Hands-on Lab: Views in PostgreSQL



Estimated time needed: 15 minutes

In this lab, you will learn how to create, execute, and materialize views in the PostgreSQL database service using the pgAdmin graphical user interface (GUI) tool. Materialized views behave differently compared to regular views. The result set is materialized or saved for future use in the materialized views. You can not insert, update, or delete rows like in regular views. Materialized views store the results of a database query as a separate table-like object so that someone can access the results later 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 <u>PostgreSQL Database</u>. PostgreSQL is a relational database management system (RDBMS) designed to store, manipulate, and retrieve data efficiently.

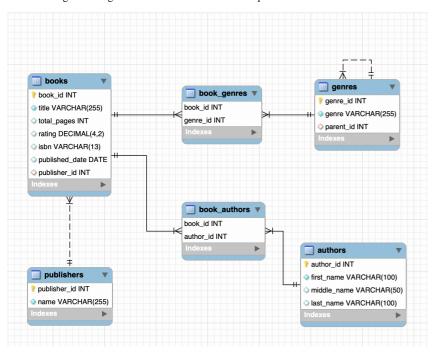


To complete this lab, you will utilize the PostgreSQL relational database service available as part of IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs is a virtual lab environment used in this course.

Database used in this lab

You will use the eBooks database in the 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 to learn how to create and execute views and materialized views in the PostgreSQL database service using the pgAdmin graphical user interface (GUI) tool.

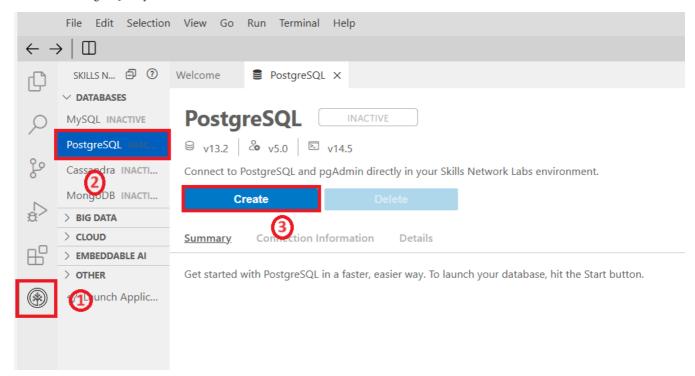
Task A: Restore a database schema and data

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:

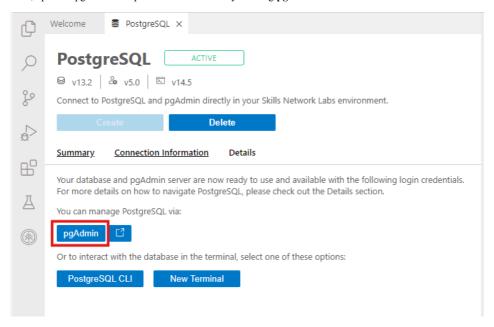
about:blank 1/22

1. Download the following eBooks PostgreSQL dump file (containing the eBooks database schema and data) to your local computer.

- o eBooks pgsql dump.tar
- 2. Click the Skills Network extension button on the left side of the window.
- 3. Open the DATABASES menu and click PostgreSQL.
- 4. Click Create. PostgreSQL may take a few moments to start.

