



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.

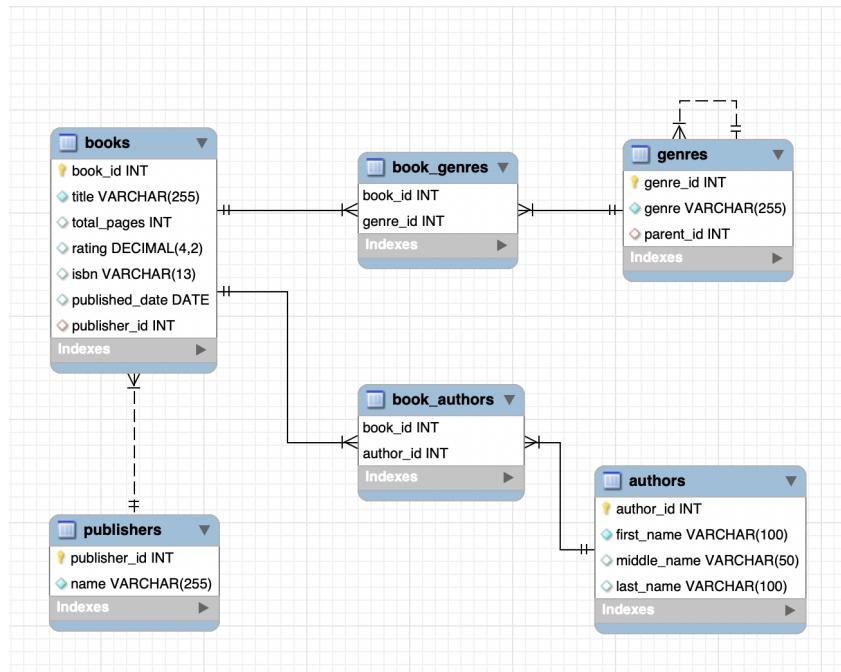


To complete this lab you will utilize the PostgreSQL relational database service available as part of IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs is a virtual lab environment used in this course.

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.
 - o [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.

The screenshot shows the IBMCloud Skills Network interface. On the left, there's a sidebar with various icons and sections like 'Lab', 'IBMCloud', and 'Launch Application'. The main area is titled 'PostgreSQL' and shows the following steps:

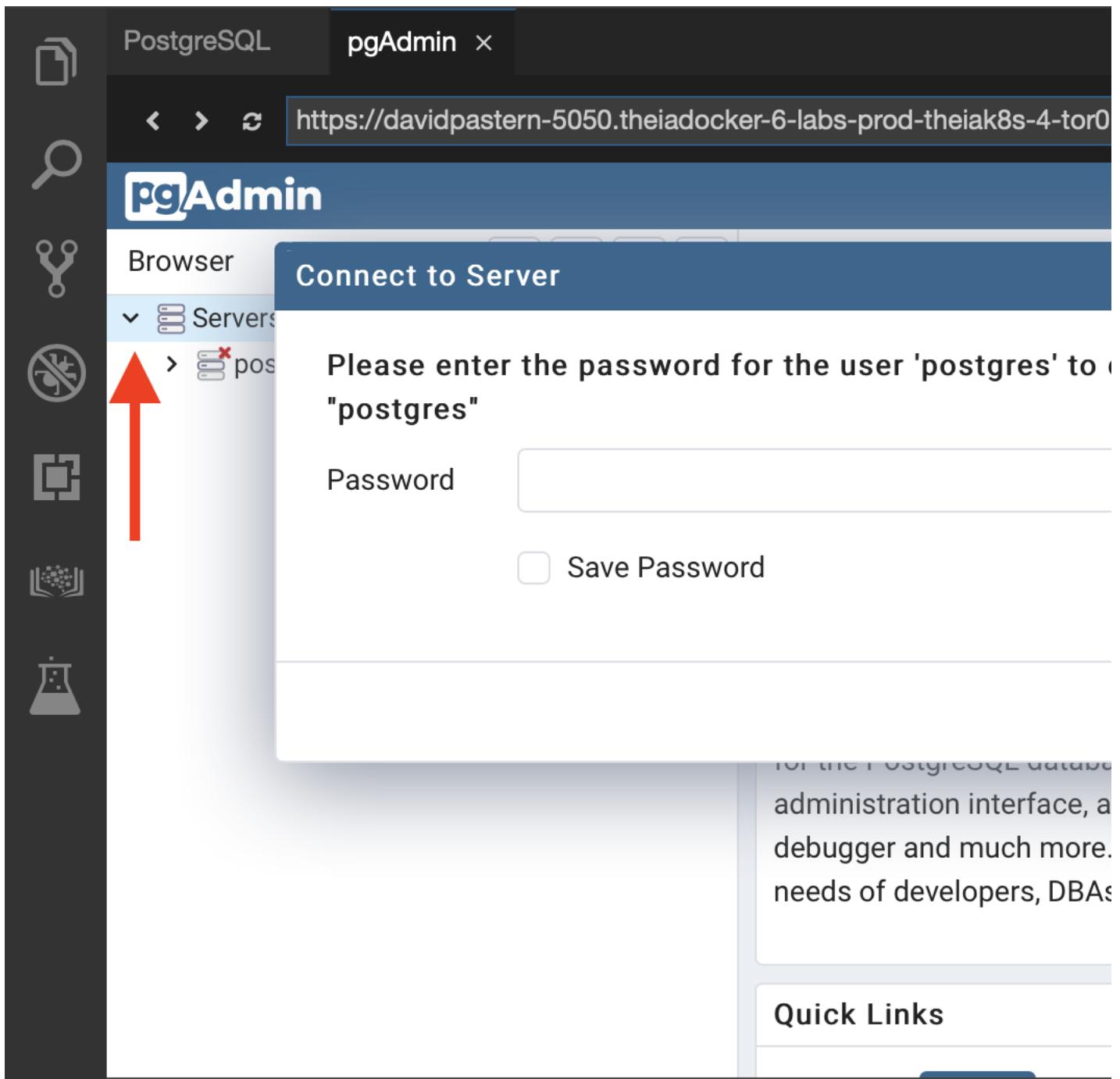
- 1**: A red circle highlights the 'Start' button for the PostgreSQL service.
- 2**: A red circle highlights the 'PostgreSQL' service entry in the 'DATABASES' section of the sidebar.
- 3**: A red circle highlights the 'Start' button for the PostgreSQL service in the main summary view.

