



# **SQL Server Database Auditing**

**INFiNi**  
TECHNO  
SOLUTION

1 / 9
Animated

# SQL SERVER AUDIT

COMPLETE STEP-BY-STEP GUIDE

Track CREATE TABLE, PROCEDURE, DELETE, DROP TABLE at Database Level

Improve Security

Ensure Compliance

Monitor Activities

---

2 / 9
Animated

## 1. Create an Audit Object (Server Level)

This defines where audit logs will be stored.

```

SQL
USE master;
GO
CREATE SERVER AUDIT Audit_SQL_Tracking
TO FILE
(
    FILEPATH = 'C:\SQLAudit\',
    MAXSIZE = 100 MB,
    MAX_ROLLOVER_FILES = 10
)
WITH (ON_FAILURE = CONTINUE);
GO

```

✔

Audit object created.  
Logs will be stored in C:\SQLAudit\

SWIPE

---

3 / 9
Animated

## 2. Enable the Audit (Server Level)

Turn on the audit at the server level.

```

SQL
ALTER SERVER AUDIT Audit_SQL_Tracking
WITH (STATE = ON);
GO

```

✔

Audit is now enabled on the server.

SWIPE

---

4 / 9
Animated

## 3. Create Database Level Audit Specification

Define audit specification in the target database.

```

SQL
USE YourDatabaseName;
GO
CREATE DATABASE AUDIT SPECIFICATION
Audit_DB_Tracking
FOR SERVER AUDIT Audit_SQL_Tracking;
GO

```

✔

Database audit specification created.

SWIPE

---

5 / 9
Animated

## 4. Add Audit Actions (Your Requirements)

Add actions to track CREATE/DROP TABLE, PROCEDURE and DELETE.

```

SQL
ALTER DATABASE AUDIT SPECIFICATION
Audit_DB_Tracking
ADD (SCHEMA_OBJECT_CHANGE_GROUP),
ADD (DELETE ON DATABASE::YourDatabaseName
BY PUBLIC),
ADD (SCHEMA_OBJECT_ACCESS_GROUP);
GO

```

What Each Option Tracks

- SCHEMA\_OBJECT\_CHANGE\_GROUP**  
Tracks CREATE, ALTER, DROP of tables, procedures, views, etc.
- DELETE**  
Tracks DELETE operations in the database.
- SCHEMA\_OBJECT\_ACCESS\_GROUP**  
Tracks access events like SELECT, EXECUTE (optional).

SWIPE

---

6 / 9
Animated

## 5. Enable Database Audit Specification

Turn on the audit specification in the database.

```

SQL
ALTER DATABASE AUDIT SPECIFICATION
Audit_DB_Tracking
WITH (STATE = ON);
GO

```

✔

Database audit is now enabled.

SWIPE

---

7 / 9
Animated

## 6. View Audit Logs

Read audit logs using built-in function.

```

SQL
SELECT *
FROM sys.fn_get_audit_file (
'C:\SQLAudit\*.sqlaudit',
DEFAULT,
DEFAULT
);

```

Sample Output

event_time	action_id	session_id	server_principal_name	statement
2024-05-26 10:15:21.123	SC	58	sa	CREATE TABLE dbo...
2024-05-26 10:16:05.450	DL	58	sa	DELETE FROM dbo...
2024-05-26 10:17:33.789	LO	58	sa	CREATE PROCEDURE...
2024-05-26 10:18:45.012	SC	58	sa	DROP TABLE dbo...

SWIPE

---

8 / 9
Animated

## 7. Filter Specific Events (Example)

Only show DELETE operations.

```

SQL
SELECT event_time, action_id, statement
FROM sys.fn_get_audit_file (
'C:\SQLAudit\*.sqlaudit',
DEFAULT,
DEFAULT
);
WHERE action_id = 'DL'; -- DELETE

```

Action\_ID Reference

Action_ID	Description
SC	Schema Change (Create/Alter/Drop)
DL	Delete
LO	Login
LG	Logout
AL	Audit Login
AE	Audit Logout
EX	Execute (for procedures)

SWIPE

---

9 / 9
Animated

Important Notes

- ✔ Make sure SQL Server service account has Write permission on C:\SQLAudit.
- ✔ Audit is available in SQL Server Standard (basic) and Enterprise (advanced filtering).
- ✔ Avoid enabling too many audit events in production (performance impact).

