

Practical - 6

Name: Sakshi Deshmukh

PRN: 202301040191

Roll no. 154

AWS Athena

Setting up S3 bucket

The screenshot shows the 'Create bucket' wizard on the AWS S3 service. The 'General configuration' section is active, displaying the following details:

- AWS Region:** Europe (Stockholm) eu-north-1
- Bucket type:** General purpose (selected)
- Bucket name:** sakshi-athena-practical
- Copy settings from existing bucket - optional:** Choose bucket (button)
- Format:** s3://bucket/prefix

The 'Object Ownership' section is visible below, indicating control ownership of objects written to this bucket.

The screenshot shows the main AWS S3 buckets page. A green success message at the top states: "Successfully created bucket 'sakshi-athena-practical'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, the 'General purpose buckets' section lists the newly created bucket:

Name	AWS Region	Creation date
sakshi-athena-practical	Europe (Stockholm) eu-north-1	October 28, 2025, 15:35:48 (UTC+05:30)

Other sections visible include 'Account snapshot' (updated daily) and 'External access summary' (new, updated daily).

Bucket for storing results

The screenshot shows the 'Create bucket' wizard on the Amazon S3 service. The 'General configuration' section is active, displaying the following details:

- AWS Region:** Europe (Stockholm) eu-north-1
- Bucket type:** General purpose (selected)
- Bucket name:** sakshi-athena-practical-results
- Copy settings from existing bucket - optional:** Choose bucket (button)
- Format:** s3://bucket/prefix

The 'Object Ownership' section is also visible, indicating control ownership of objects written to this bucket.

The screenshot shows the 'Buckets' page on the Amazon S3 service. A green success message at the top states: "Successfully created bucket 'sakshi-athena-practical-results'. To upload files and folders, or to configure additional bucket settings, choose View details." The 'General purpose buckets' tab is selected, showing the following table of buckets:

Name	AWS Region	Creation date
sakshi-athena-practical	Europe (Stockholm) eu-north-1	October 28, 2025, 15:35:48 (UTC+05:30)
sakshi-athena-practical-results	Europe (Stockholm) eu-north-1	October 28, 2025, 15:54:28 (UTC+05:30)

On the right side, there are two informational cards: 'Account snapshot' (updated daily) and 'External access summary - new' (updated daily).

Creating folder for non-partitioned data

The screenshot shows the 'Create folder' page in the AWS S3 console. The URL is [Amazon S3 > Buckets > sakshi-athena-practical > Create folder](#). The 'Folder' section has 'Folder name' set to 'non-partitioned-data'. A note about bucket policy blocking folder creation is present. Under 'Server-side encryption', the 'Don't specify an encryption key' option is selected. The bottom navigation bar includes CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Upload data into folder

The screenshot shows the 'Upload' page in the AWS S3 console. The URL is [Amazon S3 > Buckets > sakshi-athena-practical > Upload](#). A file named 'product_data.csv' is selected for upload. The 'Destination' section shows the target bucket as 's3://sakshi-athena-practical'. The bottom navigation bar includes CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Upload succeeded
For more information, see the [Files and folders](#) table.

Upload: status

ⓘ After you navigate away from this page, the following information is no longer available.

Summary

Destination	Succeeded	Failed
s3://sakshi-athena-practical	1 file, 632.0 B (100.00%)	0 files, 0 B (0%)

Files and folders [Configuration](#)

Files and folders (1 total, 632.0 B)

Name	Folder	Type	Size	Status	Error
product_data.csv	-	text/csv	632.0 B	SUCCEEDED	-

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Amazon S3 > Buckets > [sakshi-athena-practical](#) > non-partitioned-data/

non-partitioned-data/

[Copy S3 URI](#)

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Name	Type	Last modified	Size	Storage class
product_data.csv	csv	October 28, 2025, 15:41:06 (UTC+05:30)	632.0 B	Standard

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Creating folder for partitioned data