The PostgreSQL service is currently listed as 'INACTIVE'. Below the service details, there are tabs for 'Summary', 'Connection Information', and 'Details', with 'Summary' being the active tab. A note at the bottom encourages users to get started with Postgres in a faster, easier way.

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

The screenshot shows the PostgreSQL setup page in pgAdmin. On the left, there's a vertical sidebar with icons for file operations, search, and database management. The main area has a dark header with the title "PostgreSQL" and a "Stop" button. Below the header, there are three tabs: "Summary" (which is selected), "Connection Information", and "Details". A message in the "Summary" tab says: "Your database and pgAdmin server are now ready to use and available with the on how to navigate PostgreSQL, please check out the Details section." It includes fields for "Username" and "Password" with copy/cut icons. Below these fields, it says "You can manage PostgreSQL via:" followed by two buttons: "pgAdmin" (which is highlighted with a red box) and a copy icon. Further down, it says "Or to interact with the database in the terminal, select one of these options:" with two more buttons: "PostgreSQL CLI" and "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.



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

PostgreSQL

ACTIVE

v13.2 | v5.0 | 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 IP address 10.0.0.11. For instructions on how to navigate PostgreSQL, please check out the Details section.

Username:

Password: **2**

You can manage PostgreSQL via:

pgAdmin **SSH**

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.

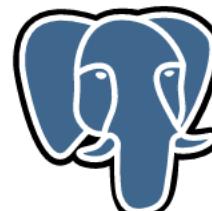
← → ⌛ ⌂ sandipsahajo-5050.theiadocker-27.proxy.cognitivecl

