

All T20 Internationals Dataset (2005 - 2023)

Database Access Control: Analysts, Operators, and Administrators | Unit 7

Avinash Bunga

Information Systems and Business Analytics, Park University CIS622DLAF2P2023

Data Architecture for Business Analytics Professor: Gulnoza Khakimova

Dec 3, 2023

Database Access Control: Analysts, Operators, and Administrators

Introduction:

This document outlines the creation and configuration of user roles within the "All_T20_Internationals_Dataset_2005_2023" database, detailing the rationale for each role and the steps taken to establish the necessary permissions within pgAdmin.

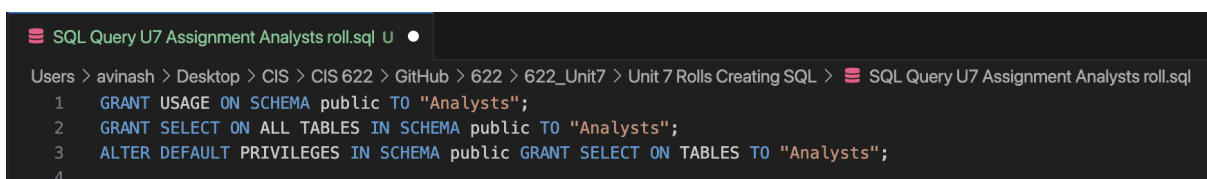
Overview and Step-by-Step Role Configuration Process:

Before delving into the specific queries and screenshots, it is essential to understand the general process undertaken to configure user roles within the pgAdmin 4 environment. This process involves establishing secure connections to the database, accessing the appropriate management tools within pgAdmin, and executing structured query language (SQL) commands that define and restrict user access based on predetermined roles. Each role is crafted to align with the responsibilities and requirements of the user groups interacting with the database, ensuring an optimal balance between operational functionality and data security. For detailed queries and visual walkthroughs from pgAdmin 4, proceed to the subsequent sections.

Group Definition and Access Rationale:

1. Analysts

- **Reason for Access:** Analysts are granted read-only access to all database tables to perform data analysis, compile statistics, and generate reports that may influence strategic decisions.
- **Group Members:** This group typically includes team statisticians, data analysts, and strategic planners.



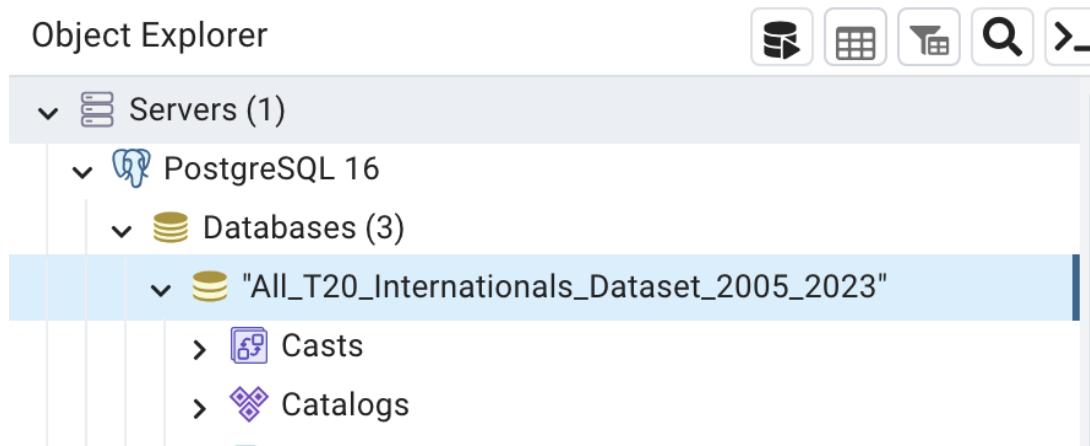
```
SQL Query U7 Assignment Analysts roll.sql U •
Users > avinash > Desktop > CIS > CIS 622 > GitHub > 622 > 622_Unit7 > Unit 7 Rolls Creating SQL > SQL Query U7 Assignment Analysts roll.sql
1 GRANT USAGE ON SCHEMA public TO "Analysts";
2 GRANT SELECT ON ALL TABLES IN SCHEMA public TO "Analysts";
3 ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT SELECT ON TABLES TO "Analysts";
4
```