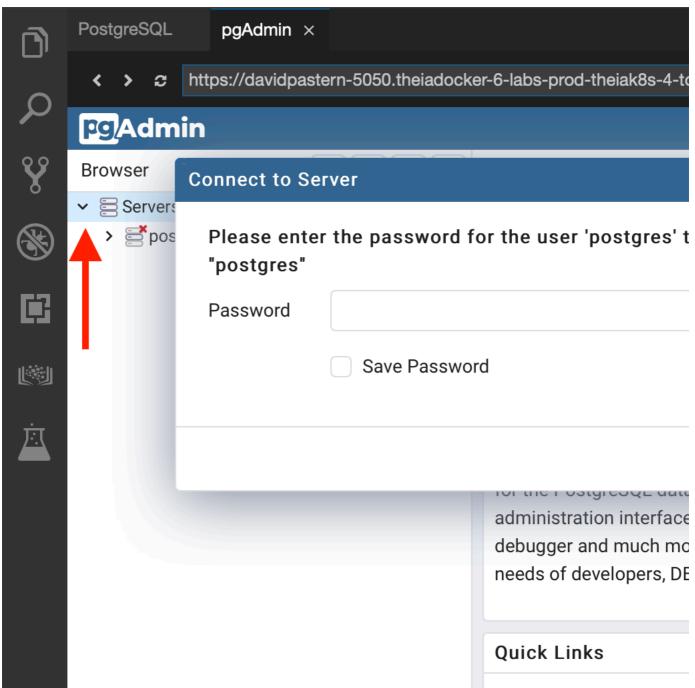


5. Next, open the pgAdmin Graphical User Interface by clicking **pgAdmin** in the Cloud IDE interface.



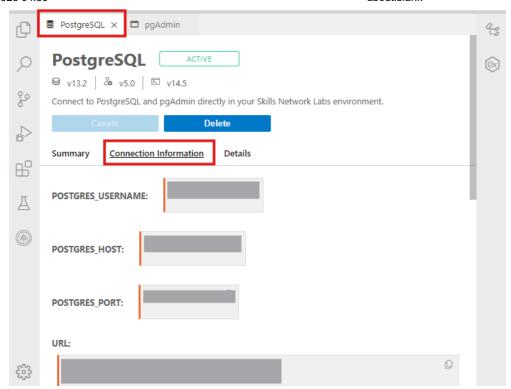
6. Once the pgAdmin GUI opens, click Servers tab on the left side of the page. You will be prompted to enter a password.

