Tecnológico Nacional de México

Instituto Tecnológico de Culiacán

Proyecto auditoría

Unidad 6 Monitoreo y auditoría

Materia:

Administración de Bases de Datos

Docente:

M.C. Daniel Esparza Soto
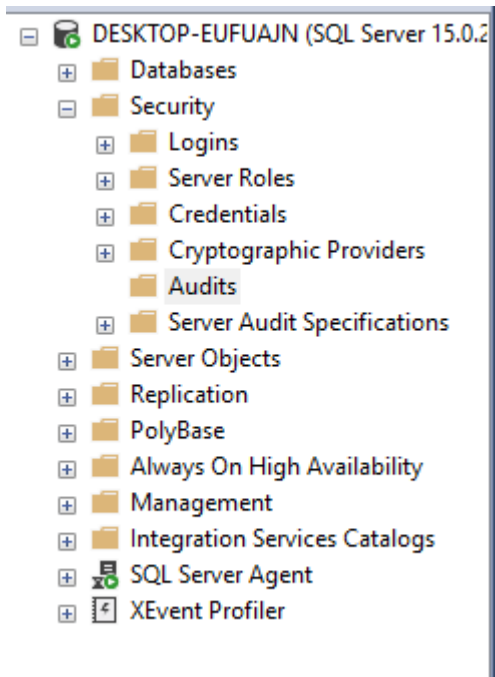
Integrantes:

Astorga Ochoa César Alfredo

Medrano Iturrios Ángel Raúl

Grupo:

6:00 pm - 7:00 pm

Fecha: 16 de mayo de 2022

1.- Proponer un método de Auditar el comando select sobre todas las tablas de la base de datos Norhwind. Realizar 5 consultas y exponga el método de revisión de la auditoria.

Create Audit

— □ ✕

ℹ Ready

Select a page
🔧 General
🔧 Filter

🗂 Script ▾ | ❓ Help

Audit name: | SelectNorthwind

Queue delay (in milliseconds): | 1000

On Audit Log Failure:
◉ Continue
○ Fail operation
○ Shut down server

Audit destination: | File ▾

Path: | C:\ProyectoAud\Select | ...

Audit File Maximum Limit:
◉ Maximum rollover files:
  ☑ Unlimited
○ Maximum files:
  Number of files: | 2147483647

Maximum file size: | 0 ◉ MB ○ GB ○ TB
☑ Unlimited
☐ Reserve disk space

Connection
🔌 DESKTOP-EUFUAJN [sa]

View connection properties

Progress
⟳ Ready

OK | Cancel | Help

Create Server Audit Specification

Ready

Select a page
- General

Script ▾ | **Help**

Name: `ServerAuditSpecification-SelectNorthwind`

Audit: `SelectNorthwind`

Actions:

	Audit Action Type		Object Class		Object Schema	Ob
1	DATABASE_OBJECT_CHANGE_GROUP	▾		▾		
2	SCHEMA_OBJECT_ACCESS_GROUP	▾		▾		
3	DATABASE_OBJECT_PERMISSION_CHANGE_GROUP	▾		▾		
4	SCHEMA_OBJECT_CHANGE_GROUP	▾		▾		
▸ 5	DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP	▾		▾		
＊6		▾		▾		

Connection
DESKTOP-EUFUAJN [sa]

View connection properties

Progress
Ready

OK | Cancel | Help

Create Database Audit Specification

Ready

Select a page
- General

Script ▾ | **Help**

Name: `DatabaseAuditSpecification-SelectNorthwind2`

Audit: `SelectNorthwind`

Actions:

	Audit Action Type		Object Class		Object Schema	Object Name		Principal Name	
01	SELECT	▾	OBJECT	▾	dbo	Categories	...	dbo	...
02	SELECT	▾	OBJECT	▾	dbo	Customers	...	dbo	...
03	SELECT	▾	OBJECT	▾	dbo	Employees	...	dbo	...
04	SELECT	▾	OBJECT	▾	dbo	EmployeeTerritories	...	dbo	...
05	SELECT	▾	OBJECT	▾	dbo	Order Details	...	dbo	...
06	SELECT	▾	OBJECT	▾	dbo	Orders	...	dbo	...
07	SELECT	▾	OBJECT	▾	dbo	Products	...	dbo	...
08	SELECT	▾	OBJECT	▾	dbo	Region	...	dbo	...
09	SELECT	▾	OBJECT	▾	dbo	Shippers	...	dbo	...
10	SELECT	▾	OBJECT	▾	dbo	Suppliers	...	dbo	...
11	SELECT	▾	OBJECT	▾	dbo	Territories	...	dbo	...

Connection
DESKTOP-EUFUAJN [sa]

View connection properties

Progress
Ready

OK | Cancel | Help

```sql
select*from Categories
select*from Customers
select*from Employees
select*from EmployeeTerritories
select*from [Order Details]
select*from Orders
select*from Products
select*from Region
select*from Shippers
select*from Suppliers
select*from Territories

select object_name , statement, server_principal_name,*
from fn_get_audit_file('C:\ProyectoAud\Select\SelectNorthwind_4DA475CE-5DFC-4E51-
9752-D8113075605A_0_132971326244290000.sqlaudit',
default, default)
where object_name is not null and statement like '%select%'
```

	object_name	statement	server_principal_name	event_time	sequence_number	action_id	succeeded	permission_bitmask	
1...	Categories	select*from Categories	sa	2022-05-16 00:18:40.2226459	1	SL	1	0x000000000000000000000000000	
1...	Customers	select*from Customers	sa	2022-05-16 00:18:40.2407131	1	SL	1	0x000000000000000000000000000	
1...	Employees	select*from Employees	sa	2022-05-16 00:18:40.3363673	1	SL	1	0x000000000000000000000000000	
1...	EmployeeTerritories	select*from EmployeeTerritories	sa	2022-05-16 00:18:40.3601902	1	SL	1	0x000000000000000000000000000	
1...	Order Details	select*from [Order Details]	sa	2022-05-16 00:18:40.3601902	1	SL	1	0x000000000000000000000000000	
1...	Orders	select*from Orders	sa	2022-05-16 00:18:40.4614663	1	SL	1	0x000000000000000000000000000	
1...	Products	select*from Products	sa	2022-05-16 00:18:40.4929346	1	SL	1	0x000000000000000000000000000	
1...	Region	select*from Region	sa	2022-05-16 00:18:40.5037862	1	SL	1	0x000000000000000000000000000	
1...	Shippers	select*from Shippers	sa	2022-05-16 00:18:40.5037862	1	SL	1	0x000000000000000000000000000	
1...	Suppliers	select*from Suppliers	sa	2022-05-16 00:18:40.5037862	1	SL	1	0x000000000000000000000000000	
1...	Territories	select*from Territories	sa	2022-05-16 00:18:40.5037862	1	SL	1	0x000000000000000000000000000	

2.- Proponer un método para auditar la execución de todos los procedimientos almacenados de la bd Northwind. Crear dos procedimientos almacenado y exponga el método de revisión de la auditoría.

Server Audit Specification Properties

Ready

Select a page
🔧 General

Script ▾ | ❓ Help

Name: `ServerAuditSpecification-ExecNorthwind1`

Audit: `ExecNorthwind` ▾

Actions:

	Audit Action Type		Object Class
▶ 1	DATABASE_OBJECT_ACCESS_GROUP	⌄	
2	SCHEMA_OBJECT_ACCESS_GROUP	⌄	
3	DATABASE_OBJECT_PERMISSION_CHANGE_GROUP	⌄	
4	DATABASE_OBJECT_CHANGE_GROUP	⌄	
5	DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP	⌄	
＊6		⌄	