[SQL Link](#)

To run the specified queries in pgAdmin, follow these steps:

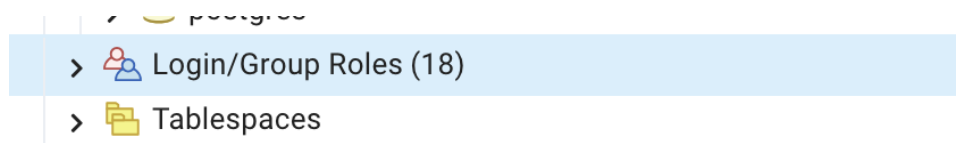
1. Open pgAdmin and Connect to the Database:

Open pgAdmin, then connect to your PostgreSQL server. Navigate to your "All_T20_Internationals_Dataset_2005_2023" database.



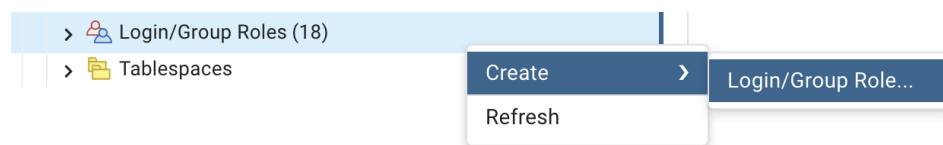
2. Navigate to Login/Group Roles:

- Expand the 'Login/Group Roles' tree under your server.

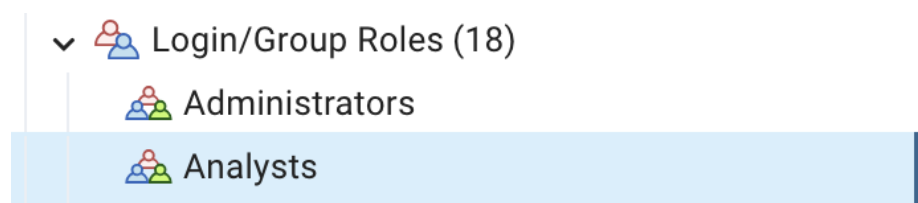


3. Create the Administrators Role:

- Right-click on 'Login/Group Roles' and select 'Create' > 'Login/Group Role'.

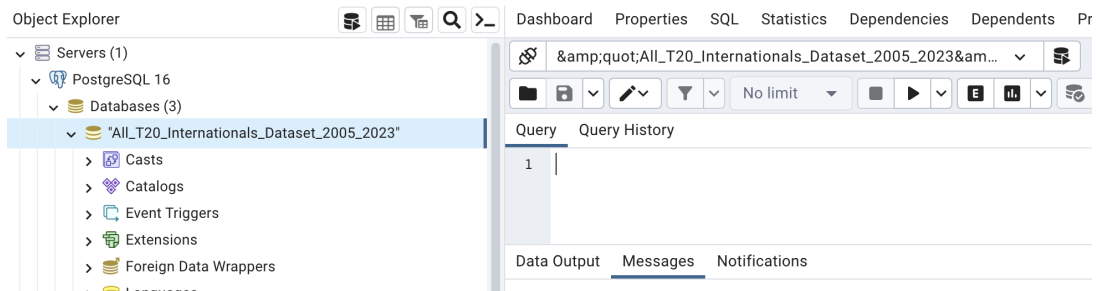


- Name the role "Analysts".



4. Open the Query Tool:

Right-click on the database name and select 'Query Tool' from the context menu to open a new query editor window.