about:blank 2/22

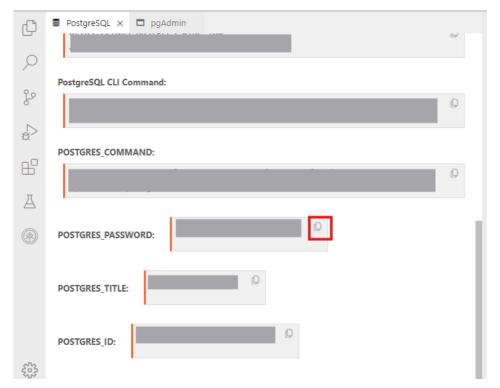


7. To retrieve your password, click PostgreSQL tab near the top of the interface and select Connection Information tab.

about:blank 3/22

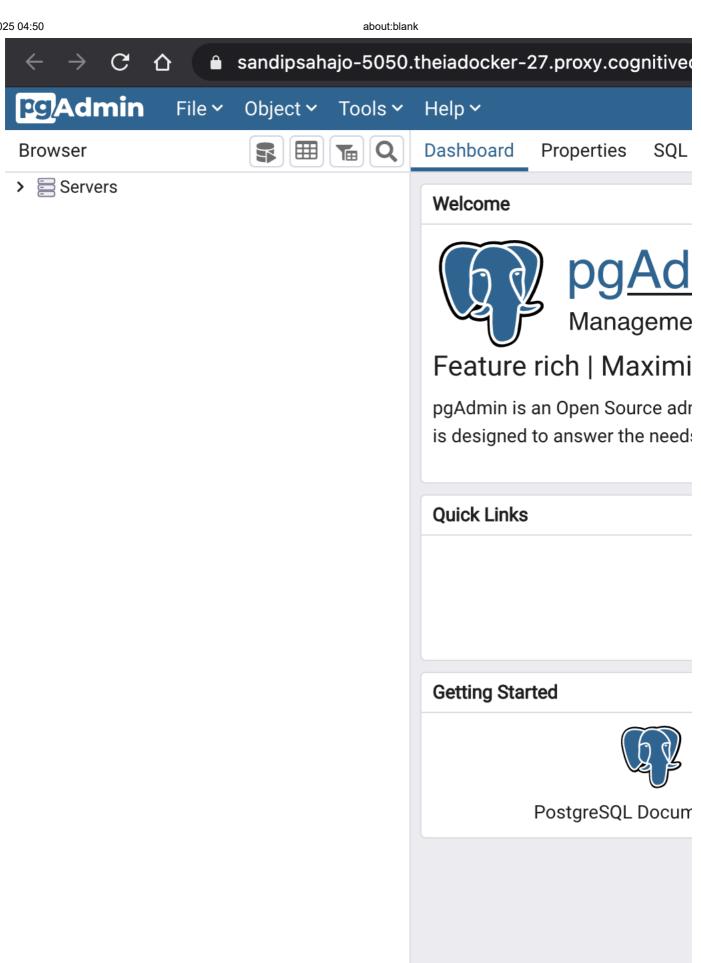


8. Scroll down and click the Copy icon on the left of your password to copy the session password onto your clipboard.



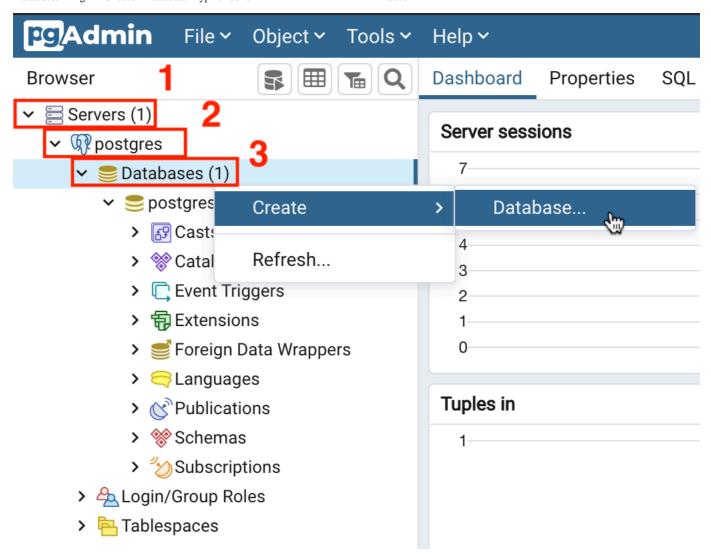
- 9. Navigate back to the **pgAdmin** tab and paste your password, then click **OK**.
- 10. You will then be able to access the pgAdmin GUI tool.

4.01.2025 04:50

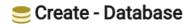


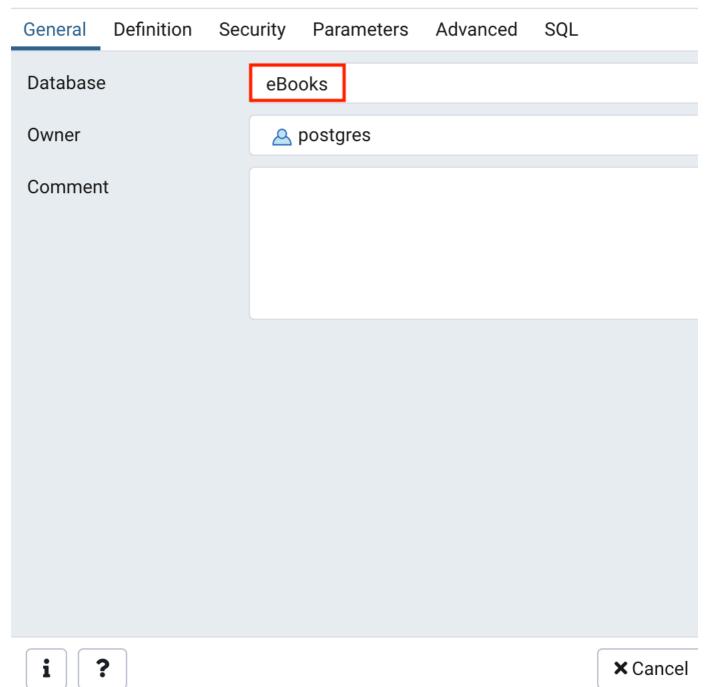
5/22 about:blank

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 the database name and click Save.



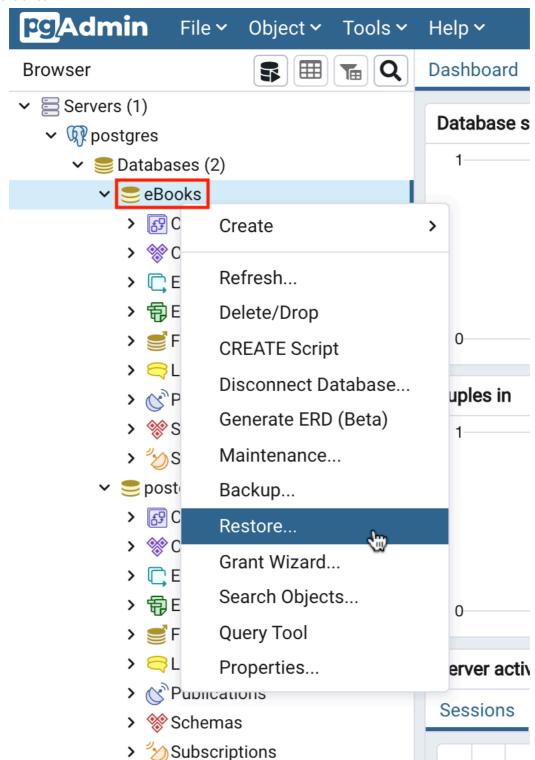
about:blank 6/22





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

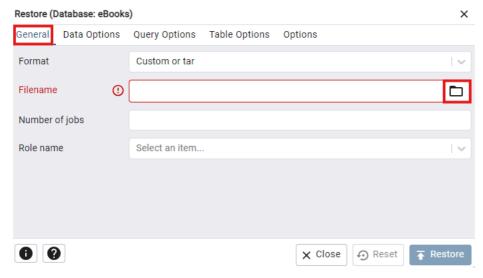
about:blank 7/22



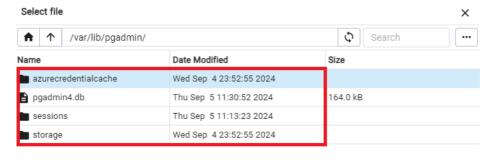
13. Follow the instructions below to restore and proceed to Task B:

about:blank 8/22

[•] On the General tab, click Select file by the Filename box.

