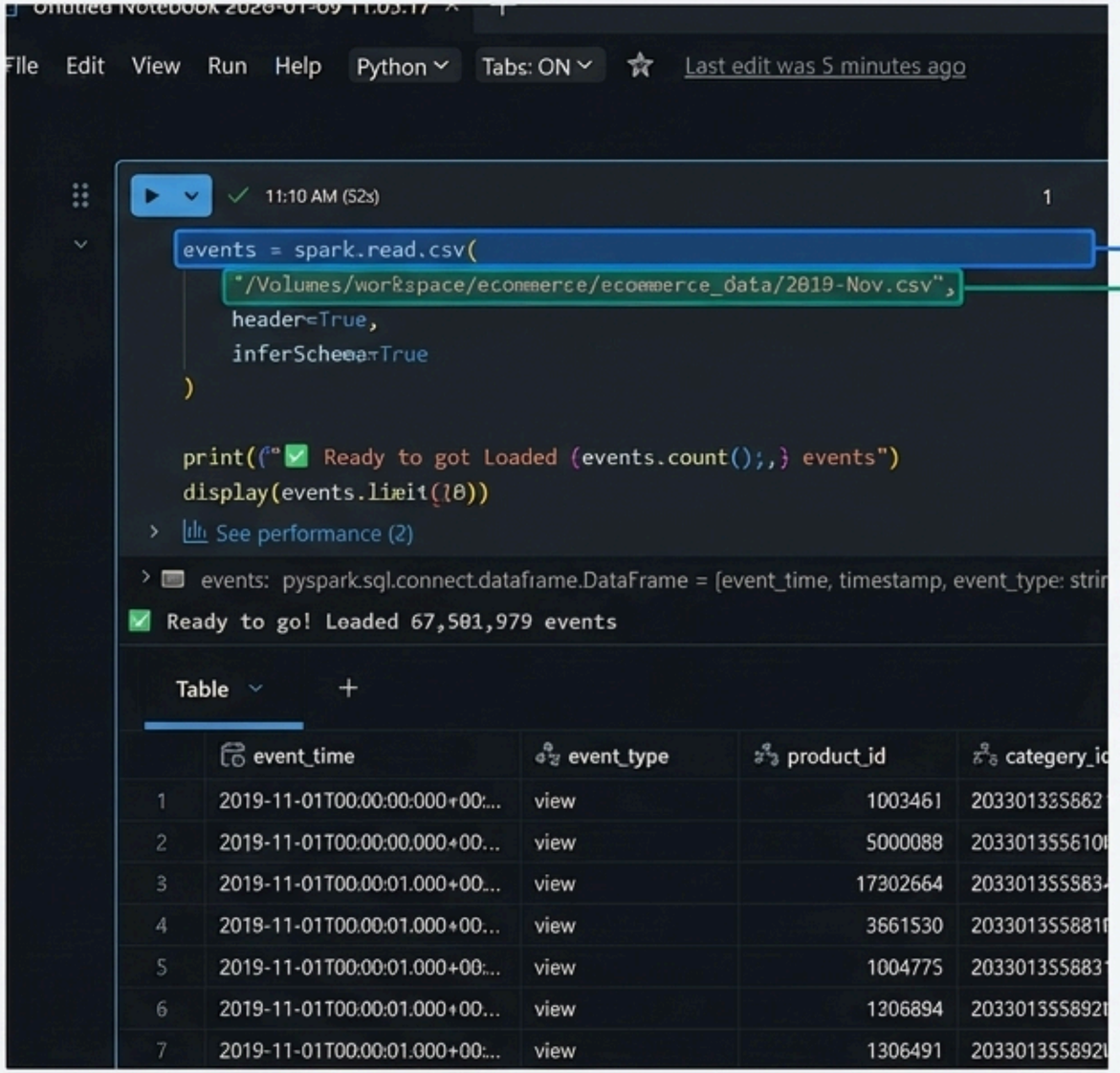
Our Workspace: The Interactive Databricks Notebook