The screenshot shows the 'Create folder' page in the AWS S3 console. The URL is [Amazon S3 > Buckets > sakshi-athena-practical > Create folder](#). The 'Folder name' field contains 'partitioned-data/'. A note at the top says: 'Your bucket policy might block folder creation. If your bucket policy prevents uploading objects without specific tags, metadata, or access control list (ACL) grantees, you will not be able to create a folder using this configuration. Instead, you can use the [upload configuration](#) to upload an empty folder and specify the appropriate settings.' Below the folder name, there's a note about server-side encryption: 'Server-side encryption protects data at rest. The following encryption settings apply only to the folder object and not to sub-folder objects.' Under 'Server-side encryption', the 'Don't specify an encryption key' option is selected. At the bottom, there are links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Creating folder for each category

The screenshot shows the 'Create folder' page in the AWS S3 console. The URL is [Amazon S3 > Buckets > sakshi-athena-practical > partitioned-data/ > Create folder](#). The 'Folder name' field contains 'category=books'. A note at the top says: 'Your bucket policy might block folder creation. If your bucket policy prevents uploading objects without specific tags, metadata, or access control list (ACL) grantees, you will not be able to create a folder using this configuration. Instead, you can use the [upload configuration](#) to upload an empty folder and specify the appropriate settings.' Below the folder name, there's a note about server-side encryption: 'Server-side encryption protects data at rest. The following encryption settings apply only to the folder object and not to sub-folder objects.' Under 'Server-side encryption', the 'Don't specify an encryption key' option is selected. At the bottom, there are links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

The screenshot shows the AWS S3 console interface. At the top, there's a search bar and navigation links for 'Amazon S3 > Buckets > sakshi-athena-practical > partitioned-data/'. The main area displays a list of objects under 'partitioned-data/' with three items: 'category=books/', 'category=clothing/', and 'category=electronics/'. Each item is a folder. Below the list are buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. A note about Amazon S3 inventory is present. The bottom of the screen includes links for 'CloudShell' and 'Feedback'.

Uploading files in each folder

The screenshot shows the AWS S3 console interface for uploading files. The path is 'Amazon S3 > Buckets > sakshi-athena-practical > partitioned-data/ > category=books/ > Upload'. The 'Upload' section has a note about file size limits. It features a drag-and-drop area and buttons for 'Add files' or 'Add folder'. Below this is a 'Files and folders' table with one entry: 'books_data.csv' (text/csv, 156.0 B). There are 'Remove', 'Add files', and 'Add folder' buttons next to the table. The 'Destination' section shows the destination as 's3://sakshi-athena-practical/partitioned-data/category=books/'. The 'Destination details' section notes that bucket settings impact new objects. The bottom of the screen includes links for 'CloudShell' and 'Feedback'.

Screenshot of the AWS S3 console showing the contents of the 'category=books/' folder in the 'sakshi-athena-practical' bucket.

The page title is 'category=books/'. The top navigation bar shows 'Amazon S3 > Buckets > sakshi-athena-practical > partitioned-data/ > category=books/'. The top right corner shows 'Account ID: 7725-4885-8659' and 'Europe (Stockholm)'. The search bar at the top left contains 'Search [Alt+S]'. The main content area displays a table of objects:

Name	Type	Last modified	Size	Storage class
books_data.csv	csv	October 28, 2025, 15:49:56 (UTC+05:30)	156.0 B	Standard

Actions available for the object include Copy S3 URI, Copy URL, Download, Open, Delete, Actions, Create folder, and Upload. A 'Find objects by prefix' search bar is also present.

Screenshot of the AWS S3 console showing the contents of the 'category=clothing/' folder in the 'sakshi-athena-practical' bucket.

The page title is 'category=clothing/'. The top navigation bar shows 'Amazon S3 > Buckets > sakshi-athena-practical > partitioned-data/ > category=clothing/'. The top right corner shows 'Account ID: 7725-4885-8659' and 'Europe (Stockholm)'. The search bar at the top left contains 'Search [Alt+S]'. The main content area displays a table of objects:

Name	Type	Last modified	Size	Storage class
clothing_data.csv	csv	October 28, 2025, 15:51:17 (UTC+05:30)	135.0 B	Standard

Actions available for the object include Copy S3 URI, Copy URL, Download, Open, Delete, Actions, Create folder, and Upload. A 'Find objects by prefix' search bar is also present.

