**Task No. 1:** Create

## After Insert

* After Update
* After Delete

Triggers on Shippers Table in Northwind Database. Record query and their respective results in output.

**Solution:**

**Audit Table Shipper**

create table Shippers\_Audit( ShipperID int not null, CompanyName nvarchar(40) not null, Phone nvarchar(24), ActionPerformed nvarchar(20), date datetime,

Description nvarchar(max))

# Insert

create trigger TR\_InsertAfter\_For\_Shipper on [Shippers] after insert

as begin

declare @id int,@companyname nvarchar(50),@phone nvarchar(50)

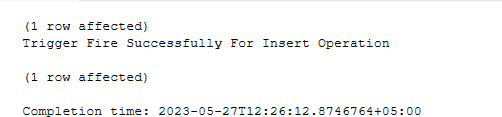
select @id=[ShipperID],@companyname=[CompanyName],@phone=[Phone] from inserted

insert into Shippers\_Audit values(@id,@companyname,@phone,'Insert',GETDATE(), cast(SUSER\_SNAME() as varchar(50))+' Inserted The Value In Shipper Table' ) print ('Trigger Fire Successfully For Insert Operation')

end

# Output:

insert into Shippers values('Time Square',094512345678)



# Update

create trigger TR\_InsertUpdate\_For\_Shipper on [Shippers] after update

as begin

declare @id int,@companyname nvarchar(50),@phone nvarchar(50), @companyname1 nvarchar(50),@phone1 nvarchar(50)

select @id=[ShipperID],@companyname=[CompanyName],@phone=[Phone] from inserted select @companyname1=[CompanyName],@phone1=[Phone] from deleted

insert into Shippers\_Audit values(@id,@companyname,@phone,'Update',GETDATE(), cast(SUSER\_SNAME() as varchar(50))+' Update The Value '+cast(@companyname1 as

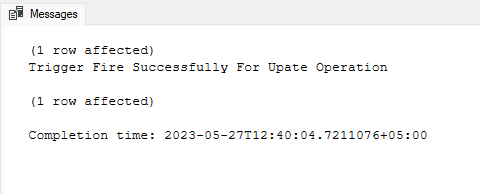
varchar(50))

+' to '+cast(@companyname as varchar(50)) +' And '+@phone1 +' To '+ @phone+' In Shipper Table' )

print ('Trigger Fire Successfully For Upate Operation') end

# Output:

update Shippers set CompanyName='New York Time',Phone='1234567789' where ShipperID=9



**Delete**

create trigger TR\_InsertDelete\_For\_Shipper on [Shippers] after delete

as begin

declare @id int,@companyname nvarchar(50),@phone nvarchar(50)

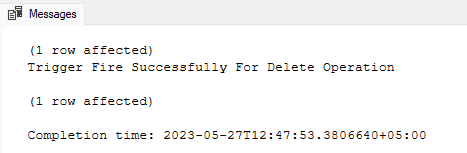
select @id=[ShipperID],@companyname=[CompanyName],@phone=[Phone] from deleted

insert into Shippers\_Audit values(@id,@companyname,@phone,'Delete',GETDATE(), cast(SUSER\_SNAME() as varchar(50))+' Delete The Row Where ID = '+cast(@id as

varchar(10))+' In Shipper Table' )

print ('Trigger Fire Successfully For Delete Operation') end

# Output:

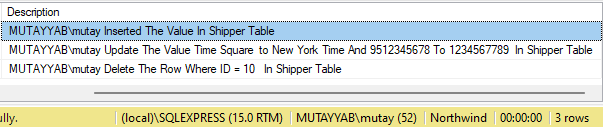
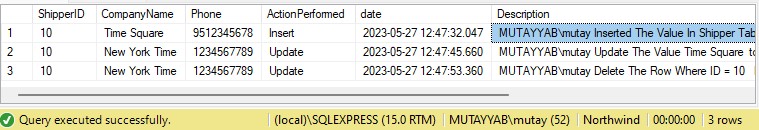


**Audit Table Shipper**

select \* from Shippers\_Audit

SABIR

SABIR



**Task No. 2:** Create

s

## Instead of Insert

* + Instead of Update
  + Instead of Delete

Triggers on Orders Table in Northwind Database. Record query and their respective results in output.

SABIR

**Solution:**

**Audit Table Order**

create table Orders\_Audit( [OrderID] int NOT NULL, [CustomerID] nchar(5) NULL, [EmployeeID] int NULL, [ShipVia] int NULL, [Freight] money NULL, [ShipName] nvarchar(40) NULL, ActionPerformed nvarchar(50)

)