The Databricks Notebook is our canvas. It's a multi-language, interactive environment where we can write and execute code, see results, and document our findings all in one place. Here, we'll use Python with Spark to process our data.

```python
# Load e-commerce data
df = spark.read.format('csv').load('path/to/data')

# Display data summary
df.groupBy('category').count().display()
```





Step 2: Bringing 67 Million Rows to Life

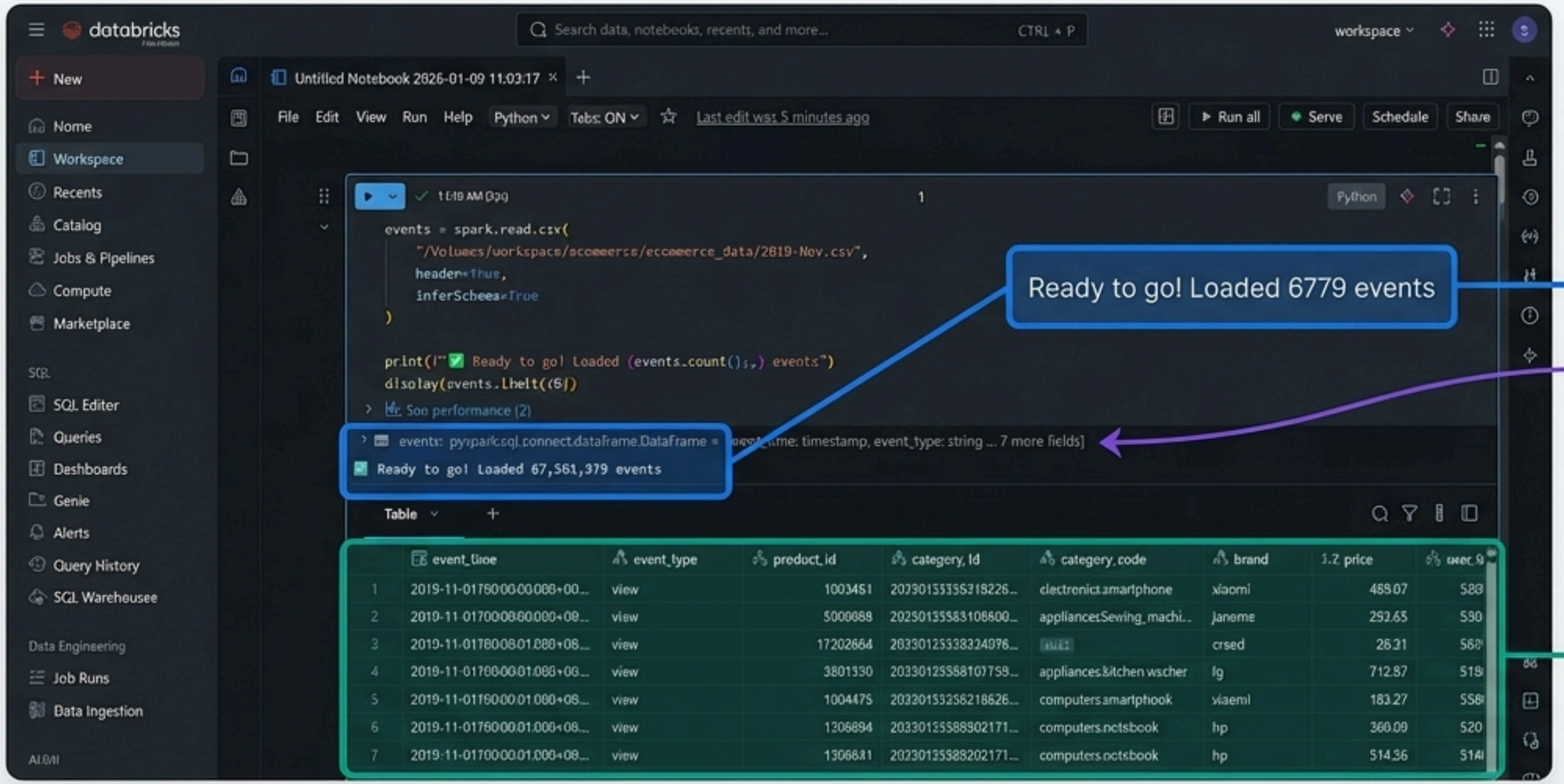
The core command for loading data.