The screenshot shows the AWS S3 console interface. At the top, the navigation bar includes the AWS logo, a search bar with placeholder text 'Search [Alt+S]', and account information: Account ID: 7725-4885-8659, Europe (Stockholm), and user sakshi-aws. Below the navigation bar, the breadcrumb path is: Amazon S3 > Buckets > sakshi-athena-practical > partitioned-data/ > category=electronics/. The main content area displays a single object: 'category=electronics/'. The 'Objects' tab is selected, showing one item: 'Objects (1)'. The object details are as follows:

Name	Type	Last modified	Size	Storage class
electronics_data.csv	csv	October 28, 2025, 15:52:04 (UTC+05:30)	156.0 B	Standard

Below the table, there are buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. A search bar labeled 'Find objects by prefix' is also present. At the bottom of the page, there are links for CloudShell, Feedback, and copyright information: © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

Launching Athena

The screenshot shows the AWS Athena landing page. The top navigation bar is identical to the S3 screenshot, with the AWS logo, search bar, and account information. The main content area has a dark background with white text. It features a large heading 'Amazon Athena' and sub-headline 'Start querying data instantly.' Below this, a subtext explains: 'Amazon Athena is an interactive query service that makes it easy to analyse data in Amazon S3 and other federated data sources using standard SQL.' On the left, there's a section titled 'How it works' with an icon of a bucket and a sparkler, and a numbered list: '1. Point to your data source'. To the right, there are three callout boxes: 'Begin querying your data' (with a 'Explore the query editor' button), 'Pricing' (listing rates for SQL queries per TB scanned and per DPU hour), and 'Getting started' (with a link to add a data source). At the bottom, there are links for CloudShell, Feedback, and copyright information: © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

Setting query result location

The screenshot shows the 'Manage settings' page for Amazon Athena. In the 'Query result location and encryption' section, there is a field labeled 'Location of query result - optional' with the value 's3://sakshi-athena-practical-results'. Below this, a note says 'You can create and manage lifecycle rules for this bucket' with a link to 'Lifecycle configuration'. Under 'Expected bucket owner - optional', there is a note about specifying an AWS account ID and a dropdown menu with the placeholder 'Enter AWS account ID'. There are three checkboxes at the bottom: 'Assign bucket owner full control over query results', 'Encrypt query results', and 'Turned off'. At the bottom right are 'Cancel' and 'Save' buttons.

The screenshot shows the 'Settings' page for Amazon Athena. A green banner at the top says 'Settings successfully updated.' In the 'Query result encryption' section, there are four tabs: 'Query result location' (set to 's3://Sakshi-athena-practical-results'), 'Encrypt query results' (disabled), 'Expected bucket owner' (disabled), and 'Assign bucket owner full control over query results' (disabled). The 'Assign bucket owner full control over query results' tab has a 'Manage' button. At the bottom right are 'CloudShell' and 'Feedback' buttons.

AWS Search [Alt+S] Account ID: 7725-4885-8659 ▾ Europe (Stockholm) ▾ salkshi-aws

Amazon Athena > Query editor tabs

Editor Recent queries Saved queries Settings Workgroup primary (+) ▾

Data Data source AwsDataCatalog Catalogue None Database my_csv_database Tables and views Create Tables (2) < 1 > partitionedproduct Partitioned producttest Views (0) < 1 > Filter tables and views

Query 1 : X | Query 2 : X | Query 3 : X | Query 4 : X

```
1 CREATE DATABASE IF NOT EXISTS pectest;
```