Connection
🔌 DESKTOP-EUFUAJN [sa]
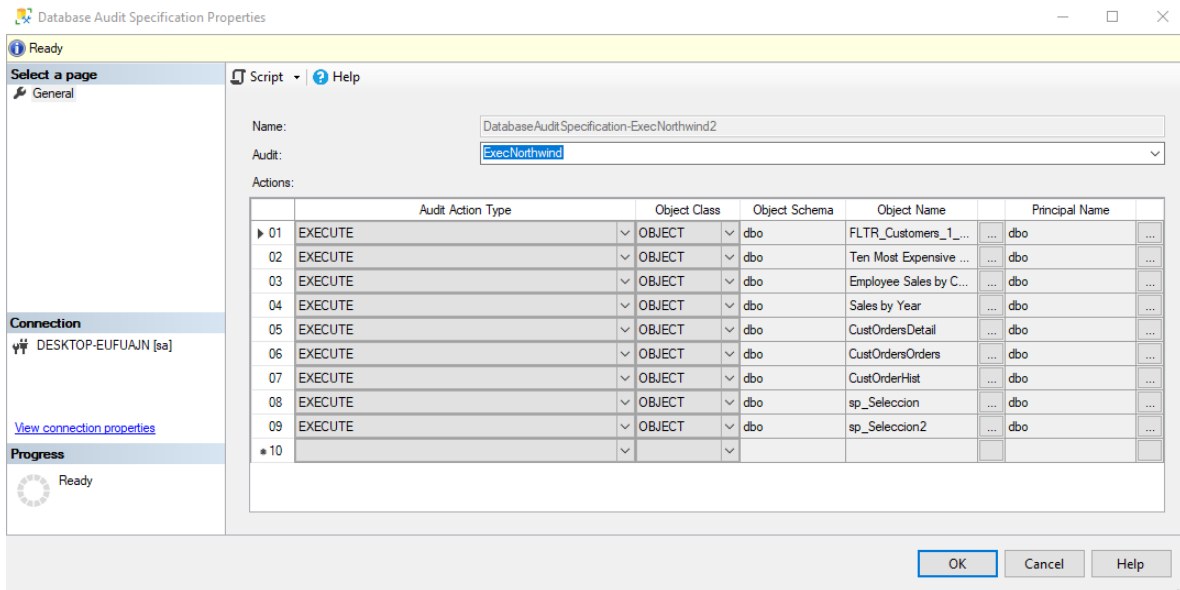
View connection properties

Progress
⟳ Ready

OK | Cancel | Help

```sql
create proc sp_Seleccion as
select*from products where Categoryid = 1

create proc sp_Seleccion2 as
select*from orders

exec sp_Seleccion
exec sp_Seleccion2

select object_name , statement, server_principal_name,*
from fn_get_audit_file('C:\ProyectoAud\Exec\ExecNorthwind_525F7D04-B524-4B90-86ED-
33C77ED78298_0_132971369253130000.sqlaudit',
default, default)
where object_name is not null and statement like 'exec%'
```