pgAdmin File ▾ Object ▾ Tools ▾ Help ▾

Browser     Dashboard Properties SQL

>  Servers

Welcome



pgAdn
Management

Feature rich | Maximise

pgAdmin is an Open Source admir
is designed to answer the needs o

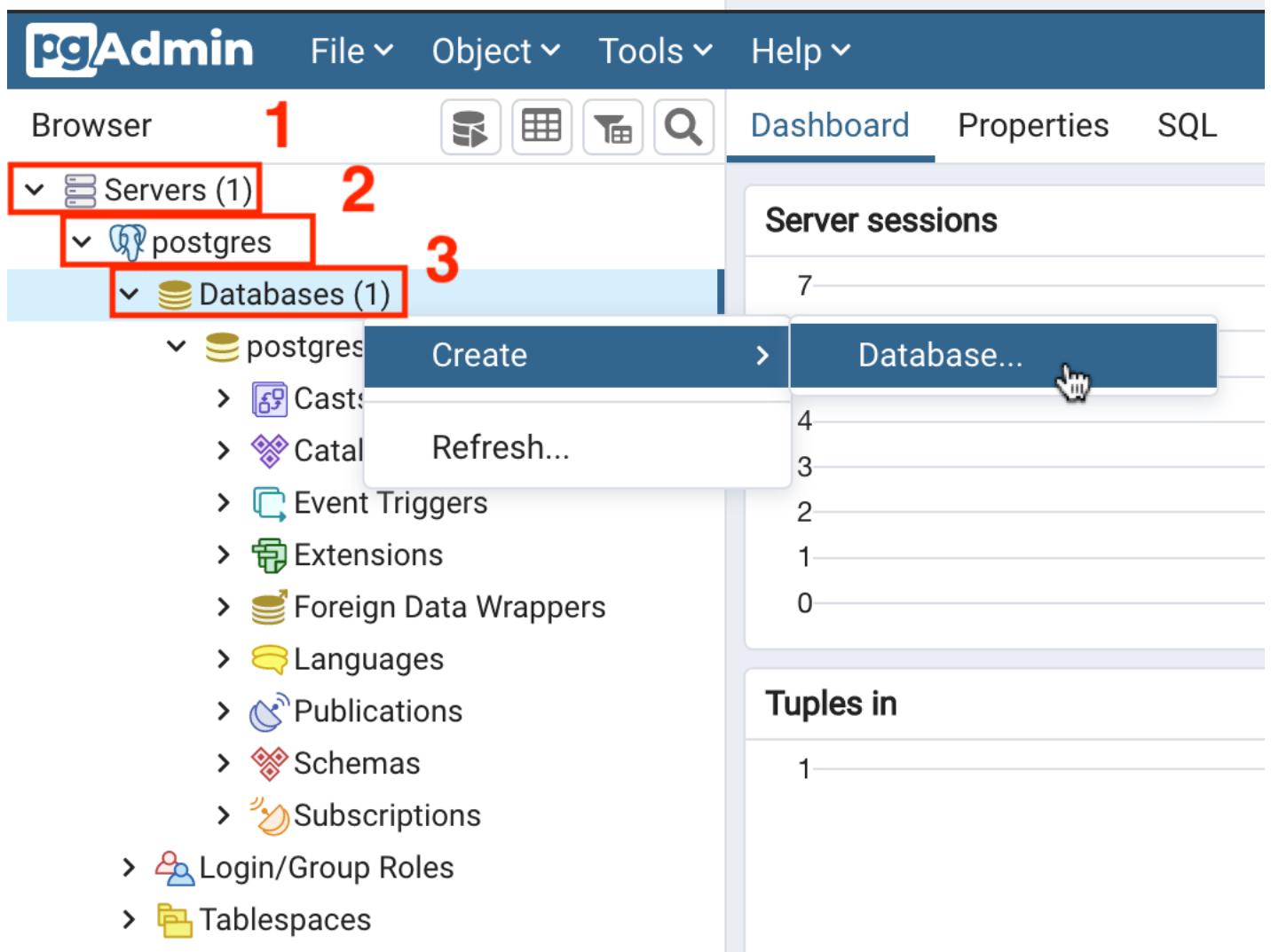
Quick Links

Getting Started



PostgreSQL Document

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**.



Create - Database

General Definition Security Parameters Advanced SQL

Database

eBooks

Owner

postgres

Comment



X Cancel

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

The screenshot shows the pgAdmin interface. The left sidebar displays a tree structure of servers and databases. Under the 'Servers' node, there is one entry for 'postres'. Under 'postres', there are two entries: 'Databases' (containing 'eBooks') and 'postres'. The 'eBooks' database is highlighted with a red box. A context menu is open over the 'eBooks' entry, listing various options: Create, Refresh..., Delete/Drop, CREATE Script, Disconnect Database..., Generate ERD (Beta), Maintenance..., Backup..., Restore..., Grant Wizard..., Search Objects..., Query Tool, Properties..., Publications, Schemas, and Subscriptions. The 'Restore...' option is highlighted with a blue background and a cursor icon pointing at it.

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...



- o Click the **Upload File** button.

Select file

/var/lib/pgadmin/

Name	Size
sessions	4.0 kB
storage	4.0 kB

Show hidden files and folders?

- 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

Show hidden files and folders?

- 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

Show hidden files and folders?

- 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
eBooks_pgsql_dump.tar	74.2 kB
pgadmin4.db	156.0 kB
sessions	4.0 kB
storage	4.0 kB

Show hidden files and folders?

- 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

Role name

Select an item...