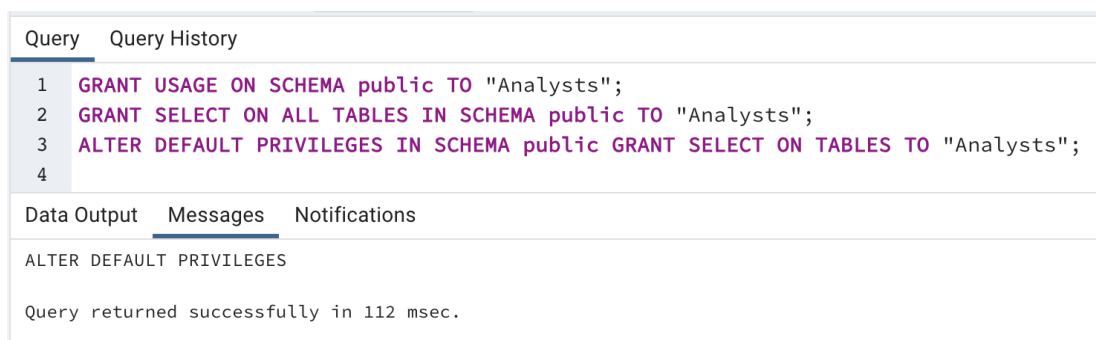
5. Enter the Grant Commands:

In the query editor, enter the following [SQL statements](#):

```
GRANT USAGE ON SCHEMA public TO "Analysts";
GRANT SELECT ON ALL TABLES IN SCHEMA public TO "Analysts";
ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT SELECT ON TABLES TO
"Analysts";
```

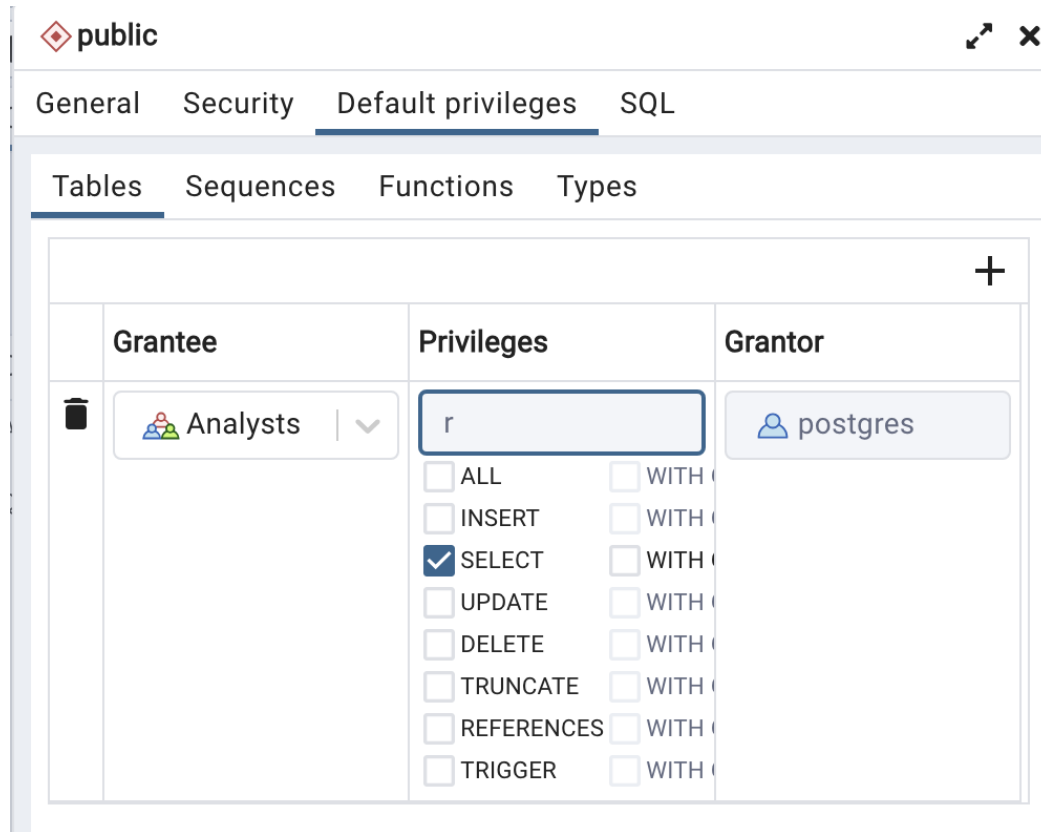
6. Execute the Query:

Click the 'Execute/Run' button to run the commands and apply the permissions.



