



# Hands-on Lab : Views in PostgreSQL

**Estimated time needed:** 15 minutes

In this lab, you will learn how to create and execute views and materialized views in the PostgreSQL database service using the pgAdmin graphical user interface (GUI) tool. Materialized views behave differently compared to regular views. In materialized views, the result set is materialized, or saved for future use. You can't insert, update, or delete rows like in regular views. Essentially, materialized views store the results of a database query as a separate table-like object so that the query results can be accessed at a later time without having to re-run the query. As a result, materialized views can improve database performance compared to regular views.

## Software Used in this Lab

In this lab, you will use the [PostgreSQL Database](#). PostgreSQL is a Relational Database Management System (RDBMS) designed to efficiently store, manipulate, and retrieve the data.

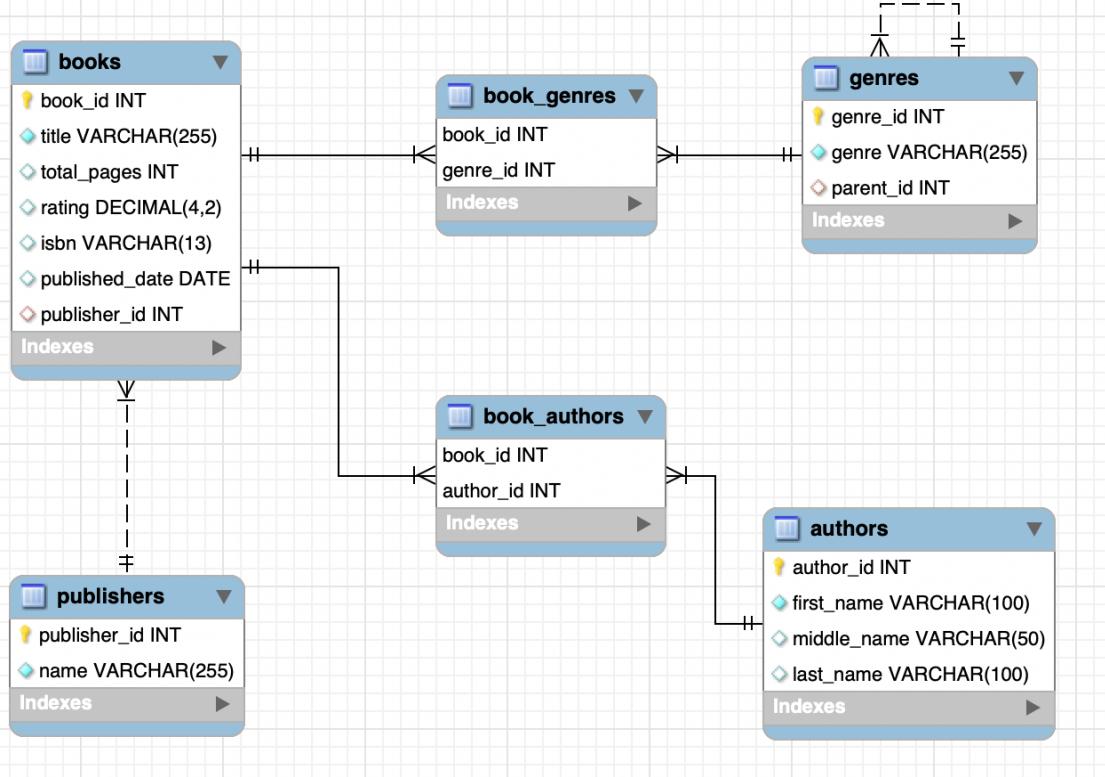


To complete this lab you will utilize the PostgreSQL relational database service available as part of IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs is a virtual lab environment used in this course.

## Database Used in this Lab

The eBooks database has been used in this lab.

The following ERD diagram shows the schema of the complete eBooks database used in this lab:



## Objectives

After completing this lab, you will be able to use pgAdmin with PostgreSQL to:

- Restore a database schema and data.
- Create and execute a view.
- Create and execute a materialized view.