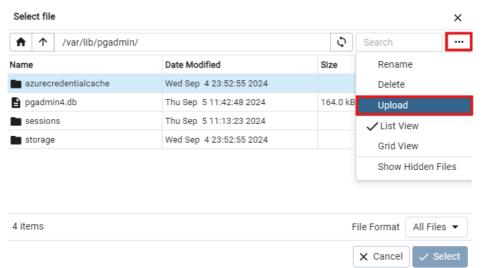


• Ensure that you upload the files to this path: /var/lib/pgadmin/. To do this, you can either manually navigate to the path (or) copy /var/lib/pgadmin/, replace /home/ with it, and press Enter. You should then see some default files in that path, as shown below.



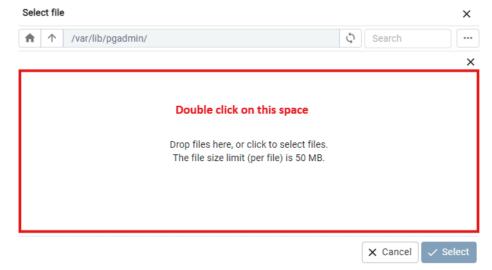


• Click on the three dots, then select Upload.

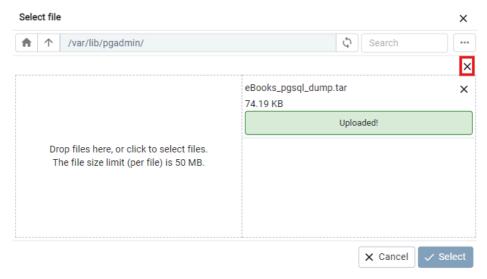


o Double-click on the drop files area and load the eBooks_pgsql_dump.tar you downloaded earlier on your local computer.

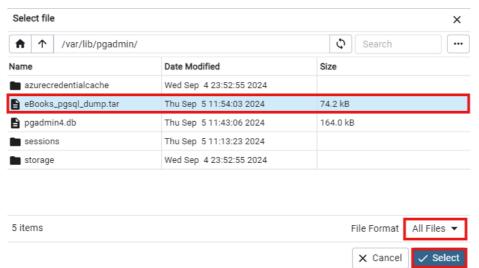
about:blank 9/22



 $\circ~$ When the upload is complete, close the drop files area by clicking $\boldsymbol{X}.$

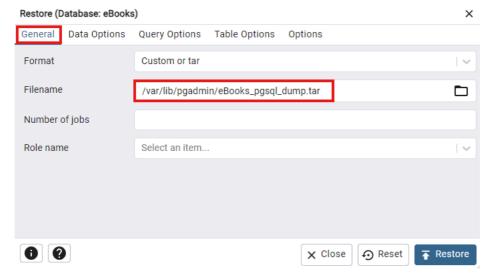


 $\circ \ \ Ensure \ \textbf{Format} \ is \ set \ to \ \textbf{All Files}, \ select \ the \ uploaded \ \textbf{eBooks_pgsql_dump.tar} \ file \ from \ the \ list, \ and \ then \ click \ \textbf{Select}.$

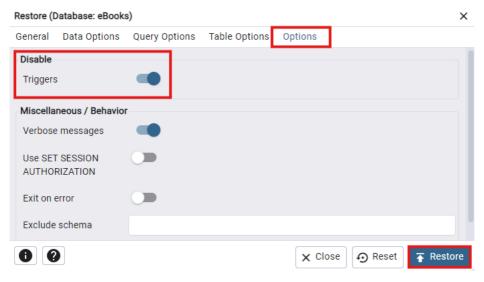


o In the General tab, ensure the filename path matches the one shown below. If you see a different path that includes "None," modify it accordingly.

