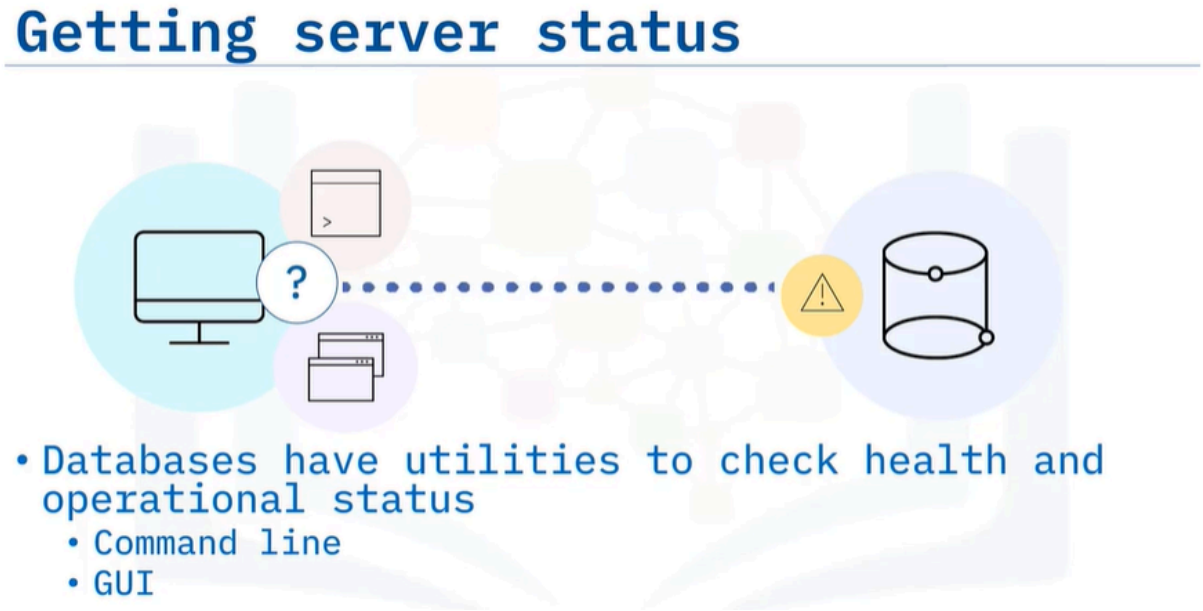
Using Status Variables, Error Codes, and Documentation

Sure! Let's talk about how to check the health and status of a database, which is a crucial part of managing data effectively.

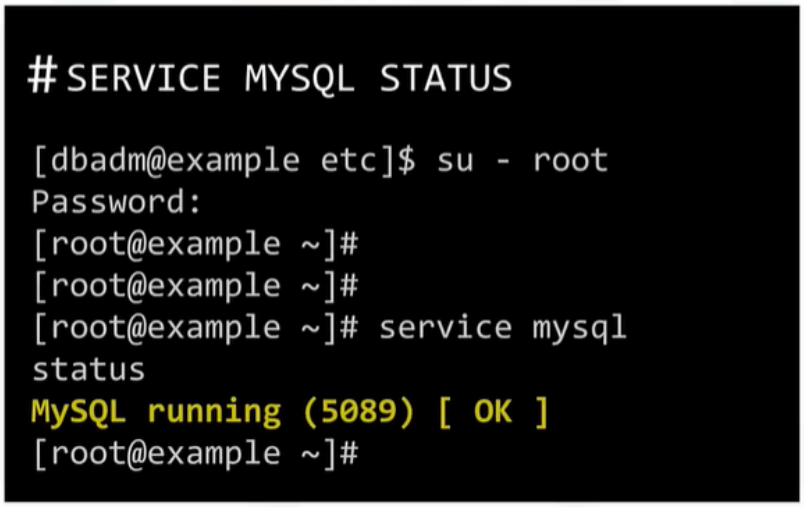
When a database is running, it's like a car on the road. You want to make sure everything is working smoothly. To do this, you can use special commands or tools to get a quick look at how the database is doing. For example, if you're using MySQL, you might type a command like `SERVICE MYSQL STATUS` to see if the database is running well. Just like checking the dashboard of a car for warning lights, these commands help you understand if there are any issues.