## Lab Structure

In this exercise, you will go through three tasks where you will learn how to create and execute views and materialized views in the PostgreSQL database service using the pgAdmin graphical user interface (GUI) tool.

### Task A: Restore a database schema and data

To get started with this lab, you will first download the relevant **eBooks** database dump file, then launch PostgreSQL and pgAdmin using the Cloud IDE. You can do this by following these steps:

1. Download the **eBooks** PostgreSQL dump file (containing the eBooks database schema and data) below to your local computer storage.
  - [eBooks\\_pgsql\\_dump.tar](#)
2. Click on the Skills Network extension button on the left side of the window.
3. Open the “DATABASES” drop down menu and click on “PostgreSQL”
4. Click on the “Start” button. PostgreSQL may take a few moments to start.



SKILLS NETWO...



## ▼ DATABASES



MySQL



PostgreSQL



MongoDB

## &gt; CLOUD

## &gt; BIG DATA



Cassandra

PostgreSQL x

# PostgreSQL

INACTIVE

2

v13.2 | v5.0 | v13.2

Start

3

Summary

Connection Information

Details

Get started with Postgres in a faster, easier way. To launch your database

1



✖ 0 ⚠ 0

5. Next, open the pgAdmin Graphical User Interface by clicking the “pgAdmin” button in the Cloud IDE interface.

[Stop](#)[\*\*Summary\*\*](#)[\*\*Connection Information\*\*](#)[\*\*Details\*\*](#)

Your database and pgAdmin server are now ready to use and available with the following on how to navigate PostgreSQL, please check out the Details section.

**Username:****Password:**

You can manage PostgreSQL via:

[pgAdmin](#)

Or to interact with the database in the terminal, select one of these options:

[PostgreSQL CLI](#)[New Terminal](#)

- Once the pgAdmin GUI opens, click on the Servers tab on the left side of the page. You will be prompted to enter a password.

PostgreSQL

pgAdmin x



<https://davidpastern-5050.theiadocker-6-labs-prod-theiak8s-4-tor01.proxy.com>

**pgAdmin**

Browser

Servers

pos

## Connect to Server

Please enter the password for the user 'postgres' to connect  
"postgres"

Password



Save Password



for the PostgreSQL database. It includes an administration interface, an SQL query debugger and much more. The tool needs of developers, DBAs and sys

## Quick Links

7. To retrieve your password, click on the “PostgreSQL” tab near the top of the interface.
8. Click on the Copy icon to the left of your password to copy the session password onto your clipboard.



PostgreSQL x pgAdmin

1

# PostgreSQL

ACTIVE

DB v13.2 | User v5.0 | Databases v13.2

Connect to PostgreSQL and pgAdmin directly in your Skills Network Labs environment.

Stop

Summary

Connection Information

Details

Your database and pgAdmin server are now ready to use and available with the following credentials. For more information on how to navigate PostgreSQL, please check out the Details section.

Username:

Password:

You can manage PostgreSQL via:

pgAdmin



Or to interact with the database in the terminal, select one of these options:

Or to interact with the database in the terminal, select one of these options.

PostgreSQL CLI

New Terminal

9. Navigate back to the “pgAdmin” tab and paste in your password, then click **OK**

10. You will then be able to access the pgAdmin GUI tool.

# pgAdmin

File ▾ Object ▾ Tools ▾ Help ▾

Browser



Dashboard

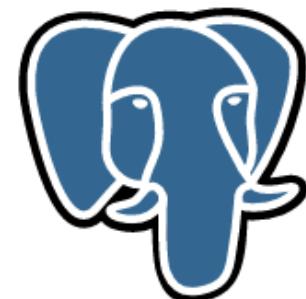
Properties

SQL

Statistics

> Servers

Welcome



**pgAdmin**

Management Tools

Feature rich | Maximises Pos

pgAdmin is an Open Source administration  
is designed to answer the needs of develop

Quick Links

## Getting Started



PostgreSQL Documentation

11. In the tree-view, expand **Servers** > **postgres** > **Databases**. Enter your PostgreSQL service session password if prompted during the process. Right-click on **Databases** and go to **Create** > **Database**. Type **eBooks** as name of the database and click **Save**.