object_name	statement	server_principal_name	event_time	sequence_number	action_id	si
4... sp_replmonitorrefreshagentdata	exec @retcode = sys.sp_replmonitorrefreshagentdata	NT SERVICE\SQLSERVERAGENT	2022-05-16 01:45:07.0180030	1	EX	1
4... sp_replmonitorrefreshwarningdata	exec @retcode = sys.sp_replmonitorrefreshwarningd...	NT SERVICE\SQLSERVERAGENT	2022-05-16 01:45:07.2766712	1	EX	1
4... sp_replmonitorsynchronize	exec @retcode = sys.sp_replmonitorsynchronize @m...	NT SERVICE\SQLSERVERAGENT	2022-05-16 01:45:07.4776443	1	EX	1
4... sp_getapplock	exec @retcode = sp_getapplock @Resource = 'rep...	sa	2022-05-16 01:45:07.4776443	1	EX	1
4... xp_userlock	exec @result = sys.xp_userlock 0, @dbid, @DbPrinc...	sa	2022-05-16 01:45:07.4776443	1	EX	1
4... sp_replmonitorsynchronize	exec @retcode = sys.sp_replmonitorsynchronize @m...	NT SERVICE\SQLSERVERAGENT	2022-05-16 01:45:07.4856247	1	EX	1
4... sp_releaseapplock	exec @retcode = sp_releaseapplock @Resource = ...	sa	2022-05-16 01:45:07.4856247	1	EX	1
4... xp_userlock	exec @result = sys.xp_userlock 1, @dbid, @DbPrinc...	sa	2022-05-16 01:45:07.4856247	1	EX	1
4... sp_replmonitorsynchronize	exec @retcode = sys.sp_replmonitorsynchronize @m...	NT SERVICE\SQLSERVERAGENT	2022-05-16 01:45:07.4856247	1	EX	1
4... sp_releaseapplock	exec @retcode = sp_releaseapplock @Resource = ...	sa	2022-05-16 01:45:07.4856247	1	EX	1
4... xp_userlock	exec @result = sys.xp_userlock 1, @dbid, @DbPrinc...	sa	2022-05-16 01:45:07.4856247	1	EX	1
4... sp_Seleccion	exec sp_Seleccion	sa	2022-05-16 01:45:08.3985669	1	EX	1
4... sp_Seleccion2	exec sp_Seleccion2	sa	2022-05-16 01:45:08.4047411	1	EX	1

3.- Utilizando trigger DDL a nivel servidor auditar la creación, eliminación o modificación de base de datos y de inicios de sesión. Grabar en una tabla:

- ID
- fecha del evento,
- inicio de sesión,
- acción,
- sentencia y
- detalle del evento XML.

Realizar 5 acciones sobre base de datos y 5 acciones sobre inicios de sesión.

```sql
--tabla que se llenara dentro del trigger DLL sobre base de datos
create table CreEliMod(
ID int identity primary key,
Fecha datetime not null constraint DF_LogCambiosDDL_FechaCmabio default(getdate()),
InicioSesion varchar(500) not null,
Accion varchar(500) not null,
Sentencia varchar(max) not null,
DetalleEvento xml not null);
go

--Trigger DLL sobre servidor: auditoria sobre creacion, eliminacion o modificacion
de base de datos e inicios de sesion
create trigger tr_AuditaCambios
on all server
for
CREATE_DATABASE, DROP_DATABASE, ALTER_DATABASE,
CREATE_LOGIN, DROP_LOGIN, ALTER_LOGIN
--DLL_TABLE_VIEW_EVENTS
as begin
    set nocount on;
```

```sql
        --Declaracion de variables de proceso
        declare @InfoEvento xml, @Accion varchar(500), @Sentencia varchar(max);

        --1.-Captura de evento en variable principal
        set @InfoEvento = EVENTDATA();

        --2.- Consumir informacion en variables secundarias usando XQuery
        ------Captura la accion DDL (si es CREATE, ALTER o DROP)
        set @Accion = @InfoEvento.value('(/EVENT_INSTANCE/EventType)[1]',
'varchar(500)');

        ------Captura la secuencia ejecutada
        set @Sentencia = @InfoEvento.value('(/EVENT_INSTANCE/TSQLCommand)[1]',
'nvarchar(max)')

        --3.- Almacenar detalles de la accion en la tabla log
    insert dbo.CreEliMod(Fecha,InicioSesion,Accion,Sentencia,DetalleEvento)
        values(getdate(), SUSER_NAME(), @Accion, @Sentencia, @InfoEvento);
end;
go

create database pruebass
drop database pruebass
create database primera
create database segunda
drop database primera

create login UsuarioUno with password = 'hola'
create login UsuarioDos with password = 'hola'
create login UsuarioTres with password = 'hola'

drop login UsuarioUno
drop login UsuarioDos
drop login UsuarioTres

select*from dbo.CreEliMod
```