# Instead of Insert

create trigger TR\_InsteadOfInsert on Orders instead of insert

as begin

declare @orderid int,@customerid nchar(5),@employeeid int,@shipvia int, @freight money, @shipname varchar(40),@action varchar(50);

select @orderid=[OrderID],@customerid=[CustomerID],@employeeid=[EmployeeID], @shipvia=[ShipVia],@freight=[Freight],@shipname=[ShipName] from inserted ; set @action='Insert';

begin transaction if @employeeid > 8 begin

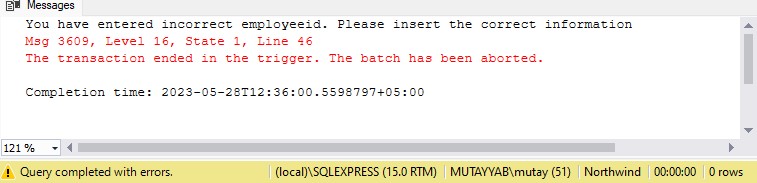
PRINT 'You have entered incorrect employeeid. Please insert the correct information'; Rollback;

end else BEGIN

insert into Orders\_Audit values(@orderid,@customerid,@employeeid,@shipvia,@freight,@shipname,@action); commit;

print 'Values Inserted - Trigger Instead of Insert Fired Successfully' end

end

**Output:** insert into Orders (CustomerID,EmployeeID,ShipVia,Freight,ShipName) values ('Wolza','10','10','00.00','Karachi Express')

SABIR

# Instead of Updatey

select @orderid=[OrderID],@customerid=[CustomerID],@employeeid=[EmployeeID], @shipvia=[ShipVia],@freight=[Freight],@shipname=[ShipName] from inserted ; set @action='Update - Instead of Trigger';

begin transaction if @freight < 50 begin

PRINT 'Too low freight. Data cannot be updated'; Rollback;

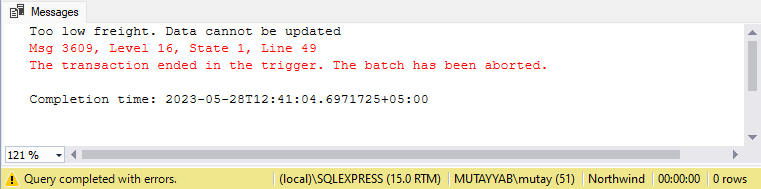
end else BEGIN

insert into Orders\_Audit values(@orderid,@customerid,@employeeid,@shipvia,@freight,@shipname,@action); commit;

print 'Values Inserted - Trigger Instead of Insert Fired Successfully'; END

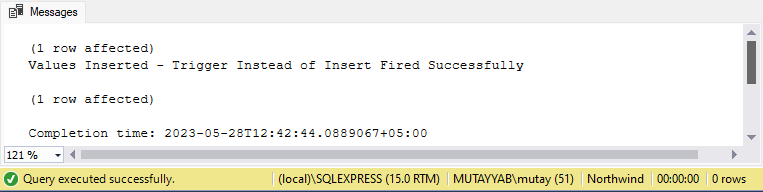
end

**Output:** update Orders set Freight=49 where OrderID=11044



SABIR

update Orders set Freight=51 where OrderID=11044



SABIR

# Instead of Delete

create trigger TR\_InsteadOfDelete on Orders instead of delete

as begin

declare @orderid int,@customerid nchar(5),@employeeid int,@shipvia int, @freight money, @shipname varchar(40),@action varchar(50);

select @orderid=[OrderID],@customerid=[CustomerID],@employeeid=[EmployeeID], @shipvia=[ShipVia],@freight=[Freight],@shipname=[ShipName] from deleted ; set @action='Delete - Instead of Trigger';

begin transaction if @freight > 70 begin

PRINT 'Too high freight. Data cannot be deleted'; Rollback;

end else

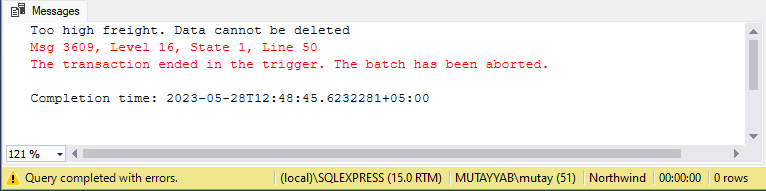
BEGIN

insert into Orders\_Audit values(@orderid,@customerid,@employeeid,@shipvia,@freight,@shipname,@action); commit;

print 'Values Deleted - Trigger Instead of Delete Fired Successfully'; END

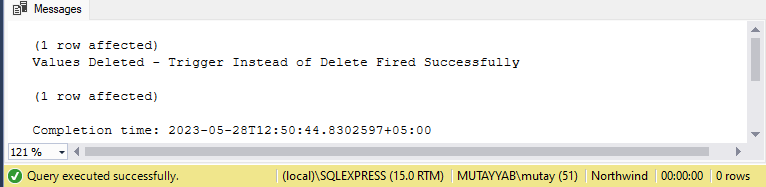
end

**Output:** delete from Orders where OrderID=10257



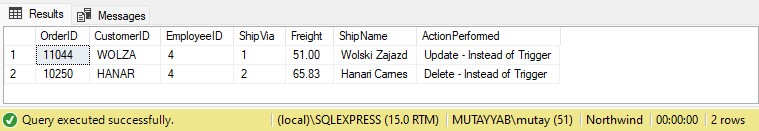
SABIR

delete from Orders where OrderID=10250



SABIR

# Audit Table Order



SABIR