Browser

1



Dashboard

Properties

SQL

Statistics

Servers (1)

2

postgres

3

Databases (1)

postgres

Create

Casts

Catalogs

Refresh...

Event Triggers

Extensions

Foreign Data Wrappers

Languages

Publications

Schemas

Subscriptions

Login/Group Roles

Tablespaces

## Server sessions

7

Database...



4

3

2

1

0

## Tuples in

1

## Create - Database

General    Definition    Security    Parameters    Advanced    SQL

Database

eBooks

Owner

 postgres

Comment

 i ? × Cancel ↕ Rese

12. In the tree-view, expand **eBooks**. Right-click on **eBooks** and select **Restore**.

Browser



Dashboard

## Servers (1)

## postres

## Databases (2)

## eBooks

Create

Copy

Edit

Delete/Drop

CREATE Script

Disconnect Database...

Generate ERD (Beta)

Sync

Sync

PostgreSQL

Create

Copy

Edit

## Database s

1

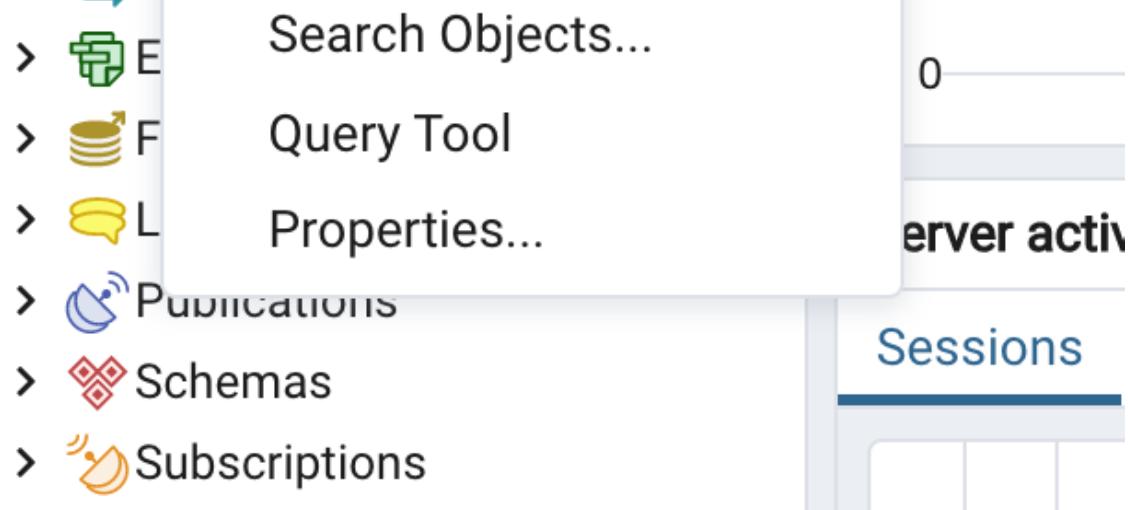
## Tables in

0

## uples in

1





13. Follow the instructions below to restore and proceed to Task B:

- On the **General** tab, click on the **Select file** button by the Filename box.

## Restore (Database: eBooks)

General

Restore options

Format

Custom or tar

Filename

Number of jobs

Role name

Select an item...



Cancel

- Click the **Upload File** button.

## Select file

		/var/lib/pgadmin/			
--	--	-------------------	--	--	--

Name	Size	Modified
sessions	4.0 kB	Mon Mar 20 10:43:20 2023
storage	4.0 kB	Mon Mar 20 10:43:20 2023

Show hidden files and folders?

**X** Ca

- Double-click on the drop files area and load the **eBooks\_pgsql\_dump.tar** you downloaded earlier from your local computer storage.

Select file

		/var/lib/pgadmin/			
--	--	-------------------	--	--	--

**Double click on this space**

Drop files here to upload. The file size limit (per file) is 50 mb.

Show hidden files and folders?

