Managing Access to Databases and Their Objects

Sure! Let's talk about granting, revoking, and denying access to database objects.

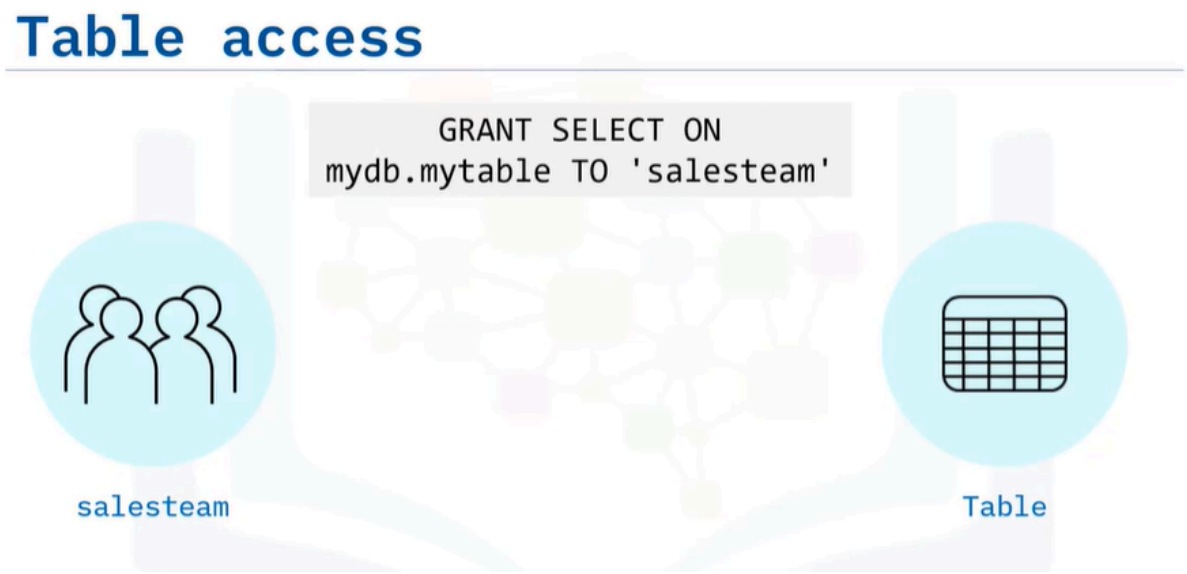
When you work with a database, it's like being in a library filled with books (the data). To read or borrow a book, you need permission. In the database world, this permission is called "privileges." You can grant these privileges to users, groups, or roles, allowing them to access certain tables or perform specific actions like reading, adding, or changing data. For example, if you want a group of salespeople to see customer data, you would grant them "select" privileges on the customer table.

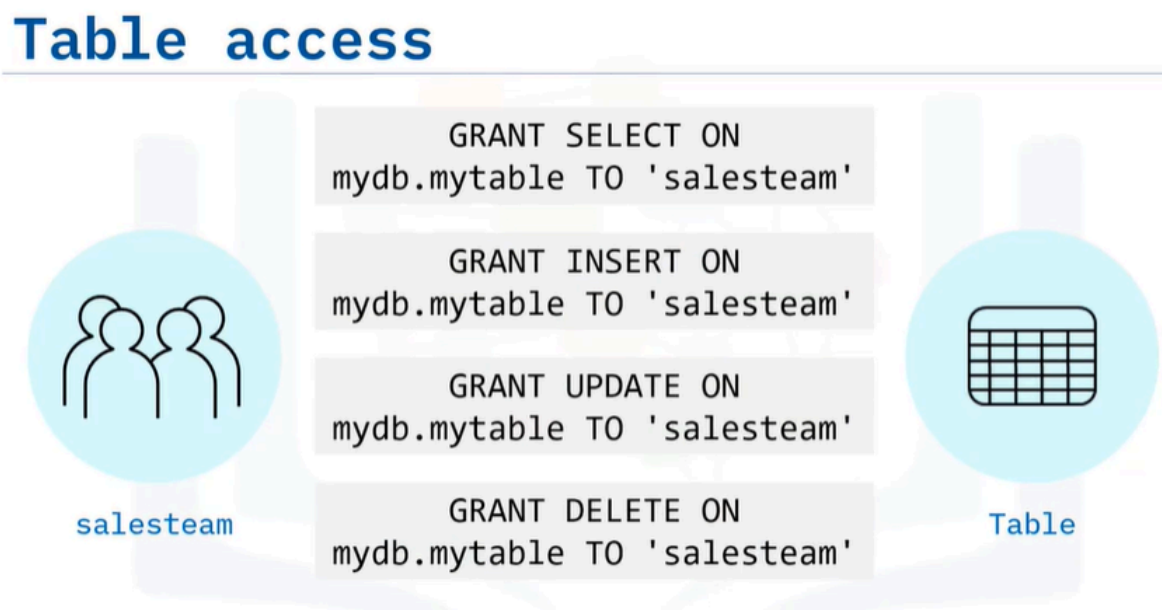
Now, sometimes you might need to take away those permissions. This is where "revoking" comes in. If a user no longer needs access, you can revoke their privileges. However, if a user is part of multiple groups, they might still have access through another group. To completely block access, you can use the "deny" command, which overrides any previous permissions. Think of it like putting a lock on a book that was previously available to everyone.