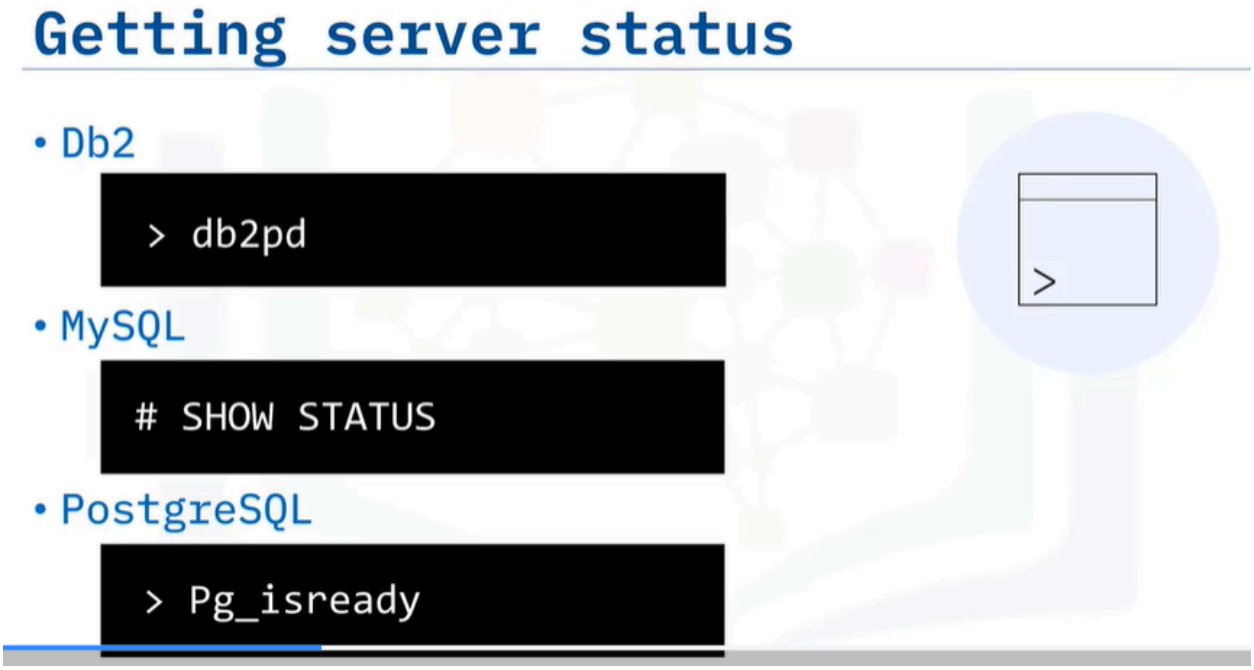
If you find that something isn't right, you can look at logs, which are like a diary for the database. They record what happens, including any errors. By checking these logs, you can find out when a problem occurred and what caused it. Additionally, if you see an error message, it often comes with a code that you can look up in online documentation to understand what went wrong and how to fix it.



- When a database is experiencing a problem, you will first want to check the health and status of the database. All databases have a variety of commands and utilities that allow you to quickly get a snapshot of a database's health and operational status. These utilities are accessed through either typed commands, or a graphical interface.