7. Verify the Role and Privileges:

You can verify the permissions have been set by looking at the Security tab for the tables within the 'public' schema.



2. Data Entry Operators

- Reason for Access:** Data Entry Operators have insert and update privileges on tables related to match data, player statistics, and other time-sensitive information to ensure the database remains up-to-date.
- Group Members:** Members typically consist of administrative staff responsible for recording match outcomes, player performance, and other relevant data.

```

Data Entry Operators Roll.sql U
Users > avinash > Desktop > CIS > CIS 622 > GitHub > 622 > 622_Unit7 > Unit 7 Rolls Creating SQL > Data Entry Operators Roll.sql
1 GRANT INSERT, UPDATE ON ALL TABLES IN SCHEMA public TO "Data Entry Operators";
2

```

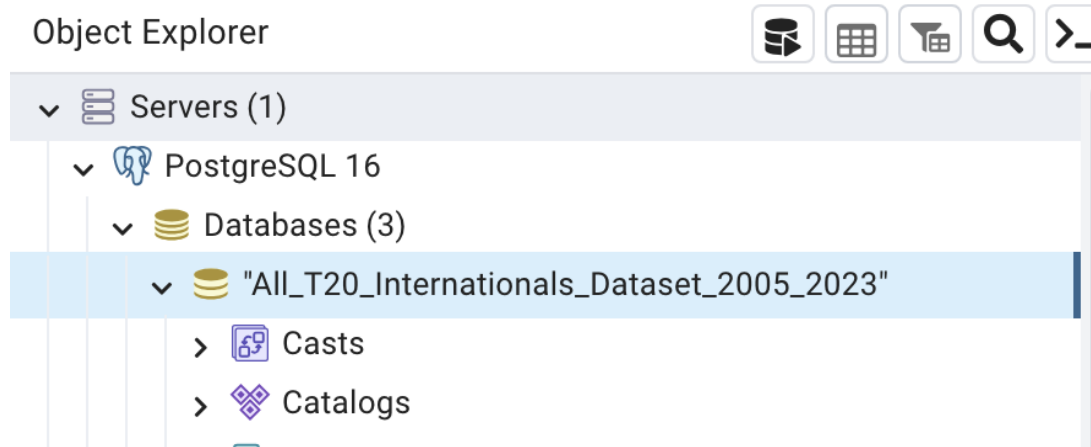
[SQL Link](#)

To run the specified queries in pgAdmin, follow these steps:

1. Open pgAdmin and Connect to the Database:

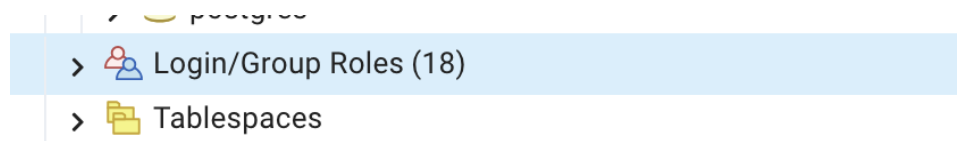
Open pgAdmin, then connect to your PostgreSQL server. Navigate to your

"All_T20_Internationals_Dataset_2005_2023" database.



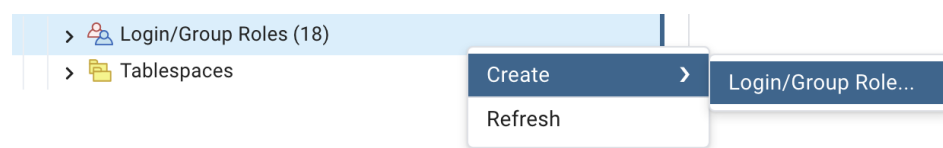
2. Navigate to Login/Group Roles:

- Expand the 'Login/Group Roles' tree under your server.



3. Create the Administrators Role:

- Right-click on 'Login/Group Roles' and select 'Create' > 'Login/Group Role'.