With a single command, `spark.read.csv`, we can ingest the entire 8.39 GB November dataset. We instruct Spark to use the first row as the header and to automatically infer the data types for each column (`header=True`, `inferSchema=True`).

We use the direct path we found in the Unity Catalog.

INDIAN DATA CLUB

From One Command to a Fully Loaded DataFrame



67.5 Million Rows

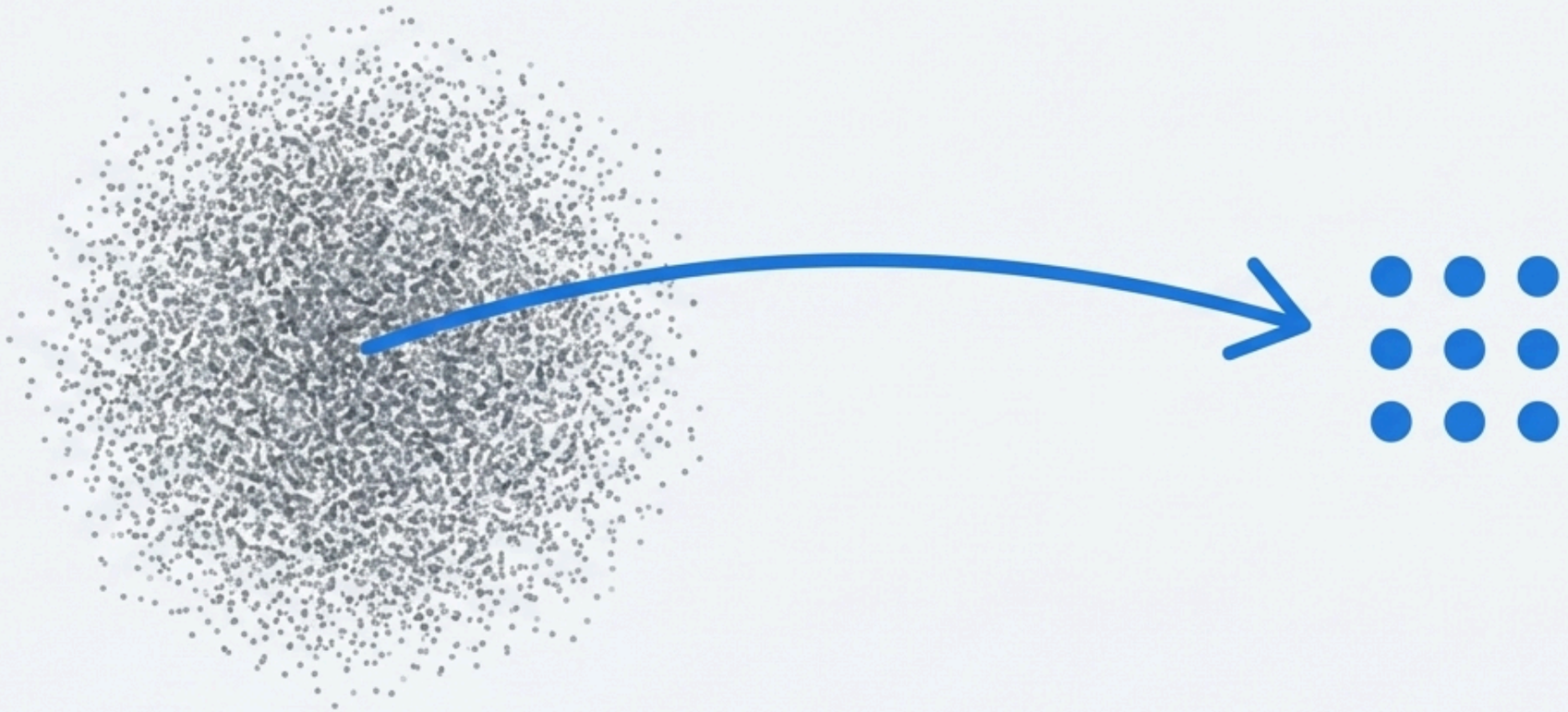
The code executes and immediately provides a confirmation count and a preview of the data.