Getting server status

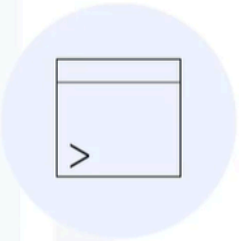




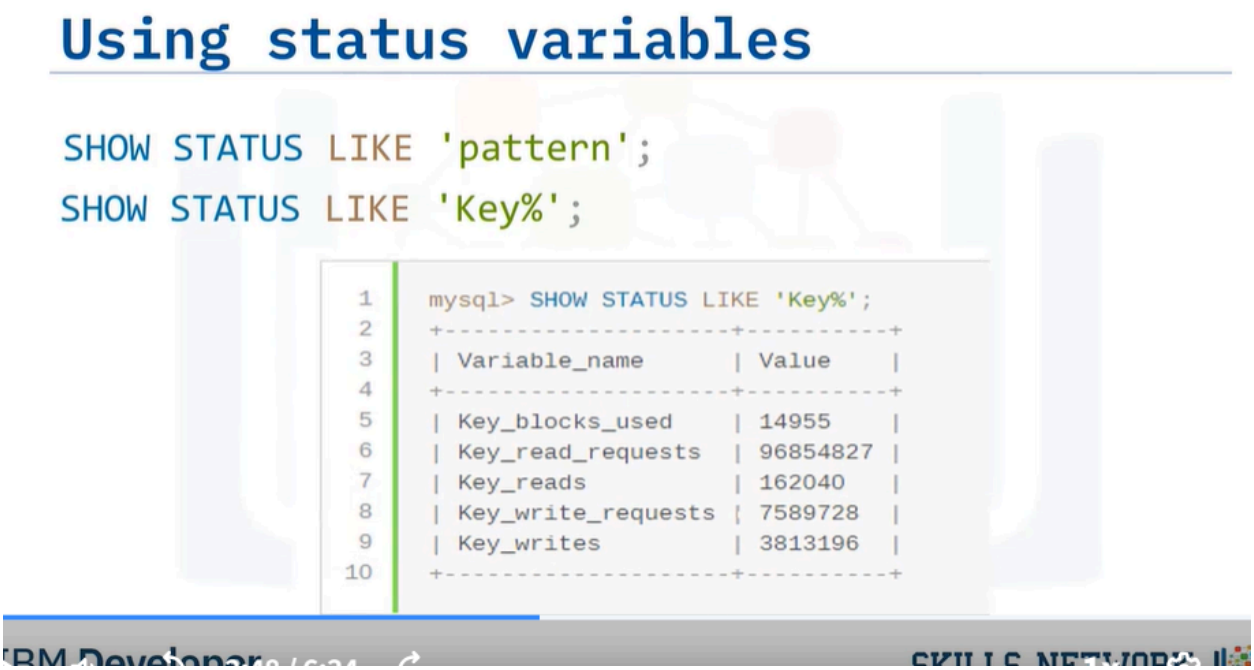
- For example, from a command line in a UNIX environment, you might use the SERVICE MYSQL STATUS command to view MySQL status: And the output might indicate that the database server status is running or is OK, as shown in this screen image. Available commands and syntax vary with the database you are using, such as Db2, MySQL, or PostgreSQL, and the environment in which you are using the database, such as UNIX or Windows. For example, in Db2 you can run the db2pd command to monitor the status of a Db2 instance and run problem diagnostics. For a database using MySQL you might use the SHOW STATUS command to get server status information.
- And on a server running PostgreSQL, you might use the PG_ISREADY command to check the connection status of a PostgreSQL database server.

Using status variables

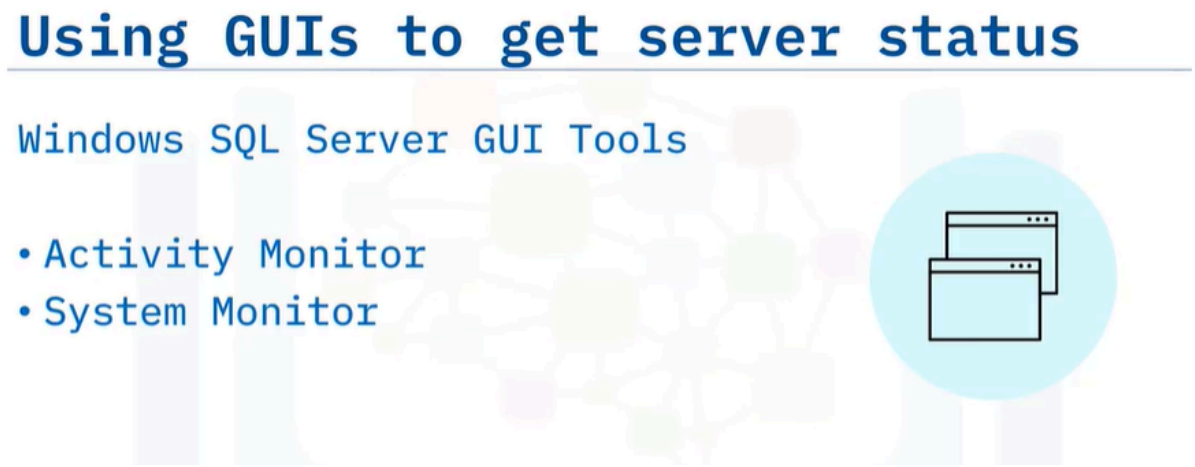
```
# SHOW STATUS

SHOW GLOBAL STATUS

SHOW SESSION STATUS
```



- Databases use many status variables to provide information about their operation, and status variables can be either GLOBAL or SESSION based. A GLOBAL status variable may represent status for some aspect of the server itself (for example, Aborted_connects), or the aggregated status over all connections to MySQL (for example, Bytes_received and Bytes_sent). If a variable has no global value, the session value is displayed. A SESSION status variable, sometimes called a LOCAL status variable, represents values for the current connection.