	ID	Fecha	InicioSesion	Accion	Sentencia	DetalleEvento
1	1	2022-05-15 22:12:41.397	sa	CREATE_DATABASE	create database pruebass	\<EVENT_INSTANCE>\<EventType>CREATE_DATABASE\</Even...
2	2	2022-05-15 22:13:28.160	sa	DROP_DATABASE	drop database pruebass	\<EVENT_INSTANCE>\<EventType>DROP_DATABASE\</EventT...
3	3	2022-05-15 22:16:42.703	sa	CREATE_DATABASE	create database primera	\<EVENT_INSTANCE>\<EventType>CREATE_DATABASE\</Even...
4	4	2022-05-15 22:16:44.747	sa	CREATE_DATABASE	create database segunda	\<EVENT_INSTANCE>\<EventType>CREATE_DATABASE\</Even...
5	5	2022-05-15 22:16:46.710	sa	DROP_DATABASE	drop database primera	\<EVENT_INSTANCE>\<EventType>DROP_DATABASE\</EventT...
6	6	2022-05-15 22:16:54.807	sa	CREATE_LOGIN	create login UsuarioUno with password = '*******'	\<EVENT_INSTANCE>\<EventType>CREATE_LOGIN\</EventTyp...
7	7	2022-05-15 22:16:56.553	sa	CREATE_LOGIN	create login UsuarioDos with password = '*******'	\<EVENT_INSTANCE>\<EventType>CREATE_LOGIN\</EventTyp...
8	8	2022-05-15 22:16:58.080	sa	CREATE_LOGIN	create login UsuarioTres with password = '*******'	\<EVENT_INSTANCE>\<EventType>CREATE_LOGIN\</EventTyp...
9	9	2022-05-15 22:17:00.503	sa	DROP_LOGIN	drop login UsuarioUno	\<EVENT_INSTANCE>\<EventType>DROP_LOGIN\</EventType>...
10	10	2022-05-15 22:17:02.347	sa	DROP_LOGIN	drop login UsuarioDos	\<EVENT_INSTANCE>\<EventType>DROP_LOGIN\</EventType>...
11	11	2022-05-15 22:17:03.987	sa	DROP_LOGIN	drop login UsuarioTres	\<EVENT_INSTANCE>\<EventType>DROP_LOGIN\</EventType>...

4.- Utilizando trigger DDL a nivel de base de datos auditar la creación, eliminación o modificación de tablas vistas, procedimientos almacenados, funciones y trigger. Grabar en una tabla:

- ID,
- Fecha del evento,
- inicio de sesión,
- usuario de base de datos,
- acción,
- objeto,
- sentencia y
- detalle del evento XML.