**X** Ca

- When the upload is complete, close the drop files area by clicking the X button.

## Select file



/var/lib/pgadmin/



76 KB



eBooks\_pgsql\_d...

100%

Drop files here to upload. The file size limit (per file) is 50 mb.

Show hidden files and folders?

**X** Ca

- Make sure Format is set to **All Files**, select the uploaded **eBooks\_pgsql\_dump.tar** file from the list, and then click the **Select** button.

## Select file

		/var/lib/pgadmin/eBooks_pgsql_dump.tar			
Name			Size	Modified	
	eBooks_pgsql_dump.tar		74.2 kB	Mon M	
	pgadmin4.db		156.0 kB	Mon M	
	sessions		4.0 kB	Mon M	
	storage		4.0 kB	Mon M	

Show hidden files and folders?

F

**X** Ca

- Now switch to **Restore options** tab.

## Restore (Database: eBooks)

General

Restore options

Format

Custom or tar

Filename

/var/lib/pgadmin/eBooks\_pgsql\_dump.tar

Number of jobs

Select an item...



Cancel

- Under Disable, set the Trigger option to Yes. Then click **Restore** button.

## Restore (Database: eBooks)

General    Restore options

### Queries

Include CREATE  
DATABASE  
statement

No

Clean before  
restore

No

Single  
transaction

No

### Disable

Trigger

Yes

No data for  
Failed Tables

No



✖ Cancel

Task B: Create and execute a view

1. In the tree-view, expand **eBooks > Schemas > public**. Right-click on **Views** and go to **Create > View**.

Browser



Dashboard

Properties

SQL

Statistics

Servers (1)

postgres

Databases (2)

1 eBooks

Casts

Catalogs

Event Triggers

Extensions

Foreign Data Wrappers

Languages

Publications

2 Schemas (1)

3 public

Collations

Domains

FTS Configurations

FTS Dictionaries

- > FTS Dictionaries
- > Aa FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Procedures
- > 1..3 Sequences
- > Tables (6)
- > Trigger Functions
- > Types

## 4 Views

- > Subscriptions
- ▼ postgres
  - > Casts
  - > Catalogs
  - > Event Triggers
  - > Extensions
  - > Foreign Data Wrappers
  - > Languages

## 5 Create

- Refresh...
- Grant Wizard...
- Search Objects...
- Query Tool

## 6 View...



2.