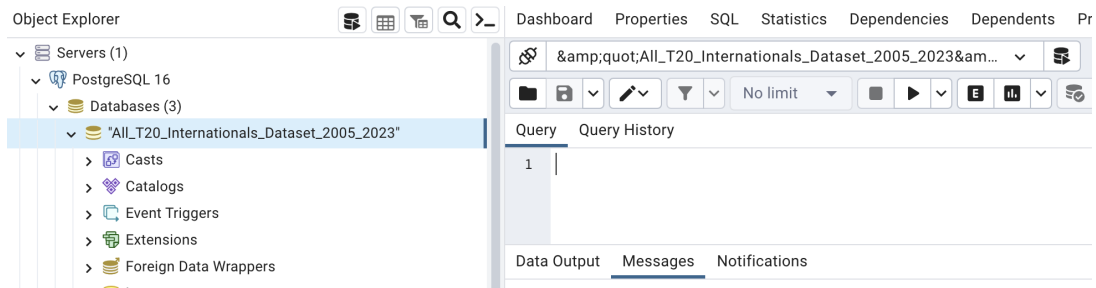


- Name the role "**Data Entry Operators**".



4. Open the Query Tool:

Right-click on the database name and select 'Query Tool' from the context menu to open a new query editor window.



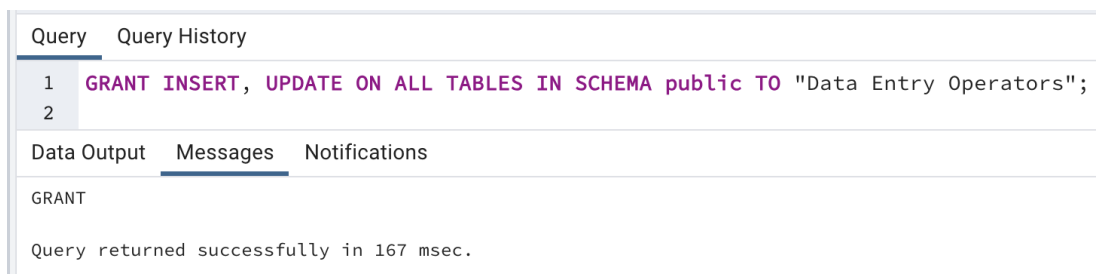
5. Enter the Grant Commands:

In the query editor, enter the following [SQL statements](#):

```
GRANT INSERT, UPDATE ON ALL TABLES IN SCHEMA public TO "Data Entry Operators";
```







6. Execute the Query:

Click the 'Execute/Run' button to run the commands and apply the permissions.



8. Verify the Changes:

You can verify the permissions have been set by looking at the Security tab for the tables within the 'public' schema.

General	Columns	Advanced	Constraints	Parameters	Security	SQL
Privileges						
	Grantee		Privileges		Grantor	
	 Analysts v		r		 postgres	
	 Data Entry Operators v		aw		 postgres	
			<input type="checkbox"/> ALL <input checked="" type="checkbox"/> INSERT <input type="checkbox"/> SELECT <input checked="" type="checkbox"/> UPDATE <input type="checkbox"/> DELETE <input type="checkbox"/> TRUNCATE <input type="checkbox"/> REFERENCES <input type="checkbox"/> TRIGGER	<input type="checkbox"/> WITH GRANT OPTION <input type="checkbox"/> WITH GRANT OPTION <input type="checkbox"/> WITH GRANT OPTION <input type="checkbox"/> WITH GRANT OPTION <input type="checkbox"/> WITH GRANT OPTION <input type="checkbox"/> WITH GRANT OPTION <input type="checkbox"/> WITH GRANT OPTION		