Realizar un comando para vistas, procedimientos almacenados, funciones y trigger.

```sql
--tabla que se llenara dentro del trigger DLL sobre base de datos
create table CambiosDDL(
ID int identity primary key,
Fecha datetime not null constraint DF_CambiosDDL_FechaCambio default(getdate()),
InicioSesion varchar(500) not null,
Usuario varchar(500) not null,
Accion varchar(500) not null,
Objeto varchar(500),
Sentencia varchar(max) not null,
DetalleEvento xml not null);
go

--Trigger DLL sobre base de datos: auditoria sobre creacion, modificacion y
eliminacion de tablas, vistas, procedimientos almacenados
-- funciones Y triggers
create trigger trDB_AuditoriaCambiosA
on database
for
CREATE_TABLE, ALTER_TABLE, DROP_TABLE,
CREATE_VIEW, ALTER_VIEW, DROP_VIEW,
CREATE_PROCEDURE, ALTER_PROCEDURE, DROP_PROCEDURE,
CREATE_FUNCTION, ALTER_FUNCTION, DROP_FUNCTION,
CREATE_TRIGGER, ALTER_TRIGGER, DROP_TRIGGER
as begin
    set nocount on;
        --Declaracion de variables de proceso
        declare @InfoEvento xml, @Accion varchar(500), @Objeto varchar(500),
@Sentencia varchar(max);

        --1.-Captura de evento en variable principal
        set @InfoEvento = EVENTDATA();

        --2.- Consumir informacion en variables secundarias usando XQuery
        ------Captura la accion DDL (si es CREATE, ALTER o DROP)
        set @Accion = @InfoEvento.value('(/EVENT_INSTANCE/EventType)[1]',
'varchar(500)');

        ------Captura el nombre objeto afectado, anteponiendo el nombre del esquema
        set @Objeto = @InfoEvento.value('(/EVENT_INSTANCE/SchemaName)[1]',
'varchar(250)')
```

```sql
                        +'.'+@InfoEvento.value('(/EVENT_INSTANCE/ObjectName)[1]',
'varchar(250)');

        ------Captura la secuencia ejecutada
        set @Sentencia = @InfoEvento.value('(/EVENT_INSTANCE/TSQLCommand)[1]',
'nvarchar(max)')

        --3.- Almacenar detalles de la accion en la tabla log
    insert
dbo.CambiosDDL(Fecha,InicioSesion,Usuario,Accion,Objeto,Sentencia,DetalleEvento)
        values(getdate(), SUSER_NAME(), CURRENT_USER, @Accion, @Objeto, @Sentencia,
@InfoEvento);
end;
go

create view vw_AuditoriaUno as
select orderid, orderdate, employeeid
from orders

create proc sp_AuditoriaUno as
select*from region

create function fn_AuditoriaUno()
returns table as
return
(select companyname from suppliers)

alter function fn_AuditoriaUno()
returns table as
return
(select ContactTitle from suppliers)

create trigger tr_AuditoriaUno on suppliers
for insert
as
begin
print 'Prueba'
end

alter trigger tr_AuditoriaUno on suppliers
for insert
as
begin
print 'Auditoria'
end

drop view vw_AuditoriaUno

drop proc sp_AuditoriaUno

drop function fn_AuditoriaUno()

drop trigger tr_AuditoriaUno

select*from dbo.CambiosDDL
```

	ID	Fecha	InicioSesion	Usuario	Accion	Objeto	Sentencia	DetalleEvento
1	1	2022-05-15 22:45:02.783	sa	dbo	CREATE_VIEW	dbo.vw_AuditoriaUno	create view vw_AuditoriaUno as select orderid, ...	\<EVENT_INSTANCE\>\<EventType\>CREATE_VIEW\<
2	2	2022-05-15 22:46:47.223	sa	dbo	CREATE_PROCEDURE	dbo.sp_AuditoriaUno	create proc sp_AuditoriaUno as select *from region	\<EVENT_INSTANCE\>\<EventType\>CREATE_PROCE
3	3	2022-05-15 23:00:53.457	sa	dbo	CREATE_FUNCTION	dbo.fn_AuditoriaUno	create function fn_AuditoriaUno() returns table ...	\<EVENT_INSTANCE\>\<EventType\>CREATE_FUNCT
4	4	2022-05-15 23:03:10.923	sa	dbo	CREATE_TRIGGER	dbo.tr_AuditoriaUno	create trigger tr_AuditoriaUno on suppliers for ins...	\<EVENT_INSTANCE\>\<EventType\>CREATE_TRIGG
5	5	2022-05-15 23:05:59.203	sa	dbo	ALTER_FUNCTION	dbo.fn_AuditoriaUno	alter function fn_AuditoriaUno() returns table as ...	\<EVENT_INSTANCE\>\<EventType\>ALTER_FUNCTIO
6	6	2022-05-15 23:06:03.377	sa	dbo	ALTER_TRIGGER	dbo.tr_AuditoriaUno	alter trigger tr_AuditoriaUno on suppliers for inser...	\<EVENT_INSTANCE\>\<EventType\>ALTER_TRIGGEI
7	7	2022-05-15 23:06:06.270	sa	dbo	DROP_VIEW	dbo.vw_AuditoriaUno	drop view vw_AuditoriaUno	\<EVENT_INSTANCE\>\<EventType\>DROP_VIEW\</E
8	8	2022-05-15 23:06:08.493	sa	dbo	DROP_PROCEDURE	dbo.sp_AuditoriaUno	drop proc sp_AuditoriaUno	\<EVENT_INSTANCE\>\<EventType\>DROP_PROCEDI
9	9	2022-05-15 23:06:12.620	sa	dbo	DROP_TRIGGER	dbo.tr_AuditoriaUno	drop trigger tr_AuditoriaUno	\<EVENT_INSTANCE\>\<EventType\>DROP_TRIGGEF