about:blank 10/22



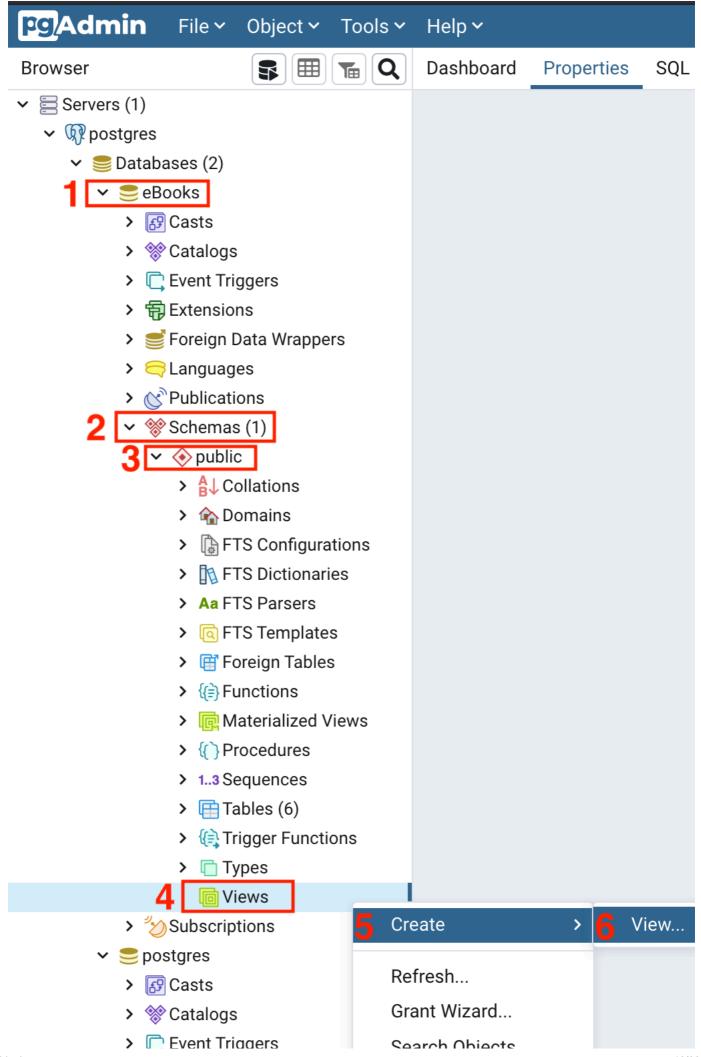
• Now switch to the **Options** tab. Under **Disable**, toggle on the **Triggers** option, and then click Restore.



Task B: Create and execute a view

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

about:blank 11/22



ocaron objects...

