

# Laboratorio Numero 07

Escalante Alanoca Jesús Humberto (2015050641)

## I. Introducción

Es el proceso que permite medir, asegurar, demostrar, monitorear y registrar los accesos

a la información almacenada en las bases de datos incluyendo la capacidad de determinar:

- Quién accede a los datos.
- Cuándo se accedió a los datos.
- Desde qué tipo de dispositivo/aplicación.
- Desde que ubicación en la Red.
- Cuál fue la sentencia SQL ejecutada.– Cuál fue el efecto del acceso a la base de datos.

## II. Objetivos

- Mitigar los riesgos asociados con el manejo inadecuado de los datos.
- Apoyar el cumplimiento regulatorio.
- Satisfacer los requerimientos de los auditores.
- Evitar acciones criminales.
- Evitar multas por incumplimiento.

## III. Marco teorico

### i. Azure Data studio

Azure Data Studio es una herramienta de base de datos multiplataforma para profesionales de datos que utilizan la familia de plataformas de datos en la nube y locales de Microsoft en Windows, MacOS y Linux.

Anteriormente publicado bajo el nombre de vista previa de SQL Operations Studio, Azure Data Studio ofrece una experiencia de edición

### iii. Paso 3

Crear una especificacion de auditoria

moderna con IntelliSense, fragmentos de código, integración de control de fuente y un terminal integrado. Está diseñado teniendo en cuenta al usuario de la plataforma de datos,

1

con un registro integrado de conjuntos de resultados de consulta y paneles personalizables.



## IV. Desarrollo

### i. Paso 1

Creando auditoria con las siguientes propiedades

```
1 USE master
2 GO
3 CREATE SERVER AUDIT activity_audit
4 TO FILE (FILEPATH='D:\Auditoria')
5 WITH (QUEUE_DELAY = 1000, ON_FAILURE = CONTINUE);
6 GO
```

Commands completed successfully.  
Commands completed successfully.  
Total execution time: 00:00:00.079

### ii. Paso 2

Activar auditoria

```
[2] 1 ALTER SERVER AUDIT activity_audit WITH (STATE = ON);
2 GO
```

Commands completed successfully.  
Total execution time: 00:00:00.126

```
[3] 1 CREATE SERVER AUDIT SPECIFICATION audit_logins
2 FOR SERVER AUDIT activity_audit ADD (SUCCESSFUL_LOGIN_GROUP)
3 WITH (STATE = ON);
4 GO
```

Commands completed successfully.  
Total execution time: 00:00:00.045

#### iv. Paso 4

Activar la especificacion

```
[4] 1 ALTER SERVER AUDIT SPECIFICATION audit_logins
    2 WITH (STATE = ON);
    3 GO

Commands completed successfully.
Total execution time: 00:00:00.004
```

#### v. Paso 5

Crear yba especificacion de auditoria de base de datos salesapp1

```
[5] 1 USE salesapp1;
    2 GO
    3 CREATE DATABASE AUDIT SPECIFICATION employees_change_audit
    4 FOR SERVER AUDIT activity_audit
    5 ADD (INSERT, UPDATE ON HR.Employees by public)
    6 WITH (STATE = ON);
    7 GO

Commands completed successfully.
Commands completed successfully.
Total execution time: 00:00:00.030
```

#### vi. Paso 6

Activar la especificacion de auditoria de la DB

```
[17] 1 ALTER DATABASE AUDIT SPECIFICATION employees_change_audit WITH (STATE = ON);
     2 GO

Commands completed successfully.
Total execution time: 00:00:00.002
```

#### vii. Paso 7

Ejecutar el codigo, escribir una consulta de sistema sys fn get audit file para devolver todos los datos de auditoria desde los archivos en D:Auditoria, filtrar los datos para que solo la actividad relacionada a la sesion actual sea visualizada.

```
1 SELECT user_defined_event_id, succeeded, user_defined_information
2 FROM sys.fn_get_audit_file ('D:\Auditoria',default,default)

(0 rows affected)

user_defined_event... succeeded user_defined_infor...

Total execution time: 00:00:00.115
```

#### viii. Paso 8

Deshabilitar la auditoria del servidor activity audit

```
[24] 1 USE master;
     2 GO
     3 ALTER SERVER AUDIT activity_audit WITH (STATE = OFF);
     4 GO

Commands completed successfully.
Commands completed successfully.
Total execution time: 00:00:00.065
```