

Enable Database Auditing for an Azure SQL Database

Understand the scenario

You are an Azure® administrator. You need to create a new Azure SQL database that uses database auditing. First, you will create an Azure SQL database. Next, you will enable database auditing by using a storage account for the audit log. Finally, you will review the audit log.

Understand your environment

You will be using an Azure resource group named corp-datalod26434568 that contains no resources.

# **Create an Azure SQL database**

* Sign in to the Azure portal

Select the Copy to clipboard icon to copy the text string to the clipboard.

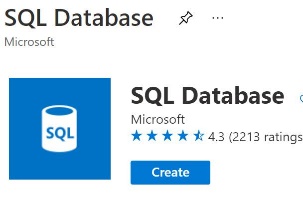
A cloud slice is a subset of an Azure subscription that has been assigned to a user account that was provisioned for you for the duration of this challenge lab. A cloud slice provides temporary access to a subset of resources available in a cloud subscription so that you can learn the concepts in this challenge lab without having to configure your own subscription.

A cloud slice has restrictions on the types of administrative activities that are allowed. Please follow the instructions carefully, especially with regard to names and other configuration details.

* Create an Azure SQL database on a new logical SQL server by using the values in the following table. For any property that is not specified, use the default value.

| **Property** | **Value** |
| --- | --- |
| Resource group | **corp-datalod26434568** |
| Database name | db26434568 |
| Server name | sql26434568 |
| Server admin login | AzureAdmin |
| Password | AzPwd26434568! |
| Compute + storage | **Standard S0** |
| DTUs | **10 (S0)** |
| Data max size | **250 GB** |
| Connectivity method | **Public endpoint** |
| Allow Azure services and resources to access this server | **Yes** |
| Add current client IP address | **Yes** |
| Enable Azure Defender for SQL | **Not now** |
| Use existing data | **Sample** |

* Expand this hint for guidance on creating an Azure SQL database.
  + On the Azure portal home page, select **Create a resource** to open the Azure Marketplace.
  + In the Azure Marketplace, search for and select SQL Database, and then select **Create**.



* + On the Create SQL Database blade, on the Basics page, in Resource group, select **corp-datalod26434568**, and then in Database name, enter db26434568.
  + In Server, select **Create new**.
  + On the New server blade, in Server name, enter sql26434568, in Server admin login, enter AzureAdmin, in Password and Confirm password, enter AzPwd26434568!, and then select **OK**.
  + On the Create SQL Database blade, on the Basics page, in Compute + storage, select **Configure database**.
  + On the Configure blade, select the **Looking for basic, standard, premium?** tile, and then ensure that the **Standard** tile is selected.
  + Ensure that the DTUs slider is set to **10 (S0)**, ensure that the Data max size slider is set to **250 GB**, and then select **Apply**.
  + On the Networking page, in Connectivity method, select **Public endpoint**, in Allow Azure services and resources to access this server, select **Yes**, and then in Add current client IP address, select **Yes**.
  + On the Security page, in Enable Azure Defender for SQL, ensure that **Not now** is selected.
  + On the Additional settings page, in Use existing data, select **Sample**.
  + Select **Review + create**, review the configuration, and then select **Create**.

The sample data will use the AdventureWorksLT sample database.

It will take approximately 3–5 minutes to deploy the Azure SQL database.

[Azure SQL Database](https://docs.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview) is a fully managed database that supports automatic patching, upgrades, and backups.

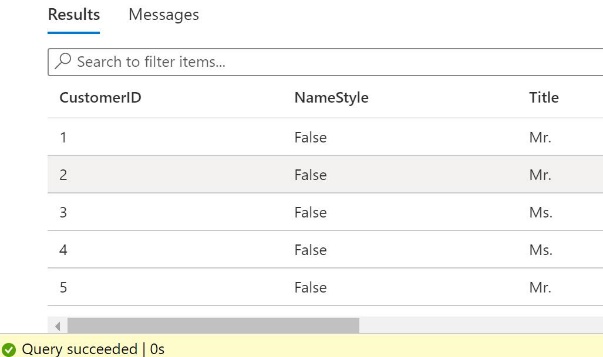
* Log in to the **db26434568** database as AzureAdmin using AzPwd26434568! as the password, and then create a query to retrieve all of the rows in the SalesLT.Customer table.

Expand this hint for guidance on querying the Azure SQL database.

* + On the Azure portal home page, select **All resources**, and then select the **db26434568** SQL database.
  + On the db26434568 resource menu, select **Query editor (preview)**.
  + On the Query editor (preview) page, in Login, ensure that AzureAdmin is entered, in Password, enter AzPwd26434568!, and then select **OK**.
  + In Query1, in line 1, enter the following Transact-SQL (T-SQL) code:

SELECT \* FROM SalesLT.Customer

* + Select **Run** to execute Query1, and then review the results.