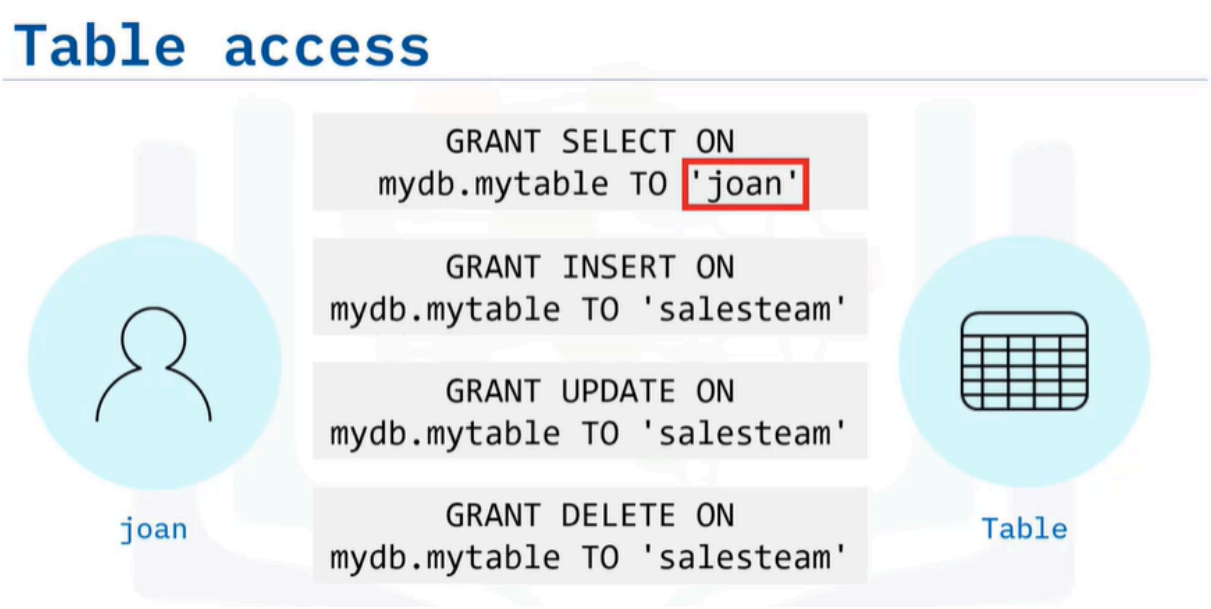
- After a user is authenticated on a database, they need permissions or privileges to access the objects and data in that database. Privileges are granted to users, and to groups or roles. The combination of a user's own personal permissions and those of the groups or roles they belong to defines their overall permissions. In this video, you will see examples of generic SQL statements. The RDBMS that you are using may use slightly different syntax to the examples shown here, and is also likely to provide a graphical interface option to perform these tasks.

- In systems where users are authenticated outside the database (such as on the operating system or through an external plugin), you may also need to grant them access to the databases they will be working with. You can use the SQL GRANT CONNECT command to grant connection access to a particular database to a particular user, group, or role. To grant connection access to an individual user rather than a group or role, you simply specify the user name in place of the group name. If your RDBMS provides a guest or public account, you may find that by default it has connect privileges to all databases.

- In this scenario it is a good practice to revoke that permission so that you ensure only users given explicit permission can access your data.