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

Restore (Database: eBooks)

General **Restore options**

Queries

Include CREATE DATABASE statement

Clean before restore

Single transaction

Disable

Trigger

No data for Failed Tables

i **?**

Include CREATE DATABASE statement	<input type="button" value="No"/>
Clean before restore	<input type="button" value="No"/>
Single transaction	<input type="button" value="No"/>
Trigger	<input checked="" type="button" value="Yes"/>
No data for Failed Tables	

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.

pgAdmin File ▾ Object ▾ Tools ▾ Help ▾

Browser Dashboard Properties SQL

Servers (1) postres Databases (2)

1 eBooks Casts Catalogs Event Triggers Extensions Foreign Data Wrappers Languages Publications

2 Schemas (1) public

3 Collations Domains FTS Configurations FTS Dictionaries FTS Parsers FTS Templates Foreign Tables Functions Materialized Views Procedures Sequences Tables (6) Trigger Functions Types

4 Views Subscriptions

5 Create 6 View... Refresh...

postgres

18/30

2. On the **General** tab, type **publisher_and_rating_view** as name of the view. Then switch to **Code** tab.

Create - View

General Definition Code Security SQL

Name: publisher_and_rating_view

Owner: postgres

Schema: public

Comment:

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!

Create - View

General Definition **Code** Security SQL

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



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

pgAdmin File ▾ Object ▾ Tools ▾ Help ▾

Browser Dashboard Properties

Servers (1) postres 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 Foreign Tables Functions Materialized Views Procedures Sequences Tables (6) Trigger Functions Types Views (1) publisher_and_rating_view Columns Rules

1 2

Database sessions
1

Tuples in
18
16
14
12
10
8
6
4
2
0

Server activity
Sessions Locks

PID
83

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

[Refresh...](#)



public.publisher_and_rating_view/eBooks/postgres@postgres

Query Editor

Query History

```
1  SELECT * FROM public.publisher_and_rating_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

Task C: Create and execute a materialized view

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

pgAdmin File ▾ Object ▾ Tools ▾ Help ▾

Browser

Dashboard Properties

Servers (1)
 postgres
 Databases (2)
 1 eBooks (highlighted)
 Casts
 Catalogs
 Event Triggers
 Extensions
 Foreign Data Wrappers
 Languages
 Publications

2 Schemas (1) (highlighted)
 3 public (highlighted)
 Collations
 Domains
 FTS Configurations
 FTS Dictionaries
 FTS Parsers
 FTS Templates
 Foreign Tables
 Functions

4 Materialized Views (highlighted)
 Procedures
 Sequences
 Tables (6)
 Trigger Functions
 Types
 Views (1)
 Subscriptions

5 Create
 Refresh...
 Grant Wizard...
 Search Objects...
 Query Tool

6 Materialized Views (highlighted)

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 **?** **Cancel**

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 = pu  
3
```



× Cal

4. In the tree-view, expand **Materialized Views**. Right-click on **publisher_and_rating_materialized_view** and go to **Refresh View > With data**.

The screenshot shows a PostgreSQL pgAdmin interface. On the left, a tree view displays database objects under 'eBooks'. A context menu is open over a 'Materialized Views' node, with several items highlighted by red numbers:

- 1** **Materialized Views (1)**
- 2** **publisher_and_rating_materialized_view**
- 3** Refresh View
- 4** With data

The context menu items are:

- Create >
- Refresh...
- Delete/Drop
- Drop Cascade
- Scripts >
- 3 Refresh View > **4 With data**
- View/Edit Data > With no data
- Search Objects... > With data
- Query Tool > With no data
- Properties...

On the right, a vertical pane lists database schema details:

- Gene
- Name
- OID
- Owner
- System
- Comm
- Secu
- Privileg
- Stora
- Tables
- Storage

5. Right-click on **publisher_and_rating_materialized_view** again and go to **View/Edit Data > All Rows**.

pgAdmin File ▾ Object ▾ Tools ▾ Help ▾

Browser

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
 - Foreign Tables
 - Functions
 - Materialized Views (1)
 - publisher_and_rating (materialized view)
 - Columns
 - Indexes
 - Procedures
 - Sequences
 - Tables (6)
 - Trigger Functions
 - Types
 - Views (1)
 - publisher_and_rating (view)
 - Columns

6. You will access the materialized view you created.

COLUMNS



public.publisher_and_rating_materialized_view/eBooks/postgres@postgres

Query Editor Query History

```

1  SELECT * FROM public.publisher_and_rating_materialized_vie
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

© IBM Corporation 2021. All rights reserved.