If you are prompted for a certificate, select **Cancel**.

* Close the Query editor page, and then if prompted by a **Your unsaved edits will be discarded** message, select **OK**.

## Check your work

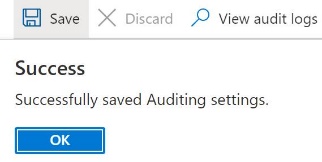
* Confirm that you created an Azure SQL database named db26434568.
* Confirm that you created a new logical SQL server named sql26434568.
* Confirm that you ran a query to retrieve all of the rows in the SalesLT.Customer table.

# **Enable database auditing**

* Enable SQL auditing on the **db26434568** database, and then store the audit log in in a new storage account named sa26434568 in the challenge subscription by using the default settings.

Expand this hint for guidance on enabling auditing at the database level.

* + On the All resources page, select the **db26434568** SQL database.
  + On the db26434568 resource menu, in Security, select **Auditing**.
  + On the Auditing page, turn on the **Enable Azure SQL Auditing** toggle, in Audit log destination, select the **Storage** check box, in Subscription, select the challenge subscription, and then in Storage account, select **Create new**.
  + On the Create storage account blade, in Name, enter sa26434568, review the remaining default values, select **OK**, and then wait for the storage account to be created.
  + On the Auditing page, on the command bar, select **Save**.
  + In the Success message box, select **OK**.



* + Close the **Auditing** page.

If the storage account name is not displayed on the Auditing page immediately after you create the account, save the settings, refresh the page, and then verify that the storage account is displayed.

Azure supports [auditing](https://docs.microsoft.com/en-us/azure/azure-sql/database/auditing-overview" \o "Auditing for Azure SQL Database and Azure Synapse Analytics" \t "_blank) at the server and database level. This allows you to track database events and capture them in an audit log for security, troubleshooting, and compliance reasons. You can store the audit log in an Azure storage account, a Log Analytics workspace, or an event hub.

## Check your work

* Confirm that you enabled database auditing for the db26434568.
* Confirm that you created a storage account named sa26434568 to store the audit log.

# **Test database auditing**

* Log in to the **db26434568** database as AzureAdmin using AzPwd26434568! as the password.

Expand this hint for guidance on logging in to a database.

* + On the All resources page, select the **db26434568** SQL database.
  + On the db26434568 resource menu, select **Query editor (preview)**.
  + On the Query editor page, in SQL server authentication, in Login, ensure that AzureAdmin is entered, in Password, enter AzPwd26434568!, and then select **OK**.
* Create a database user named user1 in the database by using AzPwd26434568! as the password, and then add the new user to the db\_datareader database role.

Expand this hint for guidance on creating a user in a database.

* + In the Query editor page, in Query1, enter the following T-SQL statements:

CREATE USER user1 WITH PASSWORD = 'AzPwd26434568!';

ALTER ROLE db\_datareader ADD MEMBER user1;

* + Select **Run** to execute Query1.
  + On the Messages page, verify that the message **Query succeeded: Affected rows: 0** is returned.
  + Close the Query editor page, and then if prompted by a **Your unsaved edits will be discarded** message, select **OK**.
* Log in to the db26434568 database as user1 using AzPwd26434568! as the password.

Expand this hint for guidance on logging in to a database.

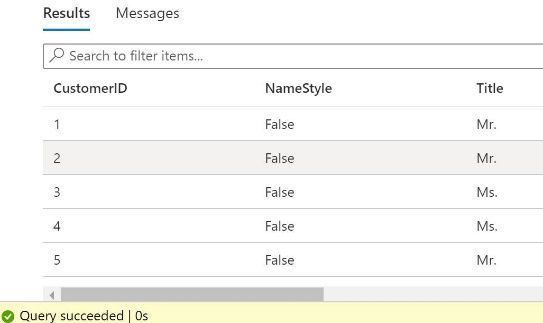
* + On the All resources page, select the **db26434568** SQL database.
  + On the db26434568 resource menu, select **Query editor (preview)**.
  + On the Query editor page, in SQL server authentication, in Login, enter user1, in Password, enter AzPwd26434568!, and then select **OK**.
* Create a query that selects all of the data in the SalesLT.Customer table, and then run the query.

Expand this hint for guidance creating a query.

* + On the Query editor page, in Query1, enter the following SQL code:

SELECT \* FROM SalesLT.Customer;

* + Select **Run** to execute Query1, and then review the results.



* Attempt to delete all of the rows from the table, and then review the error message.