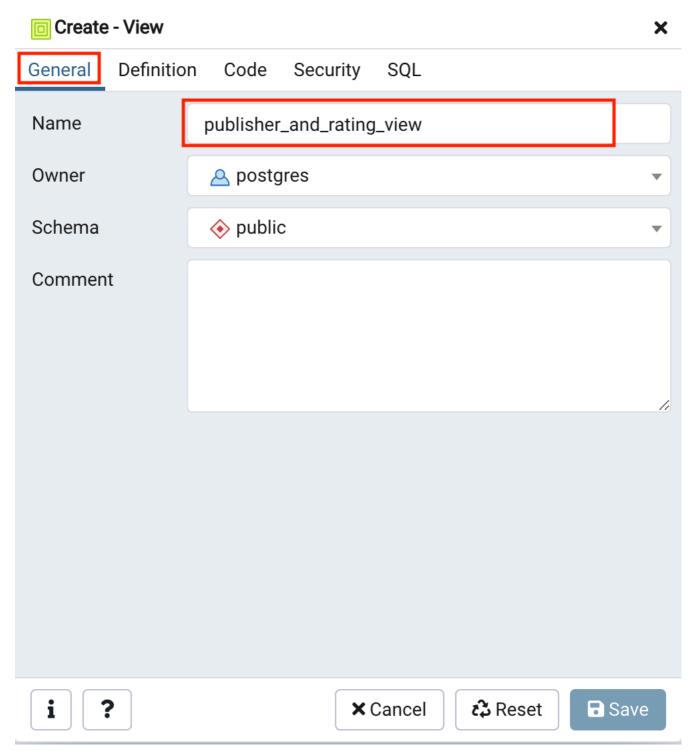
Query Tool







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

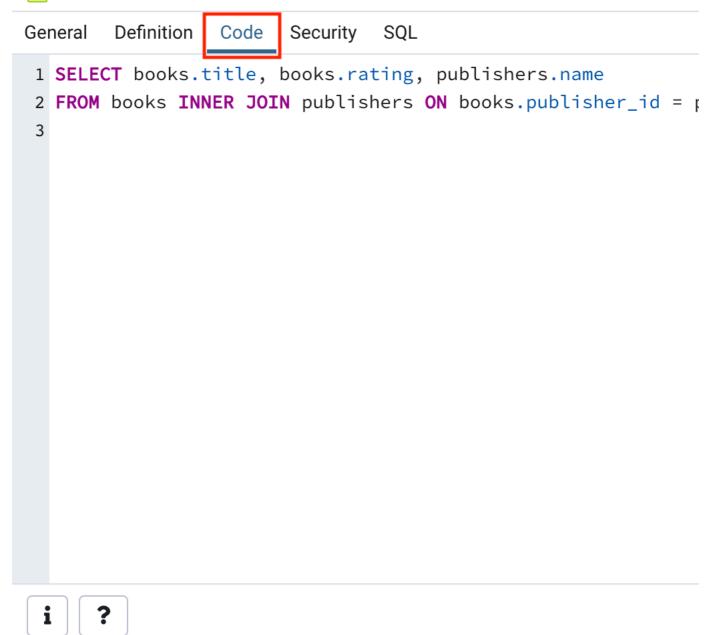


3. On the Code tab, copy and paste the following code. Then click Save.

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

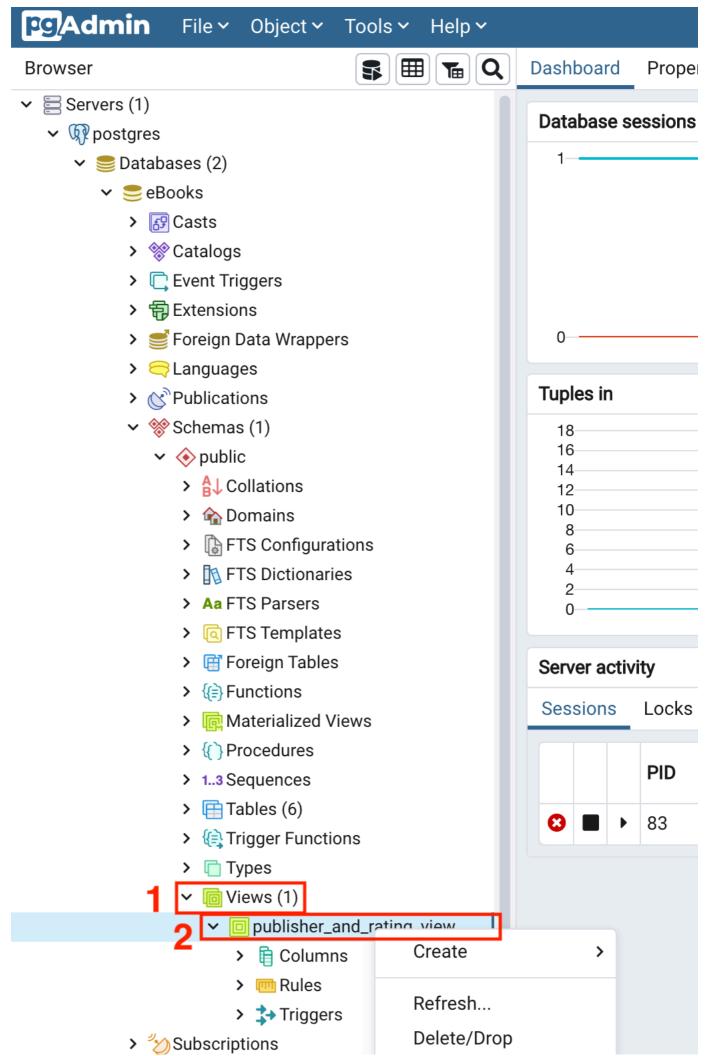
about:blank 13/22

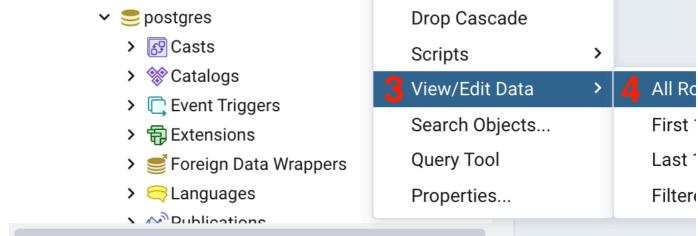




4. In the tree view, expand Views. Right-click publisher_and_rating_view and go to View/Edit Data > All Rows.

about:blank 14/22





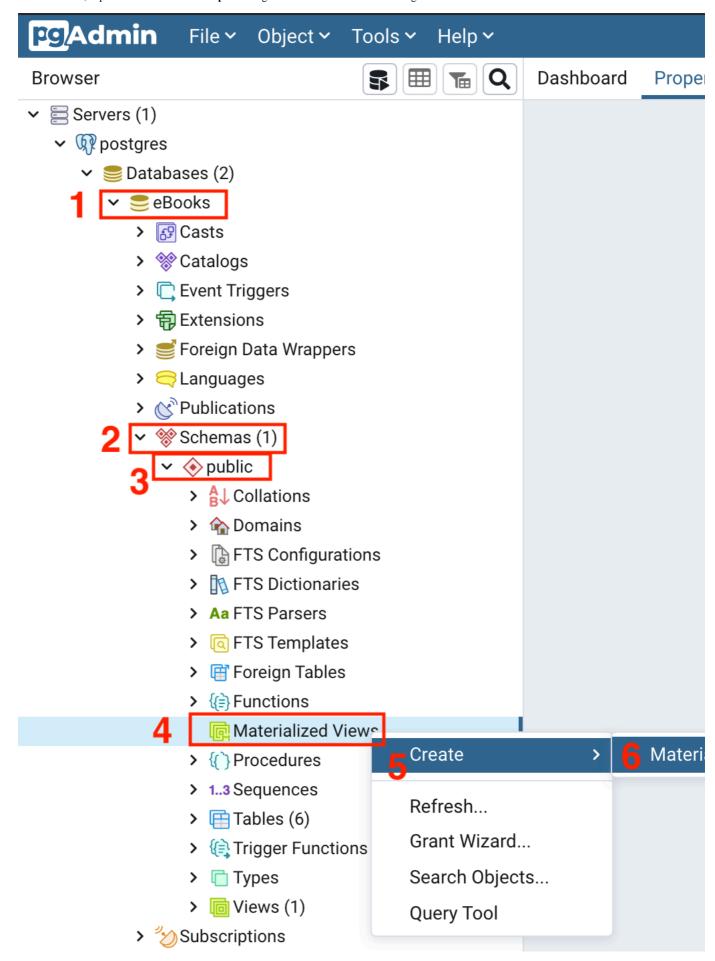
5. You will access the view you created. This action allows you to access and view the tables in your database.