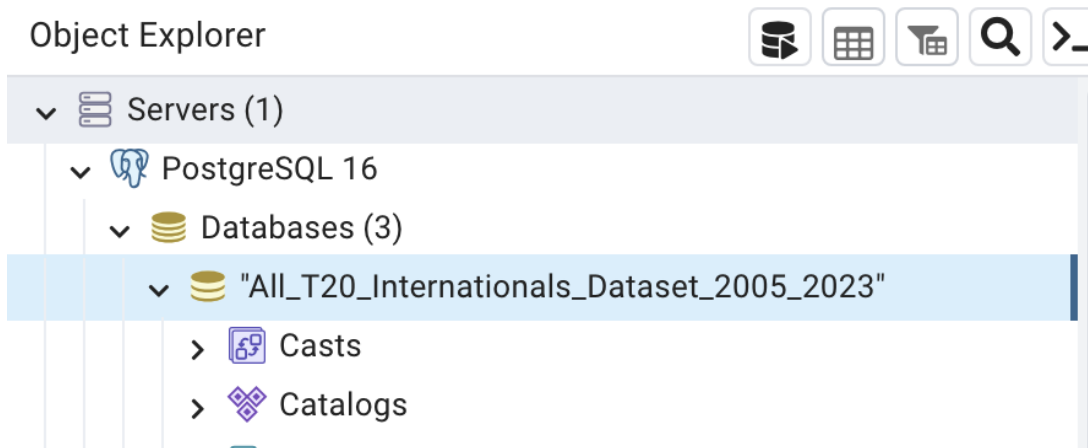
3. Administrators

- Reason for Access:** Administrators possess full privileges across the database to perform tasks that include schema modifications, performance tuning, backup management, and user role assignments.
- Group Members:** This group is composed of database administrators (DBAs) and IT support staff who are tasked with the maintenance and security of the database system.

To set up the "Administrators" group in pgAdmin with full privileges, follow these steps:

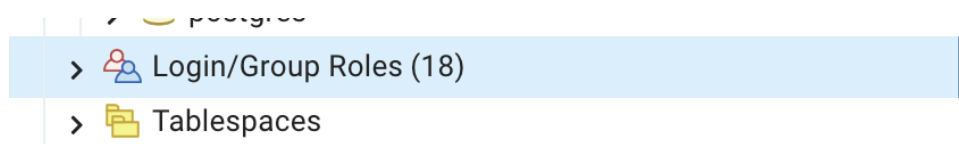
1. Open pgAdmin and Connect to the Database:

Open pgAdmin, then connect to your PostgreSQL server. Navigate to your "All_T20_Internationals_Dataset_2005_2023" database.



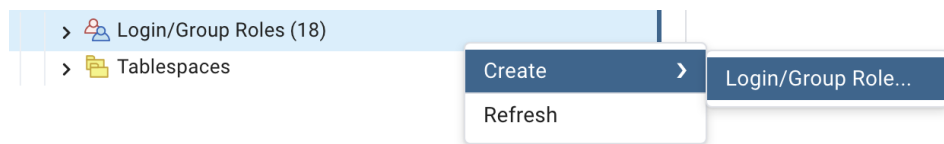
2. Navigate to Login/Group Roles:

- Expand the 'Login/Group Roles' tree under your server.

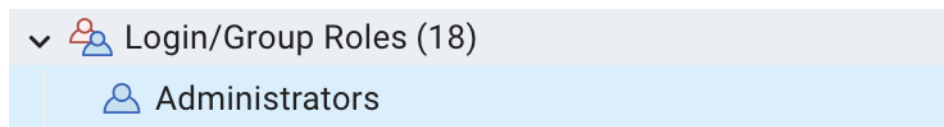


3. Create the Administrators Role:

- Right-click on 'Login/Group Roles' and select 'Create' > 'Login/Group Role'.

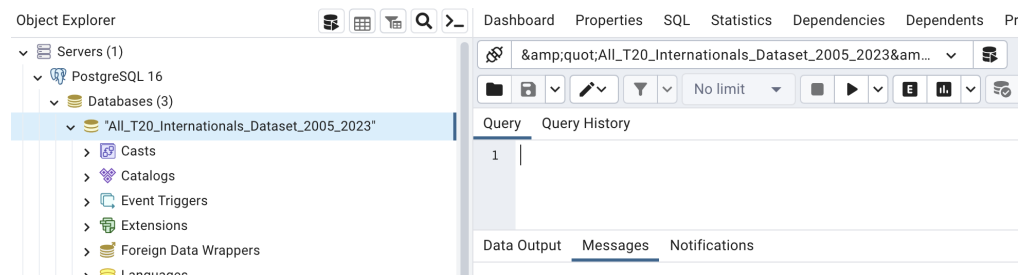


- Name the role "**Administrators**".



4. Open the Query Tool:

Right-click on the database name and select 'Query Tool' from the context menu to open a new query editor window.