 Create - View**General**

Definition

Code

Security

SQL

Name

publisher\_and\_rating\_view

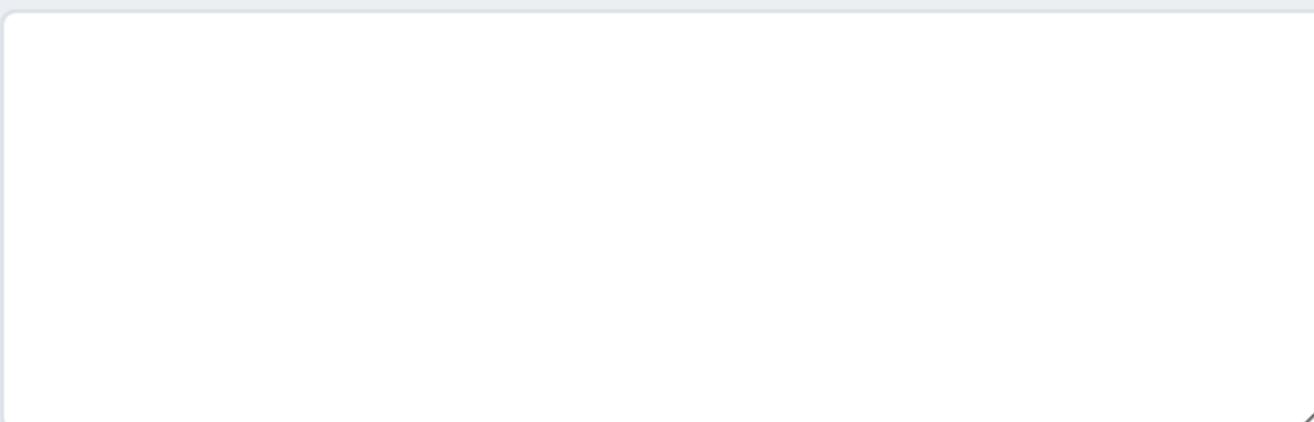
Owner

 postgres

Schema

 public

Comment



 i ? Cancel Reset Save

3. On the **Code** tab, copy and paste the code below. Then click **Save**.

```
1. 1
2. 2
1. SELECT books.title, books.rating, publishers.name
2. FROM books INNER JOIN publishers ON books.publisher_id = publishers.publisher_id
```

Copied!

General Definition **Code** Security SQL

```
1 SELECT books.title, books.rating, publishers.name  
2 FROM books INNER JOIN publishers ON books.publisher_id = publisher  
3
```

 i ?

✖ Cancel

4. In the tree-view, expand **Views**. Right-click on **publisher\_and\_rating\_view** and go to **View/Edit Data > All Rows**.

Browser



- ▼ Servers (1)
  - ▼ postgres
    - ▼ Databases (2)
      - ▼ eBooks
        - > Casts
        - > Catalogs
        - > Event Triggers
        - > Extensions
        - > Foreign Data Wrappers
        - > Languages
        - > Publications
      - ▼ Schemas (1)
        - ▼ public
          - > Collations
          - > Domains
          - > FTS Configurations
          - > FTS Dictionaries

Dashboard

Properties

SQL

## Database sessions

1

0

## Tuples in

18  
16  
14  
12  
10  
8  
6  
4

- > Aa FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Procedures
- > 1..3 Sequences
- > Tables (6)
- > Trigger Functions
- > Types

**1** Views (1)

**2** publisher\_and\_rating\_view

- > Columns
- > Rules
- > Triggers

> Subscriptions

**3** postgres

> Casts

> Catalogs

> Event Triggers

Create >

Refresh...

Delete/Drop

Drop Cascade

Scripts >

**4** View/Edit Data >

## Server activity

Sessions

Locks

Prepared

		PID	User
		83	postgres

- >  Event Triggers
- >  Extensions
- >  Foreign Data Wrappers
- >  Languages
- >  Publications

Search Objects...  
Query Tool  
Properties...

First 100 Rows  
Last 100 Rows  
Filtered Rows...

5. You will access the view you created. This allows you to actually access and view the contents of tables in your database.



## Query Editor    Query History

```
1  SELECT * FROM public.publisher_and_rating_view  
2
```

Data Output

Explain

Messages

Notifications

	<b>title</b> character varying (255)	<b>rating</b> numeric (4,2)	<b>name</b> character varying (255)
1	Lean Software Development: ...	4.17	Addison Wesley
2	Facing the Intelligence Explosi...	3.87	Machine Intelligence Researc...
3	Scala in Action	3.74	Manning
4	Patterns of Software: Tales fr...	3.84	Oxford University Press, USA
5	Anatomy Of LISP	4.43	McGraw-Hill
6	Computing machinery and int...	4.17	MSAC Philosophy Group
7	XML: Visual QuickStart Guide	3.66	Peachpit Press
8	SQL Cookbook	3.95	O'Reilly Media
9	The Apollo Guidance Comput...	4.29	Praxis Publications Inc

10	Minds and Computers: An Intr...	3.54	Edinburgh University Press
11	The Architecture of Symbolic ...	4.50	McGraw-Hill
12	Nmap Network Scanning: The...	4.32	Nmap Project
13	The It Handbook for Business:...	4.40	Createspace Independent Pub...
14	Accidental Empires	4.00	Harper
15	Introducing HTML5	3.97	New Riders Publishing

### Task C: Create and execute a materialized view

1. In the tree-view, expand **eBooks > Schemas > public**. Right-click on **Materialized Views** and go to **Create > Materialized View**.