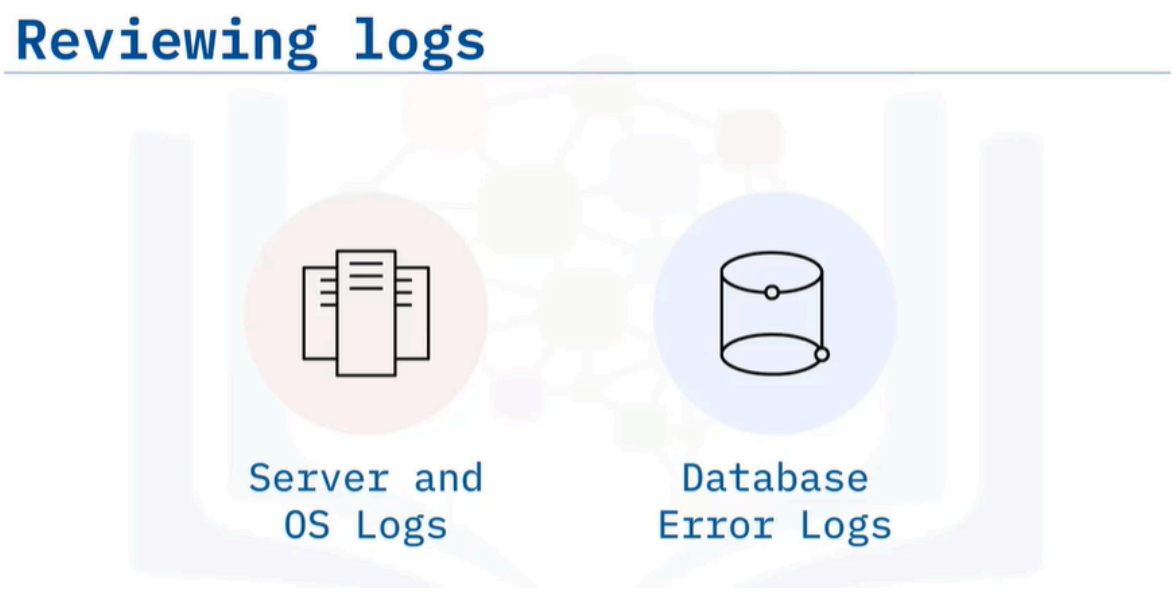


- You can also use a LIKE clause and pattern with a SHOW SESSION statement to show status variable information that matches a specific pattern. For example, if you used the statement SHOW STATUS LIKE ‘KEY%’ you would only see the status for variables with key in the name.



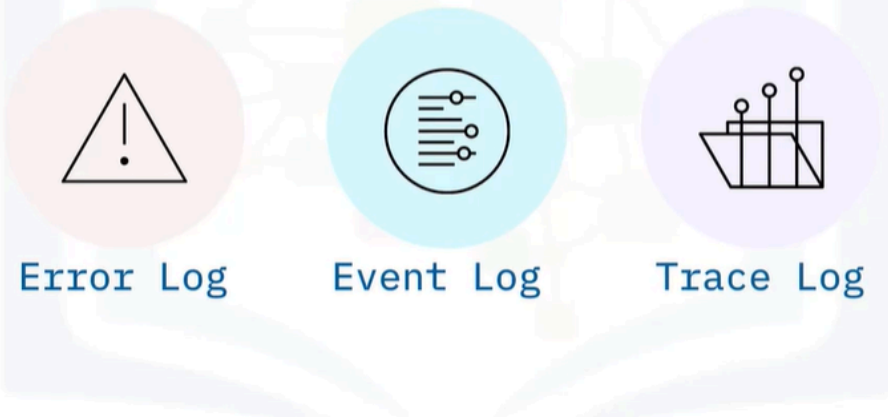
- In addition to commands, you may also have a graphical interface with dashboards and reports for monitoring database status and information in real

time. For example, on a Microsoft SQL Server running on Windows Server, you could use Activity Monitor to get information about SQL Server processes and how the processes affect the current instance of SQL Server, and use System Monitor to verify status and monitor SQL Server performance.



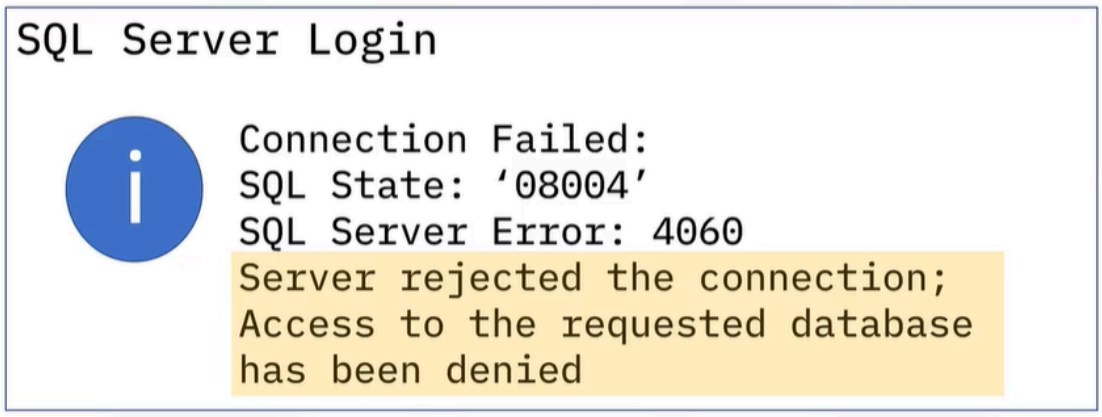
- There are many possible log files you can use to help identify when and where an error occurred. The most used ones are: Server and operating system logs, which log general server activity, connectivity, and other aspects of the server or servers running the database, and database error logs, which log information and errors specific to the database being operated, such as a MYSQL or PostgreSQL system.