The data is now a Spark DataFrame, a powerful, distributed data structure ready for analysis.

INDIAN DATA CLUB

Shifting from Macro to Micro to Master a Key Skill

We've confirmed we can handle massive datasets. To clearly understand the fundamentals of data manipulation, let's switch gears. We will now create a small, simple DataFrame from scratch. This allows us to focus on the **logic** of our analysis without the complexity of 67 million rows.



Step 3: Building a Sample DataFrame in Seconds

We can define data directly in our notebook using a simple Python list. Then, the `spark.createDataFrame` command instantly converts it into a fully functional Spark DataFrame, complete with named columns: "product" and "price".

code cell

```
# Create simple DataFrame
data = [('iPhone', 999), ('Samsung', 799), ('MacBook', 1299)]

df = spark.createDataFrame(data, ['product', 'price'])
df.show()
```

Standard Python list.

The command to convert our list into a distributed DataFrame.

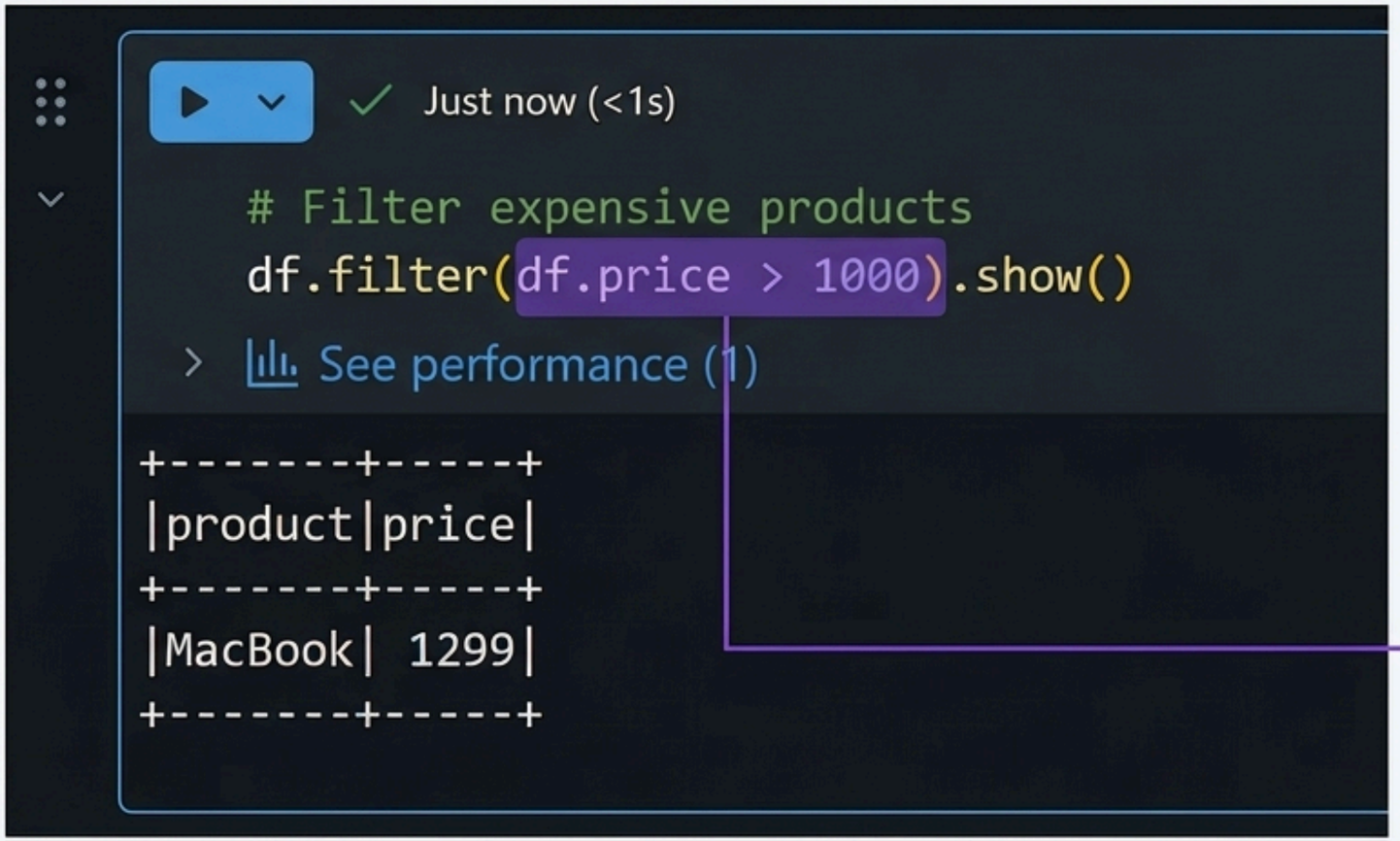
Our Sample Data, Ready for Analysis

Table +

	product	price
1	iPhone	999
2	Samsung	799
3	MacBook	1299

The result is a clean, structured table. Now that we have a simple dataset, we can easily ask questions of it.

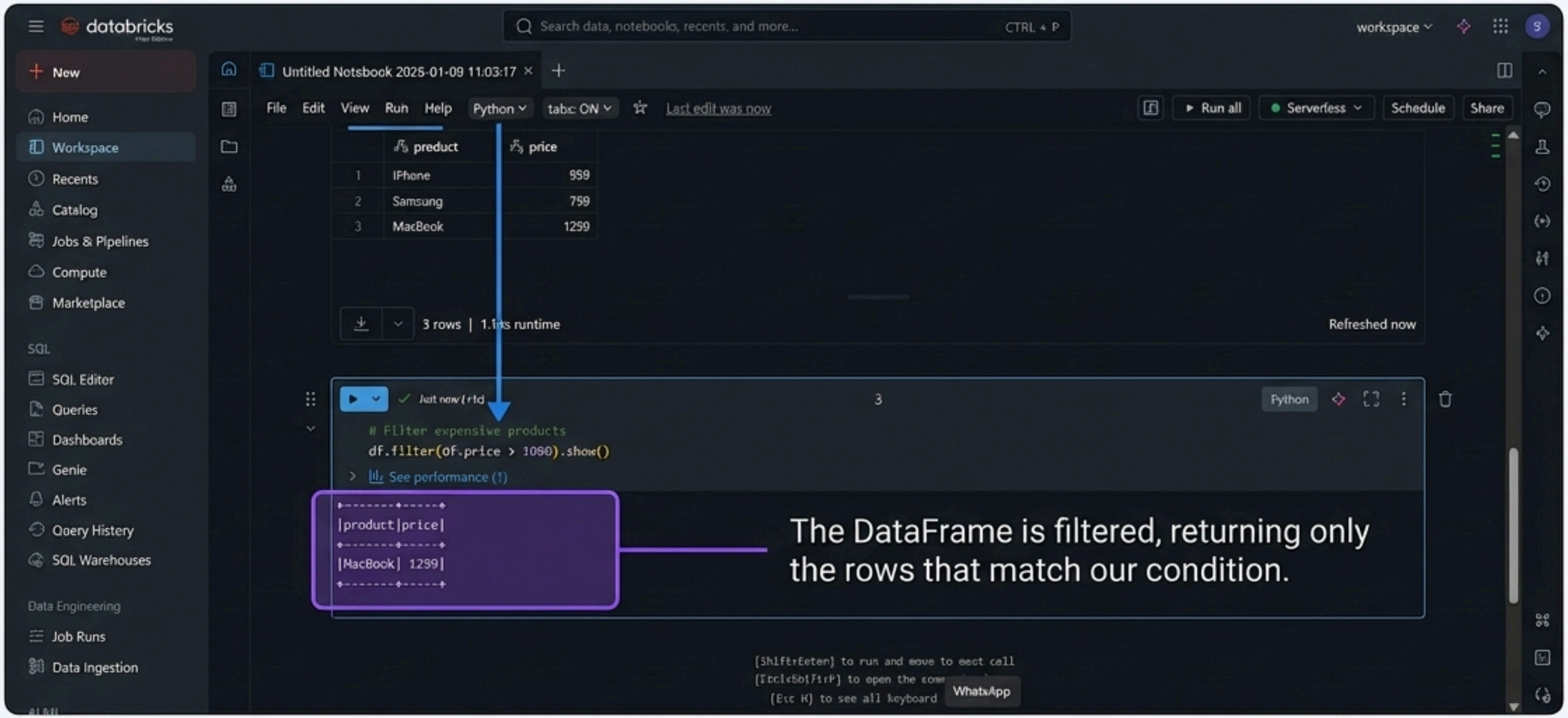
Step 4: Asking Our First Question—Which Products Cost More Than $1000?