Browser



Dashboard

Properties

SQL

▼ Servers (1)

▼ postgres

▼ Databases (2)

1    ▼ eBooks

&gt; Casts

&gt; Catalogs

&gt; Event Triggers

&gt; Extensions

&gt; Foreign Data Wrappers

&gt; Languages

&gt; Publications

2    ▼ Schemas (1)

3    ▼ public

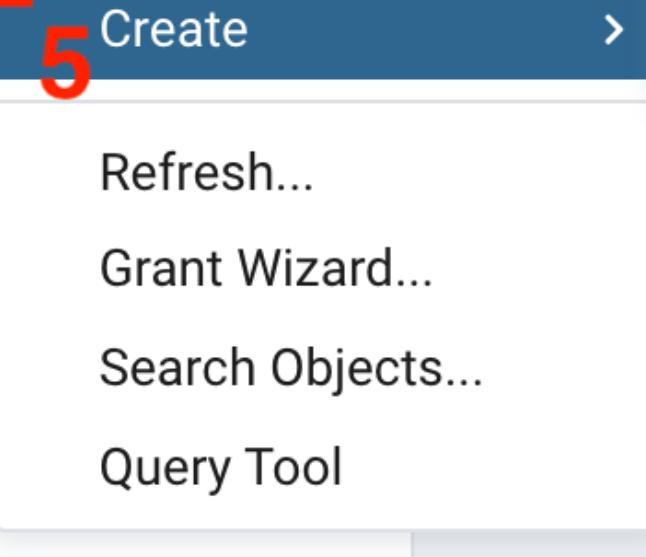
&gt; Collations

&gt; Domains

&gt; FTS Configurations

&gt; FTS Dictionaries

- > FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- 4** **Materialized Views**
- > Procedures
- > 1..3 Sequences
- > Tables (6)
- > Trigger Functions
- > Types
- > Views (1)
- > Subscriptions



2. On the **General** tab, type `publisher_and_rating_materialized_view` as name of the view. Then switch to the **Definition** tab.

## Create - Materialized View

General

Definition

Storage

Parameter

Security

SQL

Name

publisher\_and\_rating\_materialized\_view

Owner

postgres

Schema

public

Comment

 i ?

X Cancel

Rese

3. On the **Definition** tab, copy and paste the code below. Then click **Save**.

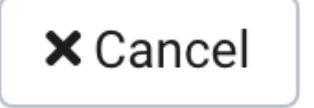
1. 1  
2. 2  
1. SELECT books.title, books.rating, publishers.name  
2. FROM books INNER JOIN publishers ON books.publisher\_id = publishers.publisher\_id

Copied!

## Create - Materialized View

General **Definition** Storage Parameter Security SQL

```
1 SELECT books.title, books.rating, publishers.name  
2 FROM books INNER JOIN publishers ON books.publisher_id = publisher  
3
```

  **Cancel**

4. In the tree-view, expand **Materialized Views**. Right-click on **publisher\_and\_rating\_materialized\_view** and go to **Refresh View > With data**.

- ▼  eBooks
  - >  Casts
  - >  Catalogs
  - >  Event Triggers
  - >  Extensions
  - >  Foreign Data Wrappers
  - >  Languages
  - >  Publications
  - ▼  Schemas (1)
    - ▼  public
      - >  Collations
      - >  Domains
      - >  FTS Configurations
      - >  FTS Dictionaries
      - >  FTS Parsers
      - >  FTS Templates
      - >  Foreign Tables
      - >  Functions

1

- ▼  Materialized Views (1)

...  publisher and rating materialized view

▼ General

Name

OID

Owner

System materia

Comment

▼ Security

Privileges

2  publisher\_and\_rating\_materialized\_view

- >  Columns
- >  Indexes
- >  Procedures
- >  1..3 Sequences
- >  Tables (6)
- >  Trigger Functions
- >  Types
- <span style="color: #800000;">>  Views (1)
- <span style="color: #800000;">>  publisher\_and\_
- >  Columns
- >  Rules
- >  Triggers

>  Subscriptions

Create



Refresh...

Delete/Drop

Drop Cascade

Scripts



3

Refresh View



4

With data



With no data

With data (concur...

With no data (conc...