Ŕ public.publisher_and_rating_view/eBooks/postgres@postgres Query Editor Query History **SELECT** * **FROM public**.publisher_and_rating_view 1 2 **Notifications** Data Output Explain Messages title rating name character varying (255) numeric (4,2) character varying (255) 1 Lean Software Development: ... 4.17 Addison Wesley 2 Facing the Intelligence Explosi... 3.87 Machine Intelligence Researc 3.74 3 Scala in Action 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 SOL Cookbook O'Reilly Media 3.95 9 The Apollo Guidance Comput... 4.29 Praxis Publications Inc 10 Minds and Computers: An Intr... 3.54 **Edinburgh University Press** 4.50 McGraw-Hill 11 The Architecture of Symbolic ... 12 Nmap Network Scanning: The... 4.32 **Nmap Project** 13 The It Handbook for Business:... 4.40 Createspace Independent Pul 14 **Accidental Empires** 4.00 Harper 15 Introducing HTML5 3.97 New Riders Publishing

about:blank 16/22

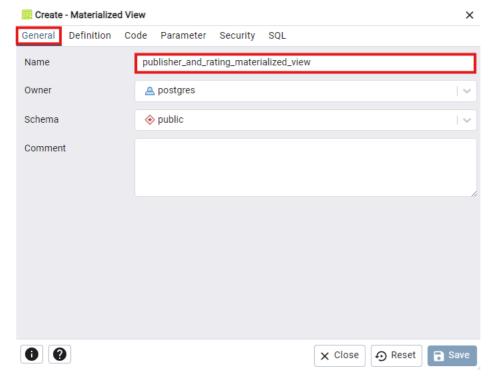
Task C: Create and execute a materialized view

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



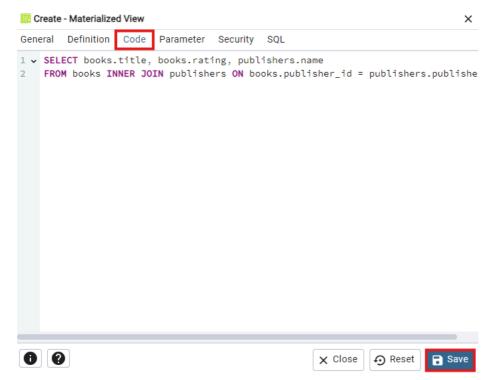
2. On the General tab, type publisher_and_rating_materialized_view as name of the view. Then switch to the Code tab.

about:blank 17/22



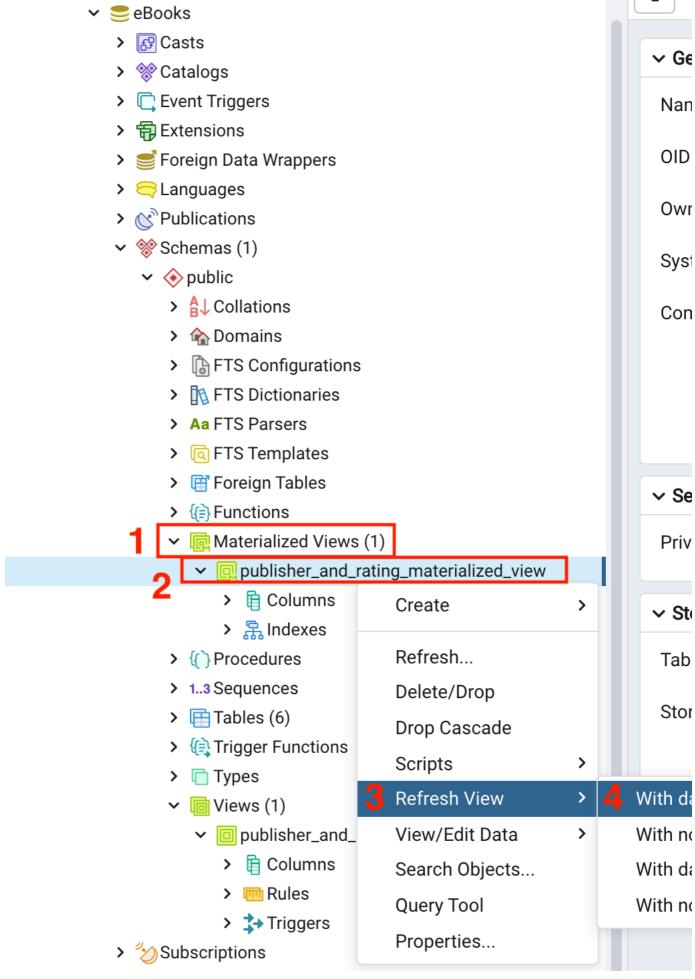
3. On the code tab, copy and paste the following code. Then click Save.