Log file examples (SQL)

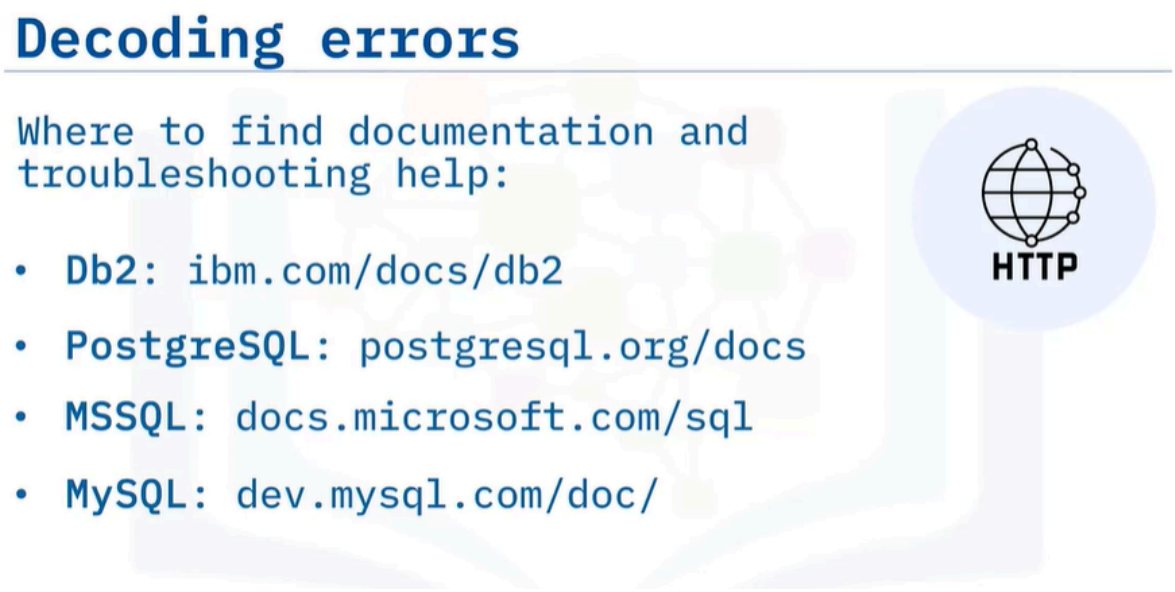


- Log files are important tools for discovering when an error occurred and the description of the error. For example, in a typical SQL server, some of the most accessed logs are: The Error Log file, which is created every time SQL server is started, the Event Log file, which shows informational and error events, and the Default Trace Log, which is an optional log file that tracks all database configuration changes.

Error code example (SQL)



- Whether you use log files or receive an error message, you will most likely need to interpret specific error messages and error codes. For example, error messages or logged errors may contain: A Message and ID number that can be used to troubleshoot the problem. They may also contain other information, such as: The name of the procedure that caused the problem, the state of the database when the error occurred, and additional descriptive information, the severity of the problem, or both. You might also see a line number referenced if the problem occurred in a script or batch file.



- After checking database status and compiling information, your next step is to learn more about the error you are experiencing. There are many documentation and help sites available on the Internet with error code tables that can help you decode and correct errors. Popular resources include: For IBM Db2, information can be found at ibm.com/docs/db2. For PostgreSQL, information can be found at postgresql.org/docs. For Microsoft SQL, information can be found at docs.microsoft.com/sql. For MySQL, information can be found at dev.mysql.com/doc/.

Summary

In this video, you learned that:

- Databases have utilities to check health and operational status
- Available commands and syntax vary with the flavor of database used
- Status variables provide information about database operations: Global or Session
- Many databases also have a GUI for monitoring
- Several log files to help identify when and where errors occur
- Many documentation and help sites to decode errors