5. Right-click on **publisher\_and\_rating\_materialized\_view** again and go to **View/Edit Data > All Rows**.

Browser



Dashboard

Pro

- ▼ Databases (2)
  - ▼ eBooks
    - > Casts
    - > Catalogs
    - > Event Triggers
    - > Extensions
    - > Foreign Data Wrappers
    - > Languages
    - > Publications
  - ▼ Schemas (1)
    - ▼ public
      - > Collations
      - > Domains
      - > FTS Configurations
      - > FTS Dictionaries
      - > FTS Parsers
      - > FTS Templates



## ▼ General

Name

OID

Owner

System material

Comment

The screenshot shows a database management interface with a sidebar and a main content area. The sidebar on the right contains sections for Security, Privileges, Storage, Tablespace, and Storage settings. The main content area displays a tree structure of database objects under a schema named 'SCOTT'. The 'Materialized Views' node is expanded, showing one item: 'publisher\_and\_rating\_materialized\_view'. A context menu is open over this item, listing the following options: Create, Refresh..., Delete/Drop, Drop Cascade, Scripts, Refresh View, View/Edit Data, Search Objects..., Query Tool, and Properties... . The 'View/Edit Data' option is highlighted with a blue background. To the right of the menu, there are four buttons: All Rows, First 100 Rows, Last 100 Rows, and Filtered Rows, with 'All Rows' being the active button.

- > Foreign Tables
- > Functions
- < Materialized Views (1)
  - < publisher\_and\_rating\_materialized\_view
    - > Columns
    - > Indexes
  - > Procedures
  - > 1..3 Sequences
  - > Tables (6)
  - > Trigger Functions
  - > Types
  - < Views (1)
    - < publisher\_and\_rating\_materialized\_view
      - > Columns
      - > Rules
      - > Triggers
  - > Subscriptions

Create >

Refresh...

Delete/Drop

Drop Cascade

Scripts >

Refresh View >

**View/Edit Data >**

Search Objects...

Query Tool

Properties...

All Rows

First 100 Rows

Last 100 Rows

Filtered Rows

6. You will access the materialized view you created.

## Query Editor    Query History

```
1 SELECT * FROM public.publisher_and_rating_materialized_view
2
```

## Data Output

## Explain

## Messages

## Notifications

	title character varying (255)	rating numeric (4,2)	name character varying (255)
1	Lean Software Development: ...	4.17	Addison Wesley
2	Facing the Intelligence Explosi...	3.87	Machine Intelligence Researc...
3	Scala in Action	3.74	Manning
4	Patterns of Software: Tales fr...	3.84	Oxford University Press, USA
5	Anatomy Of LISP	4.43	McGraw-Hill
6	Computing machinery and int...	4.17	MSAC Philosophy Group
7	XML: Visual QuickStart Guide	3.66	Peachpit Press
8	SQL Cookbook	3.95	O'Reilly Media
9	The Apollo Guidance Comput...	4.29	Praxis Publications Inc

10	Minds and Computers: An Intr...	3.54	Edinburgh University Press
11	The Architecture of Symbolic ...	4.50	McGraw-Hill
12	Nmap Network Scanning: The...	4.32	Nmap Project
13	The It Handbook for Business:...	4.40	Createspace Independent Pub...
14	Accidental Empires	4.00	Harper
15	Introducing HTML5	3.97	New Riders Publishing

As you can see, at first glance it doesn't look too different from the regular view you created earlier in this lab - indeed, from the user perspective it's essentially the same: you see the results of a query displayed in a table-like format. The difference is that this materialized view is cached in the database so that the data can be accessed again at a future time without having to re-run the database query, which can be intensive on the server depending on the complexity of the query and the size of the table being queried.

**Congratulations! You have completed this lab, and you are ready for the next topic.**

## Author

- [Sandip Saha Joy](#)

## Other Contributors

- [David Pasternak](#)

## Changelog

Date	Version	Changed by	Change Description
2021-03-25	1.0	Sandip Saha Joy	Created initial version
2021-10-18	1.1	David Pasternak	Updated instructions