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



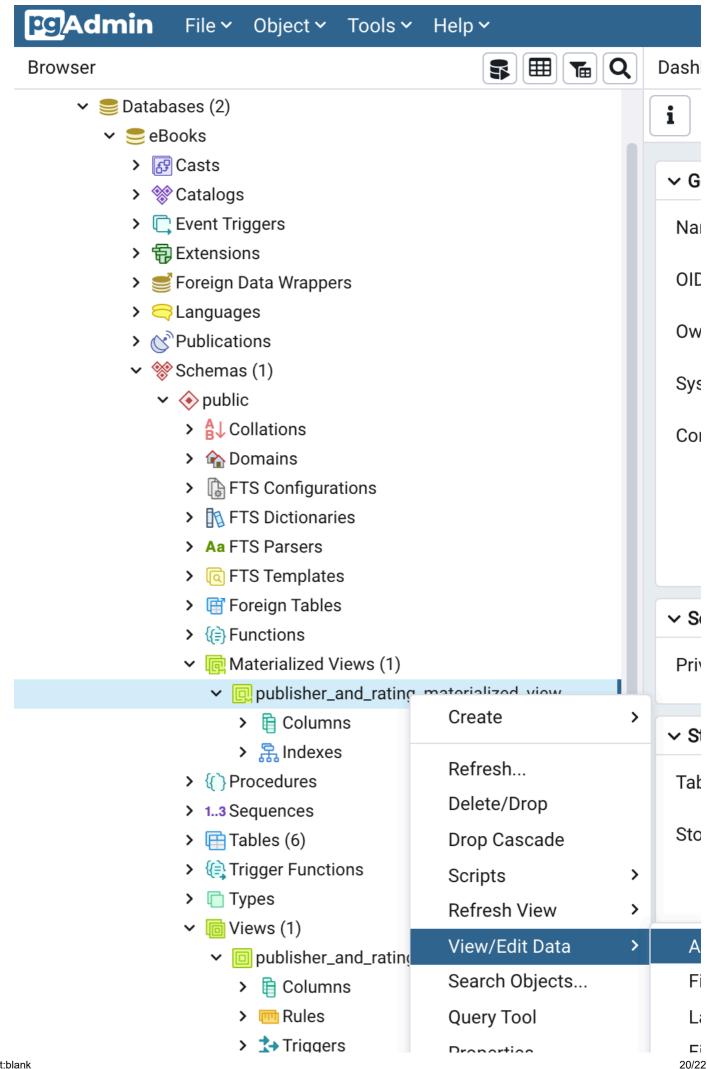
 $4. \ In the tree-view, expand \ \textbf{Materialized Views}. \ Right-click \ \textbf{publisher_and_rating_materialized_view} \ and \ go \ to \ \textbf{Refresh View} > \textbf{With data}.$

about:blank 18/22



^{5.} Right-click publisher_and_rating_materialized_view again and go to View/Edit Data > All Rows.

about:blank 19/22





6. You will access the materialized view you created.

public.publisher_and_rating_materialized_view/eBooks/postgres@postgres

Query Editor Query History

1 SELECT * FROM public.publisher_and_rating_materialized_v

2

Data Output		Explain Messages Notifications				
	title charac	eter varying (2	255)	rating numeric (4,2)	<u> </u>	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

At first glance, it does not look too different from the regular view you created earlier in this lab. From the user perspective, it is 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 someone can reaccess the data in the future without re-running the database query.

Conclusion

Congratulations! You have completed this lab and learned how to restore a database schema and data, create and execute a view, and create and execute a materialized view.

Author: Sandip Saha Joy

Other Contributors: **David Pasternak**

about:blank 21/22



© IBM Corporation. All rights reserved.

about:blank 22/22