Bonus: More Granular Control (Optional)

Track DELETE on a specific table only.

```

SQL
ALTER DATABASE AUDIT SPECIFICATION
Audit_DB_Tracking
ADD (DELETE ON OBJECT::dbo.YourTableName
BY PUBLIC);
GO

```

✔

You are now tracking what matters! Stay secure. Stay compliant.

LIKE
|
SAVE
|
SHARE

# ENABLE AUDIT ON SQL SERVER

## Track CREATE TABLE, PROCEDURE, DELETE, DROP TABLE at Database Level

### 1. Create an Audit Object (Server Level)

This defines where audit logs will be stored.

```

USE master;
GO
CREATE SERVER AUDIT Audit_SQL_Tracking
TO FILE
(
FILEPATH = 'C:\SQLAudit\',
MAXSIZE = 100 MB,
MAX_ROW_LOVER_FILES = 10
)
WITH (ON_FAILURE = CONTINUE);
GO
    
```

Audit object created. Logs will be stored in C:\SQLAudit\

### 2. Enable the Audit (Server Level)

Turn on the audit at the server level.

```

ALTER SERVER AUDIT Audit_SQL_Tracking
WITH (STATE = ON);
GO
    
```

100% Command(s) completed successfully.

Audit is now enabled on the server.

### 3. Create Database Level Audit Specification

Define audit specification in the target database.

```

USE YourDatabaseName;
GO
CREATE DATABASE AUDIT SPECIFICATION Audit_DB_Tracking
FOR SERVER AUDIT Audit_SQL_Tracking
GO
    
```

100% Command(s) completed successfully.

Database audit specification created.

### 4. Add Audit Actions (Your Requirements)

Add actions to track CREATE/DROP TABLE, PROCEDURE and DELETE.

```

ALTER DATABASE AUDIT SPECIFICATION Audit_DB_Tracking
ADD (SCHEMA_OBJECT_CHANGE_GROUP),
-- CREATE / ALTER / DROP tables, procedures, views
ADD (DELETE ON DATABASE::YourDatabaseName BY PUBLIC),
-- DELETE operations
ADD (SCHEMA_OBJECT_ACCESS_GROUP);
-- (optional) Access tracking
GO
    
```

Audit actions added.

**What Each Option Tracks**

SCHEMA\_OBJECT\_CHANGE\_GROUP : CREATE, ALTER, DROP (tables, procedures, etc.)  
DELETE : Tracks DELETE statements  
SCHEMA\_OBJECT\_ACCESS\_GROUP : SELECT, EXECUTE (optional)

### 5. Enable Database Audit Specification

Turn on the audit specification in the database.

```

ALTER DATABASE AUDIT SPECIFICATION Audit_DB_Tracking
WITH (STATE = ON);
GO
    
```

100% Command(s) completed successfully.

Database audit is now enabled.

### 6. View Audit Logs

Read audit logs using built-in function.

```

SELECT *
FROM sys.fn_get_audit_file (
'C:\SQLAudit\*.sqlaudit',
DEFAULT,
DEFAULT
);
    
```

event_time	action_id	session_id	server_principal_name	database_name	statement
2024-09-26 10:15:21.122	SC	58	sa	YourDatabaseName	CREATE TABLE dbo.TaskAudit (ID INT)
2024-09-26 10:16:05.046	SC	58	sa	YourDatabaseName	CREATE PROCEDURE dbo.TaskProc AS SELECT 1
2024-09-26 10:17:31.789	DL	58	sa	YourDatabaseName	DELETE FROM dbo.TaskAudit WHERE ID = 1
2024-09-26 10:18:45.012	SC	58	sa	YourDatabaseName	DROP TABLE dbo.TaskAudit