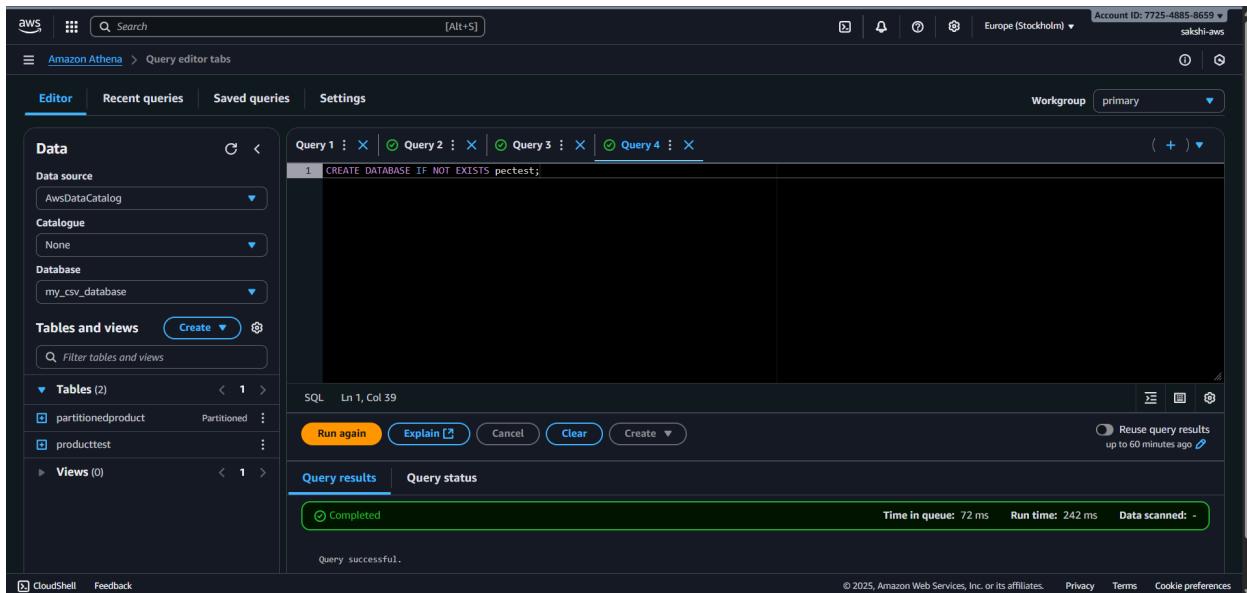
SQL Ln 1, Col 39 Run again Explain Cancel Clear Create Reuse query results up to 60 minutes ago

Query results Query status

Completed Time in queue: 72 ms Run time: 242 ms Data scanned: -

Query successful.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



AWS Search [Alt+S] Account ID: 7725-4885-8659 ▾ Europe (Stockholm) ▾ salkshi-aws

Amazon Athena > Query editor tabs

Editor Recent queries Saved queries Settings Workgroup primary (+) ▾

Data Data source AwsDataCatalog Catalogue None Database pectest Tables and views Create Tables (2) < 1 > partitionedproduct Partitioned producttest Views (0) < 1 > Filter tables and views

Query 4 : X | Query 5 : X | Query 6 : X

```
1 CREATE EXTERNAL TABLE IF NOT EXISTS pectest.producttest (
2   `product_id` int,
3   `product_name` string,
4   `category` string,
5   `price` double,
6   `stock_quantity` int
7 )
8 ROW FORMAT SERDE "org.apache.hadoop.hive.serde2.OpenCSVSerde"
9 WITH SERDEPROPERTIES (
10   'separatorChar' = ',',
11   'quoteChar' = '\"',
12   'escapeChar' = '\\'
13 )
14 LOCATION 's3://aditya-athena-pract6-data/non_partitioned_data/'
15 TBLPROPERTIES ('skip.header.line.count' = '1');
```