5.- Auditar los cambios de las tablas de la base de datos Northwind, grabar en una tabla los campos:

- TipoTransaccion: insert, update, delete
- Nombre de la Tabla
- Campo de la llave primaria
- Campo Modificado
- Valor Original
- Valor Nuevo
- Fecha de la transacción
- Nombre de Usuario

Tabla	Acción	Campos
Products	Update	unitprice
customers	Update	address, phone
Region	Insert/delete	

```sql
create table TipoCambios (
TipoTran varchar(6),
Tabla varchar(128),
PK varchar(1000),
CampoModificado varchar(128),
ValorOrginal Varchar(1000),
ValorNuevo Varchar(1000),
FechaTran DateTime Not null constraint DF_TipoCambios_FechaTran DEFAULT (gETdATE()),
Usuario varchar(128)
)

--Products, Accion = Update, Campos = Unitprice
create trigger tr_ProductsCambios on products for update
as
begin
declare @TipoTran varchar(6), @Tabla varchar(128), @PK varchar(1000),
@CampoModificado varchar(128), @ValorOriginal varchar(1000),
@ValorNuevo varchar(1000), @FechaTran datetime, @Usuario varchar(128)

if exists(select*from inserted)
  if exists(select*from deleted) -- Si es un update
   select @TipoTran = 'Update'
  else                           --Si es un insert
   select @TipoTran = 'Insert'
else                             --Si es un delete
   select @TipoTran = 'Delete'

select @Tabla = 'Products'
```

```sql
select @Usuario = system_user
select @FechaTran = getdate()

--Validar si se cambia el campo UnitPrice
if update(UnitPrice)
begin
select @ValorNuevo = convert(varchar(1000),UnitPrice) from inserted
select @ValorOriginal = convert(varchar(1000),UnitPrice) from deleted
end

if exists(select*from inserted)
  select @PK = 'Clave='+convert(varchar(1000),ProductID) from inserted
else
  select @PK = 'Clave='+convert(varchar(1000),ProductID) from deleted


insert TipoCambios
(TipoTran,Tabla,PK,CampoModificado,ValorOrginal,ValorNuevo,FechaTran,Usuario)
values
(@TipoTran,@Tabla,@PK,'UnitPrice',@ValorOriginal,@ValorNuevo,@FechaTran,@Usuario)

end

select*from products
update products set unitprice = 20 where productid = 1

select*from tipoCambios

--Customers, Accion = update, Campos = Address, Phone
create trigger tr_CustomersCambios on Customers for update
as
begin
declare @TipoTran varchar(6), @Tabla varchar(128), @PK varchar(1000),
@CampoModificado varchar(128), @ValorOriginal varchar(1000),
@ValorNuevo varchar(1000), @FechaTran datetime, @Usuario varchar(128)

if exists(select*from inserted)
  if exists(select*from deleted) -- Si es un update
   select @TipoTran = 'Update'
  else                          --Si es un insert
   select @TipoTran = 'Insert'
else                           --Si es un delete
   select @TipoTran = 'Delete'

select @Tabla = 'Customers'
select @Usuario = system_user
select @FechaTran = getdate()

--Validar si se cambia el campo Address
if update(Address)
begin
 select @CampoModificado = 'Address'