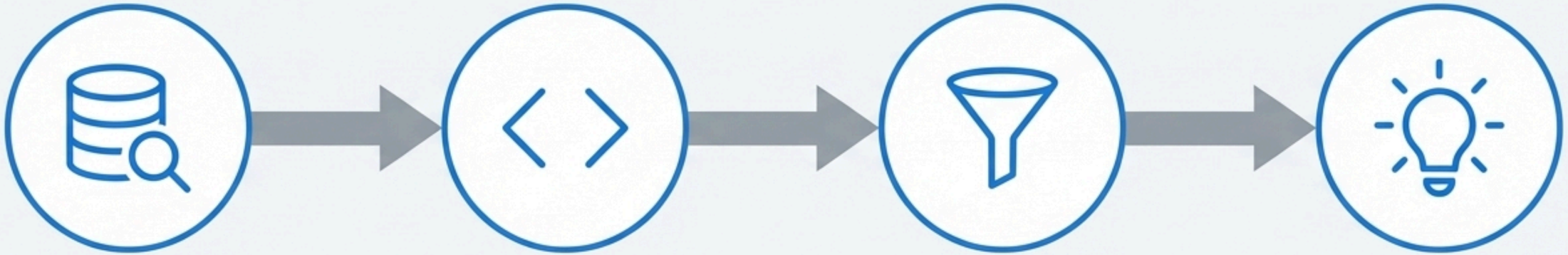
DataFrames have powerful built-in methods for analysis. To find expensive products, we use the `.filter()` method. The syntax is clear and expressive, directly translating our question into code. The `.show()` command displays the result.

The logical condition at the heart of our question.

The Answer, Instantly Delivered



The Complete Workflow: From Discovery to Insight



Discover: Located our raw data files in the Unity Catalog.

Load: Ingested 67.5M rows into a Spark DataFrame with a single command.

Analyze: Asked a specific question using the `.filter()` method on a sample DataFrame.

Insight: Received an immediate, precise answer to our question.

Databricks Notebook

Simple Tools for Complex Data

The Databricks platform provides a unified and interactive workflow. By seamlessly combining data discovery in the Unity Catalog with the large-scale processing of Spark and the analytical flexibility of DataFrames, data professionals can move from question to insight with unprecedented speed and simplicity.