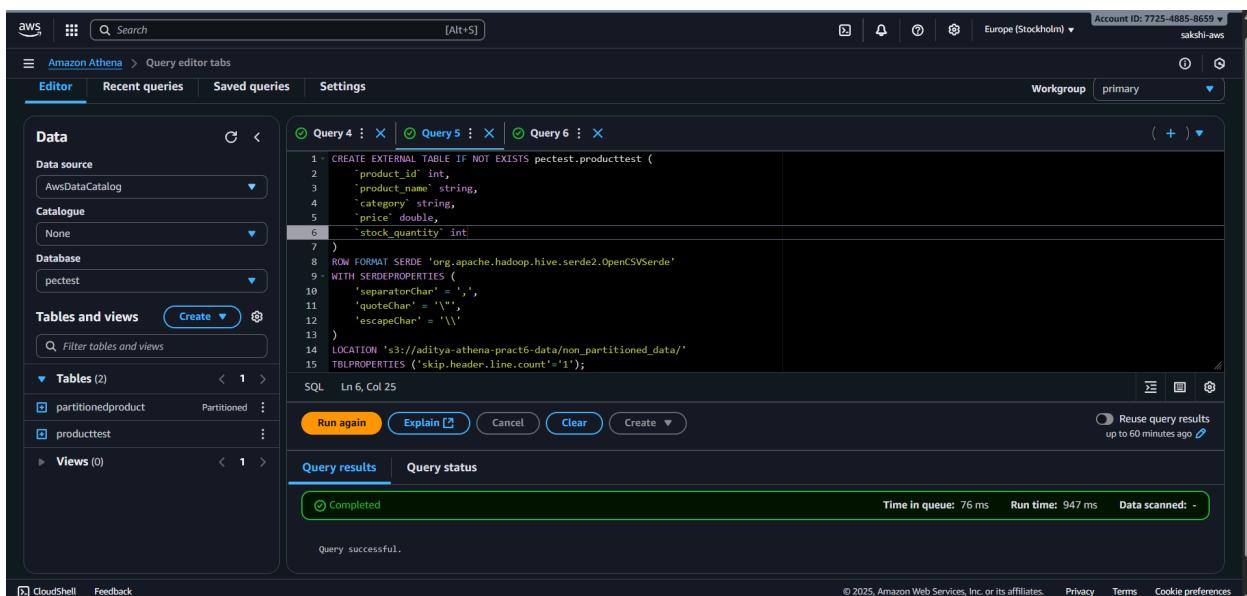
SQL Ln 6, Col 25 Run again Explain Cancel Clear Create Reuse query results up to 60 minutes ago

Query results Query status

Completed Time in queue: 76 ms Run time: 947 ms Data scanned: -

Query successful.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



aws Search [Alt+S] Account ID: 7725-4885-8659 ▾ Europe (Stockholm) ▾ sakshi-aws

Amazon Athena > Query editor tabs

Editor Recent queries Saved queries Settings Workgroup primary (+) ▾

Data

Data source AwsDataCatalog Catalogue None Database pectest Tables and views Create ▾

Tables (2) < 1 >

- partitionedproduct Partitioned
- producttest

Views (0) < 1 >

CloudShell Feedback

Query 4 : X | Query 5 : X | Query 6 : X | Query 7 : X | Query 8 : X | Query 9 : X

```
1 CREATE EXTERNAL TABLE IF NOT EXISTS pectest.partitionedproduct (
2     `product_id` int,
3     `product_name` string,
4     `price` double,
5     `stock_quantity` int
6 )
7 PARTITIONED BY(`category` string)
8 ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'
9 WITH SERDEPROPERTIES (
10     'separatorChar' = ',',
11     'quoteChar' = '\"',
12     'escapeChar' = '\\'
13 )
14 LOCATION 's3://aditya-athena-pract6-data/partitioned_data/'
15 TBLPROPERTIES ('skip.header.line.count' = '0');
```

SQL Ln 15, Col 46

Run again Explain Cancel Clear Create ▾ Reuse query results up to 60 minutes ago

Query results Query status

Completed Time in queue: 83 ms Run time: 925 ms Data scanned: -

Query successful.