 select @ValorNuevo = convert(varchar(1000),Address) from inserted
 select @ValorOriginal = convert(varchar(1000),Address) from deleted
end
--Validar si se cambia el campo Phone
else if update(Phone)
begin
```

```sql
  select @CampoModificado = 'Phone'
  select @ValorNuevo = convert(varchar(1000),Phone) from inserted
  select @ValorOriginal = convert(varchar(1000),Phone) from deleted
end

if exists(select*from inserted)
   select @PK = 'Clave='+convert(varchar(1000),CustomerID) from inserted
else
   select @PK = 'Clave='+convert(varchar(1000),CustomerID) from deleted


insert TipoCambios
(TipoTran,Tabla,PK,CampoModificado,ValorOrginal,ValorNuevo,FechaTran,Usuario)
values
(@TipoTran,@Tabla,@PK,@CampoModificado,@ValorOriginal,@ValorNuevo,@FechaTran,@Usuari
o)

end

select*from customers

update customers set address = 'Prueba 1' where CustomerID = 'ANATR'
update customers set Phone = '(171) 5555-77788' where CustomerID = 'ALFKI'

select*from TipoCambios

--Region, Accion = Insert, Delete
create trigger tr_RegionCambios on Region for Insert, Delete
as
begin
declare @TipoTran varchar(6), @Tabla varchar(128), @PK varchar(1000),
@CampoModificado varchar(128), @ValorOriginal varchar(1000),
@ValorNuevo varchar(1000), @FechaTran datetime, @Usuario varchar(128)

if exists(select*from inserted)
   if exists(select*from deleted) -- Si es un update
    select @TipoTran = 'Update'
   else                        --Si es un insert
    select @TipoTran = 'Insert'
else                           --Si es un delete
    select @TipoTran = 'Delete'

select @Tabla = 'Region'
select @Usuario = system_user
select @FechaTran = getdate()

select @CampoModificado = ' '
select @ValorNuevo = convert(varchar(1000),RegionDescription) from inserted
select @ValorOriginal = convert(varchar(1000),RegionDescription) from deleted

if exists(select*from inserted)
   select @PK = 'Clave='+convert(varchar(1000),RegionID) from inserted
else
   select @PK = 'Clave='+convert(varchar(1000),RegionID) from deleted


insert TipoCambios
(TipoTran,Tabla,PK,CampoModificado,ValorOrginal,ValorNuevo,FechaTran,Usuario)
```

```
values
(@TipoTran,@Tabla,@PK,@CampoModificado,@ValorOriginal,@ValorNuevo,@FechaTran,@Usuari
o)

end

select*from region

insert region values(5, 'Prueba')
delete region where RegionID = 5

select*from TipoCambios
```

Results | Messages

	TipoTran	Tabla	PK	CampoModificado	ValorOrginal	ValorNuevo	FechaTran	Usuario
1	Update	Products	Clave=1	UnitPrice	18.00	20.00	2022-05-16 12:08:11.790	sa
2	Update	Products	Clave=ANATR	UnitPrice	Avda. de la Constitución 2222	Prueba 1	2022-05-16 12:15:34.670	sa
3	Update	Customers	Clave=ANATR	UnitPrice	Prueba 1	Prueba 1	2022-05-16 12:18:52.760	sa
4	Update	Customers	Clave=ALFKI	UnitPrice	030-0074321	(171) 5555-77788	2022-05-16 12:18:55.273	sa
5	Insert	Region	Clave=5		NULL	Prueba	2022-05-16 12:31:51.313	sa
6	Delete	Region	Clave=5		Prueba	NULL	2022-05-16 12:32:51.320	sa