#### Filter Specific Events (Example)

Only show DELETE operations:

```

SELECT event_time, action_id, statement
FROM sys.fn_get_audit_file (
'C:\SQLAudit\*.sqlaudit',
DEFAULT,
DEFAULT,
)
WHERE action_id = 'DL'; -- DELETE
    
```

#### Common Action\_IDs

Action_ID	Description
SC	Schema Change (Create/Alter/Drop)
DL	Delete
LG	Login
LO	Logout
AL	Audit Login
AE	Audit Logout
EX	Execute (for procedures)

#### Important Notes

- Make sure SQL Server service account has Write permission on C:\SQLAudit
- Audit works in SQL Server Standard (basic) and Enterprise (advanced filtering)
- Avoid enabling too many audit events in production (performance impact)

#### Bonus: More Granular Control (Optional)

Track DELETE on a specific table only:

```

ALTER DATABASE AUDIT SPECIFICATION Audit_DB_Tracking
ADD (DELETE ON OBJECT::dbo.YourTableName BY PUBLIC);
GO
    
```



✓ You are now tracking:
📄 Table Create / Drop
📄 Procedure Create / Drop
🗑 Delete Operations
🔍 And more...



# 1. Enable SQL Server Audit (Server Level)

## Step 1: Create an Audit Object

This defines **where logs will be stored** (file, event log, etc.)

```
USE master;
GO

CREATE SERVER AUDIT Audit_SQL_Tracking
TO FILE
(
    FILEPATH = 'C:\SQLAudit\',    -- make sure folder exists
    MAXSIZE = 100 MB,
    MAX_ROLLOVER_FILES = 10
)
WITH (ON_FAILURE = CONTINUE);
GO
```

---

## Step 2: Enable the Audit

```
ALTER SERVER AUDIT Audit_SQL_Tracking
WITH (STATE = ON);
GO
```

---

# ✓ 2. Create Database-Level Audit Specification

Now we define **what actions to track inside a specific database**.

```
USE YourDatabaseName;
GO

CREATE DATABASE AUDIT SPECIFICATION Audit_DB_Tracking
FOR SERVER AUDIT Audit_SQL_Tracking
```

---

# ✓ 3. Add Audit Actions (Your Requirements)

You asked to track:

- CREATE TABLE
- DROP TABLE

- DELETE
- CREATE PROCEDURE
- DROP PROCEDURE

Here's the full script:

```
ALTER DATABASE AUDIT SPECIFICATION Audit_DB_Tracking
ADD (SCHEMA_OBJECT_CHANGE_GROUP), -- CREATE / ALTER / DROP tables,
procedures, views
ADD (DELETE ON DATABASE::YourDatabaseName BY PUBLIC), -- DELETE operations
ADD (SCHEMA_OBJECT_ACCESS_GROUP); -- Access tracking (optional but useful)
GO
```

---

## 🔍 What Each Option Means

Option	Tracks
SCHEMA_OBJECT_CHANGE_GROUP	CREATE, ALTER, DROP (tables, procedures, etc.)
DELETE	Tracks DELETE statements
SCHEMA_OBJECT_ACCESS_GROUP	SELECT, EXECUTE (optional)

---

## ✔ 4. Enable Database Audit

```
ALTER DATABASE AUDIT SPECIFICATION Audit_DB_Tracking
WITH (STATE = ON);
GO
```

---

## ✔ 5. View Audit Logs

You can read audit logs using:

```
SELECT *
FROM sys.fn_get_audit_file (
    'C:\SQLAudit\*.sqlaudit',
    DEFAULT,
    DEFAULT
);
```

---

## ✔ 6. Filter Specific Events (Example)

Only show DELETE operations:

```
SELECT event_time, action_id, statement
FROM sys.fn_get_audit_file (
    'C:\SQLAudit\*.sqlaudit',
    DEFAULT,
    DEFAULT
)
WHERE action_id = 'DL'; -- DELETE
```

---