5. Grant All Table Privileges:

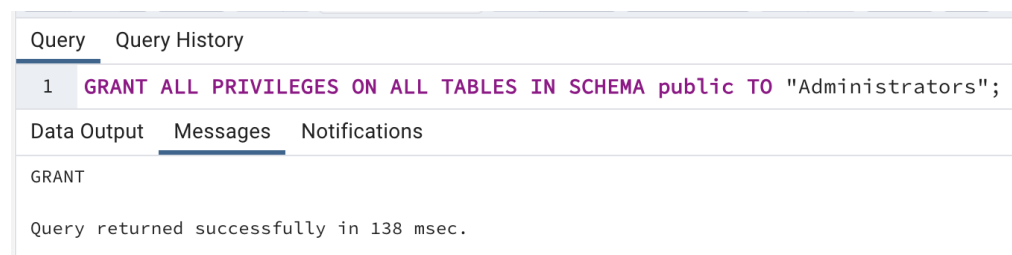
In the query editor, enter the following [SQL statements](#):

```
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO
"Administrators";
```

- This statement assigns the "Administrators" group all available privileges on existing tables within the public schema, such as SELECT, INSERT, UPDATE, and DELETE operations.

Execute the Query:

Click the 'Execute/Run' button to run the commands and apply the permissions.



6. Set Future Default Privileges:

In the query editor, enter the following [SQL statements](#):

```
ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT ALL ON TABLES TO
"Administrators";
```

- This ensures that the "Administrators" group will automatically receive all privileges on new tables created in the future within the public schema.

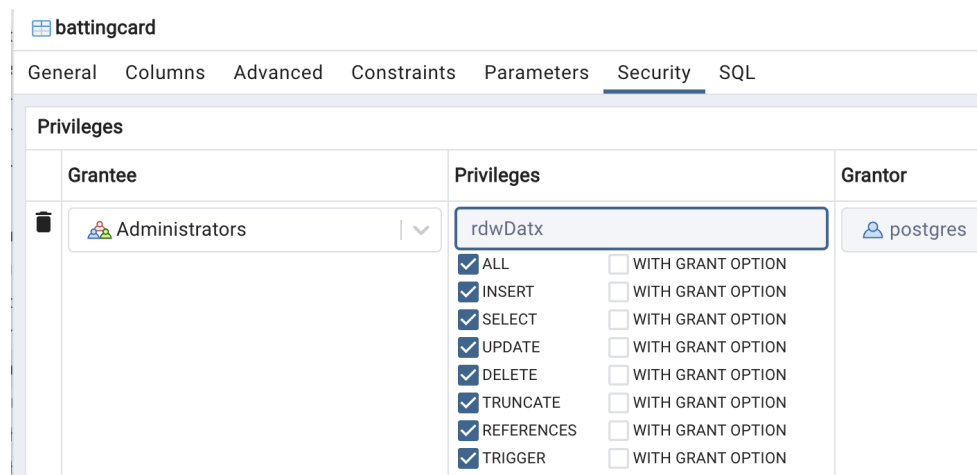
Execute the Query:

Click the 'Execute/Run' button to run the commands and apply the permissions.



7. Verify the Role and Privileges:

You can verify the permissions have been set by looking at the Security tab for the tables within the 'public' schema.



The documentation, along with the corresponding screenshots and SQL text files, demonstrates the careful consideration and execution of role-based access control within the database, ensuring security and appropriate access levels for different user groups.

Conclusion:

Establishing specific user groups within the "All_T20_Internationals_Dataset_2005_2023" database reflects a targeted approach to data governance, balancing the need for data accessibility with the imperative of security. Analysts, Data Entry Operators, and Administrators each have defined roles that align with their operational responsibilities, ensuring they can efficiently fulfill their duties without compromising the database's integrity. This methodical configuration mitigates risks and lays a foundation for a robust data management strategy, ensuring the database remains a reliable resource for informed decision-making and strategic analysis in the dynamic domain of international T20 cricket.

Reference:

(n.d.). *Login/Group Role Dialog*. Pgadmin.

https://www.pgadmin.org/docs/pgadmin4/development/role_dialog.html