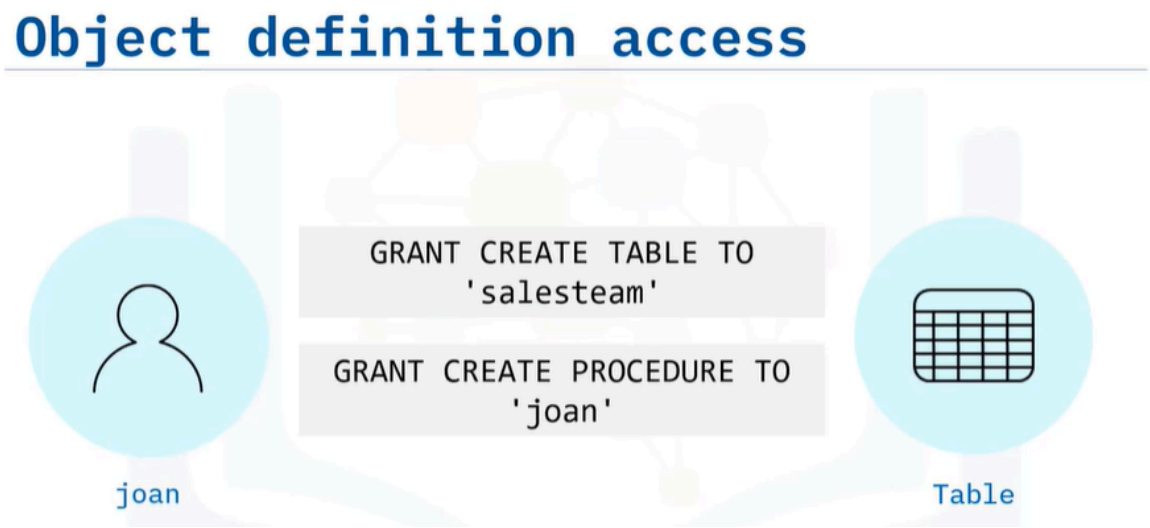




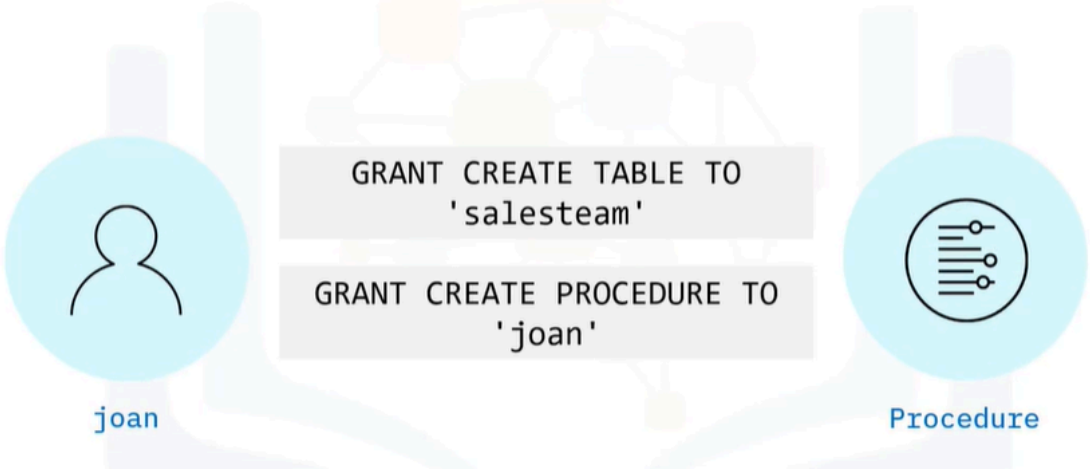


- You use the SQL GRANT command to grant privileges on tables in a database.
- In this example, the salesteam group is granted select privileges on the mytable table in the mydb database. You can use similar statements to enable users to insert, update, and delete data by granting them the relevant privileges on a table or tables.

- To provide these privileges to an individual user, just replace the salesteam group name with the name of that user. And you can also use the GRANT statement to grant privileges to create objects in a database.

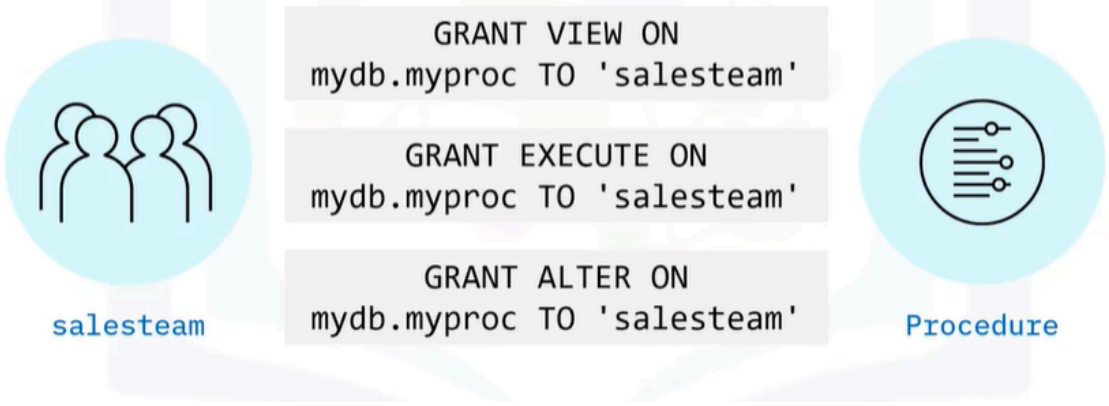


Object definition access



- This example shows how to enable a group or role to create a table. And again, you can use a similar statement to enable a user to create an object such as a table or stored procedure.

Object access



- On a procedure or function, you can permit a user or group to view the code. This means that they can view the definition of that function or procedure, but they will not be able to run it or change it. To run the code, a user needs the

execute permission. And to change the definition of a procedure, a user needs the alter permission.

- As well as granting privileges, at times you may want to remove them. You can use the revoke statement to remove granted privileges, and most RDBMSs will also provide revoke functionality in the user interface. However, because an individual's privileges are a combination of those granted to all the groups or roles they belong to, a member of the sales team role may still be able to access this table through their membership of another role. If you want to ensure that users do not have permission for a certain object or action, you can use the deny statement to override any previous grant of that permission.

Summary

In this video, you learned that:

- You grant permissions to users, groups, or roles
- Permissions control access to databases and the objects in them
- The range of permissions allow for fine tuning of database access
- You can revoke a previously granted permission
- You deny a permission to override an existing granted permission