Advanced Database Programming: Triggers, Events, and Business Rule Enforcement

Course: SQL Server Development

Database: AdventureWorks2022

Student: PARWINDER SINGH(N01730928)

1. Introduction

This lab focused on implementing SQL Server triggers to automate data validation, enforce business rules, and maintain audit trails within the AdventureWorks2022 database. The goal was to design and test DML and DDL triggers that ensure data integrity, compliance, and real-time automation across multiple modules such as Sales, Purchasing, Production, and Human Resources.

2. Trigger Summary Table

The table below summarizes all the triggers created and their business purpose:

Task 2 – Sales.trg_SalesOrderDetail_DiscountCheck: Validates discounts above 30%.
Task 3 – Purchasing.trg_PurchaseOrderHeader_Validation: Ensures valid order dates, freight, and tax.
Task 4 – Production.trg_Product_PriceAudit: Logs and audits price changes.
Task 5 – Sales.trg_SalesOrderDetail_InventorySync: Syncs inventory and logs low-stock alerts.
Task 6 – HumanResources.trg_Employee_TerminationAudit: Archives employee data and logs terminations.
Task 7 – trg_DatabaseSecurityMonitor: Logs schema changes (DDL trigger).
Task 8 – Transaction Handling: Added TRY/CATCH, TRAN, NOCOUNT, recursion prevention.

3. Trigger Descriptions and Results

Each trigger was tested to confirm proper functionality. Screenshots were captured showing trigger creation, error messages, and logs generated in the Automation schema.

Examples:

- Task 2 – Unauthorized discount >30% triggered rollback and inserted log into DiscountViolationLog.

- Task 4 – Price drops greater than 20% were logged in PriceReviewQueue for manager review.

- Task 6 – Deleting active employees raised error and created a record in TerminationAudit.

4. Transaction Handling and Optimization

All DML triggers (Tasks 2–6) were optimized with transaction blocks, TRY...CATCH error handling, and recursion prevention using TRIGGER_NESTLEVEL(). SET NOCOUNT ON was added to avoid rowcount interference. BEGIN TRAN, COMMIT, and ROLLBACK ensure atomic and consistent operations.