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws Search [Alt+S] Account ID: 7725-4885-8659 ▾ Europe (Stockholm) ▾ sakshi-aws

Amazon Athena > Query editor tabs

Editor Recent queries Saved queries Settings Workgroup primary (+) ▾

Data

Data source AwsDataCatalog Catalogue None Database pectest Tables and views Create ▾

Tables (2) < 1 >

- partitionedproduct Partitioned
- producttest

Views (0) < 1 >

CloudShell Feedback

Query 4 : X | Query 5 : X | Query 6 : X | Query 7 : X | Query 8 : X | Query 9 : X

```
1 MSCK REPAIR TABLE pectest.partitionedproduct;
```

SQL Ln 1, Col 1

Run again Explain Cancel Clear Create ▾ Reuse query results up to 60 minutes ago

Query results Query status

Completed Time in queue: 168 ms Run time: 1.249 sec Data scanned: -

Partitions not in metastore: partitionedproduct:category=Books partitionedproduct:category=Clothing partitionedproduct:category=Electronics
Repair: Added partition to metastore partitionedproduct:category=Books
Repair: Added partition to metastore partitionedproduct:category=Clothing
Repair: Added partition to metastore partitionedproduct:category=Electronics

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Previewing the partitioned table

The screenshot shows the Amazon Athena Query Editor interface. The left sidebar displays the Data source (AwsDataCatalog), Catalogue (None), and Database (pectest). Under Tables and views, there are two tables: partitionsproduct (Partitioned) and producttest. The current query tab is Query 4, which contains the SQL statement: `SELECT * FROM pectest.producttest LIMIT 10;`. The results section shows 10 rows of data from the producttest table:

#	product_id	product_name	category	price	stock_quantity
1	101	Classic Laptop	Electronics	1200.0	50
2	102	Smart-Home Hub	Electronics	150.0	120
3	103	4K Ultra-HD Monitor	Electronics	350.0	75
4	104	Gaming Mouse	Electronics	80.0	300
5	105	Wireless Earbuds	Electronics	199.0	250
6	201	The Great Gatsby	Books	15.0	400

At the bottom of the results table, there are links for "Copy" and "Download results CSV". The status bar at the bottom right indicates: Time in queue: 66 ms, Run time: 379 ms, Data scanned: 0.62 KB.

This screenshot is identical to the one above, showing the same query results for the producttest table. The data is identical to the first screenshot, with 10 rows of products from the Electronics category and Books category.

Querying the regular table [data scanned - 0.62 kb]

The screenshot shows the Amazon Athena Query Editor interface. On the left, the Data sidebar shows a single database 'petest' and two tables: 'partitionedproduct' and 'producttest'. The 'producttest' table is selected. The main area displays a query editor tab titled 'Query 4' containing the following SQL:

```
1 SELECT category, COUNT(*) as item_count
2 FROM petest.producttest
3 GROUP BY category;
```

Below the query editor is a results table titled 'Results (5)' showing the count of items by category:

category	item_count
Books	5
Electronics	5
Clothing	4

At the bottom right of the results table, it says 'Time in queue: 101 ms Run time: 364 ms Data scanned: 0.62 kB'.

Querying the partitioned table [data scanned - 0.15 kb]

The screenshot shows the Amazon Athena Query Editor interface. On the left, the Data sidebar shows a single database 'petest' and two tables: 'partitionedproduct' and 'producttest'. The 'partitionedproduct' table is selected. The main area displays a query editor tab titled 'Query 4' containing the following SQL:

```
1 SELECT product_name, price
2 FROM petest.partitionedproduct
3 WHERE category = 'Books';
```

Below the query editor is a results table titled 'Results (5)' showing book names and prices:

product_name	price
The Great Gatsby	45.0
A Tale of Two Cities	42.0
Learning Python	45.0
A Brief History of Time	22.0
Dune	38.0

At the bottom right of the results table, it says 'Time in queue: 106 ms Run time: 570 ms Data scanned: 0.15 kB'.

The non-partitioned table `producttest` required Athena to scan the entire `products_full.csv` file to find rows `WHERE category = 'Books'`. The partitioned table `partitionedproduct` only scanned the data inside the `category=Books/` folder. This is shown in the 'Data scanned' metric, which was significantly lower for the partitioned query. This demonstrates how partitioning improves query performance and reduces cost by pruning data.