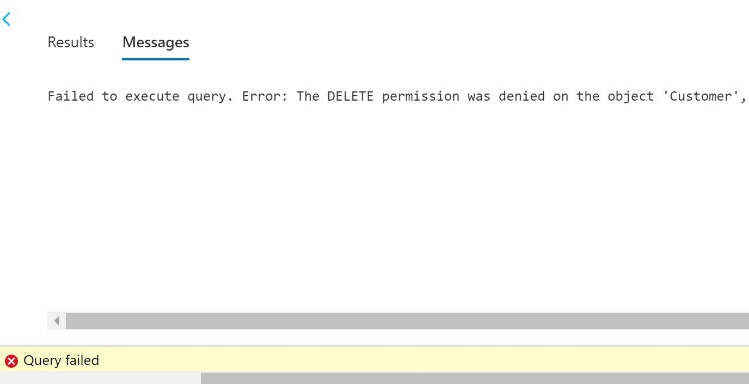
Expand this hint for guidance on deleting rows from a table.

* + In Query1, replace the existing code with the following T-SQL code:

DELETE FROM SalesLT.Customer;

* + Select **Run** to execute Query1, and then review the error message.
  + Close the Query editor page, and then if prompted by a **Your unsaved edits will be discarded** message, select **OK**.

You should see the following error message:

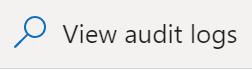


In the db\_datareader role, user1 does not have permission to delete data from a table.

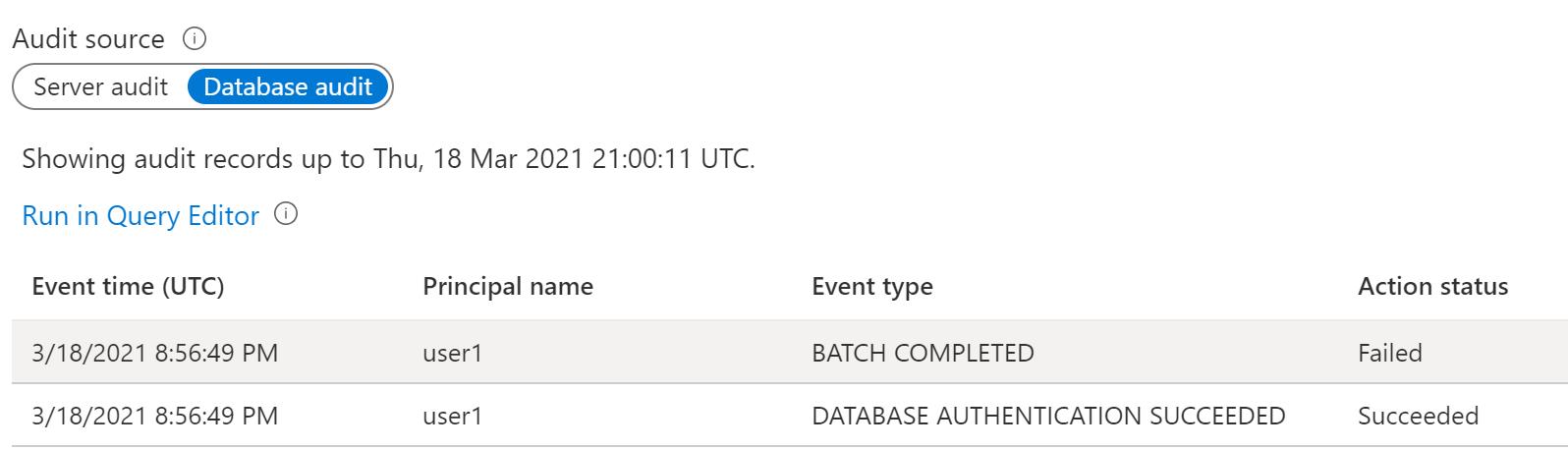
* Review the database-level audit log for the **db26434568** database, and then review the audit of the failed delete statement.

Expand this hint for guidance on viewing the audit log at the database level.

* + On the All resources page, select the **db26434568** SQL database.
  + On the db26434568 resource menu, in Security, select **Auditing**.
  + On the Auditing page, on the command bar, select **View audit logs**.



* + On the Audit records page, in Audit source, ensure that **Database audit** is selected, and then select the most recent audit record that has an Action status of **Failed**.



* + On the Audit record page, ensure that the Principal name shows **user1**, and then ensure that the Status shows **Failed**.
  + On the Audit record page, in STATEMENT, review the failed delete statement.



## Check your work

* Confirm that you created a user named user1 that has read access to the database.
* Confirm that you tested queries as user1.
* Confirm that you viewed the audit log.

# **Summary**

Congratulations, you have completed the **Enable Database Auditing for an Azure SQL Database** challenge.

You have accomplished the following:

* Created an Azure SQL database.
* Enabled database auditing.
* Tested database auditing.