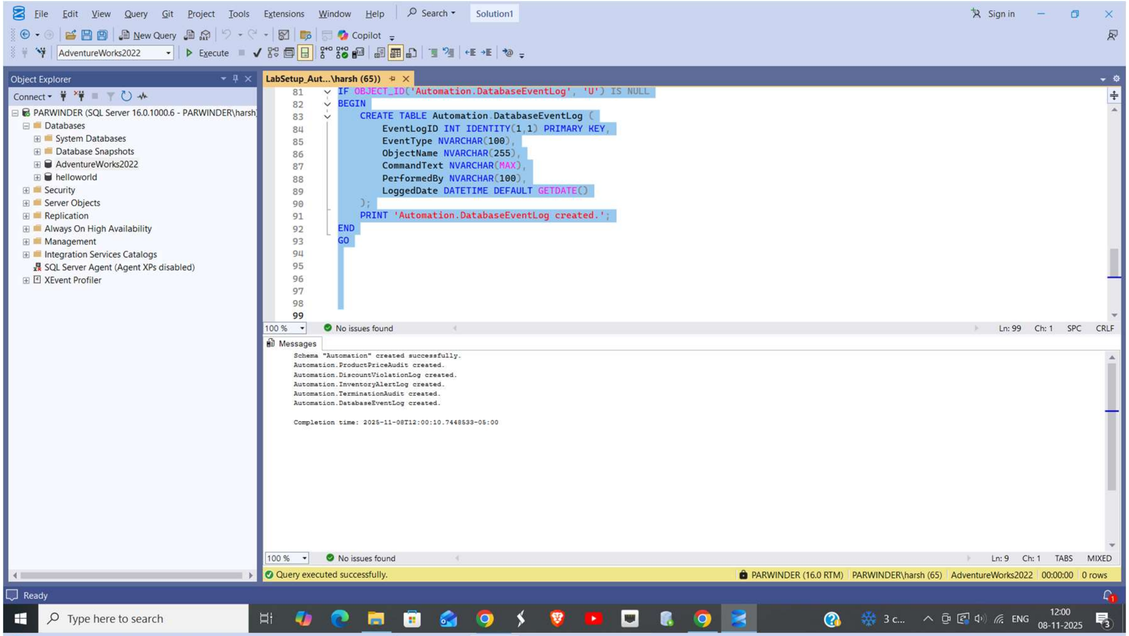
5. Performance Observations

Performance improved by restricting updates using IF UPDATE(ListPrice) in Product triggers, reducing unnecessary executions. Adding SET NOCOUNT ON minimized message traffic and improved speed during bulk operations.

6. Learning Outcomes

This lab enhanced understanding of event-driven database programming and transaction control. I learned how to use triggers for automation, auditing, and business rule enforcement, while ensuring rollback safety and compliance logging.

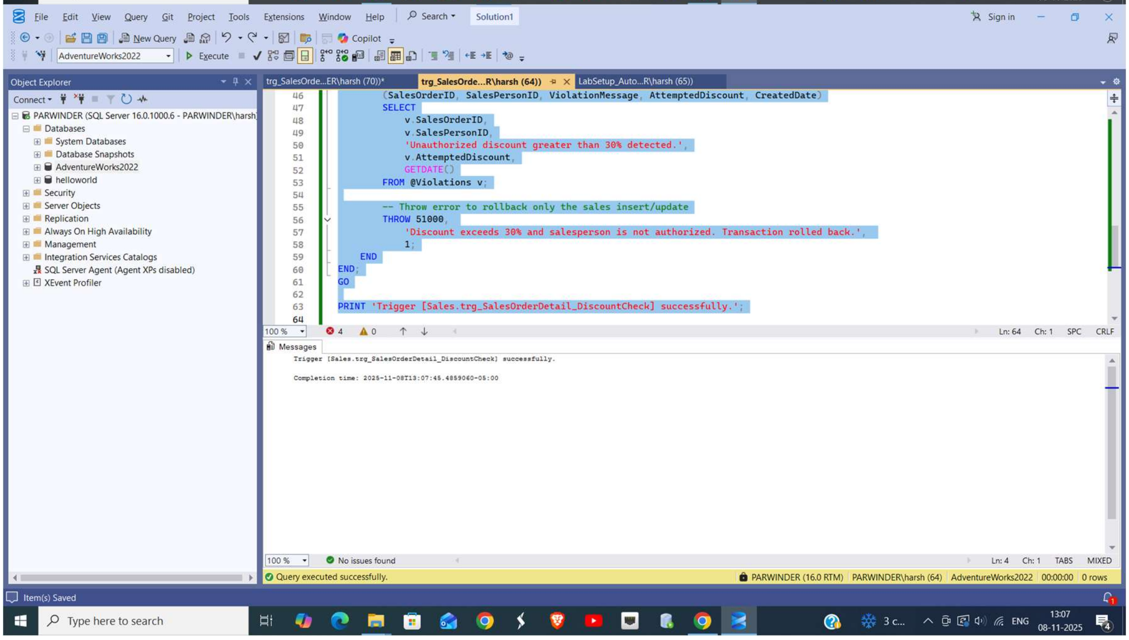
7. SCREENSHOTS-



```sql
IF OBJECT_ID('Automation.DatabaseEventLog', 'U') IS NULL
BEGIN
    CREATE TABLE Automation.DatabaseEventLog (
        EventLogID INT IDENTITY(1,1) PRIMARY KEY,
        EventType NVARCHAR(100),
        ObjectName NVARCHAR(255),
        CommandText NVARCHAR(MAX),
        PerformedBy NVARCHAR(100),
        LoggedDate DATETIME DEFAULT GETDATE()
    );
    PRINT 'Automation.DatabaseEventLog created.';
END
GO
```

```
Schema "Automation" created successfully.
Automation.ProductPriceAudit created.
Automation.DiscountViolationLog created.
Automation.InventoryAlertLog created.
Automation.TerminationAudit created.
Automation.DatabaseEventLog created.

Completion time: 2025-11-08T12:00:10.7448533-05:00
```

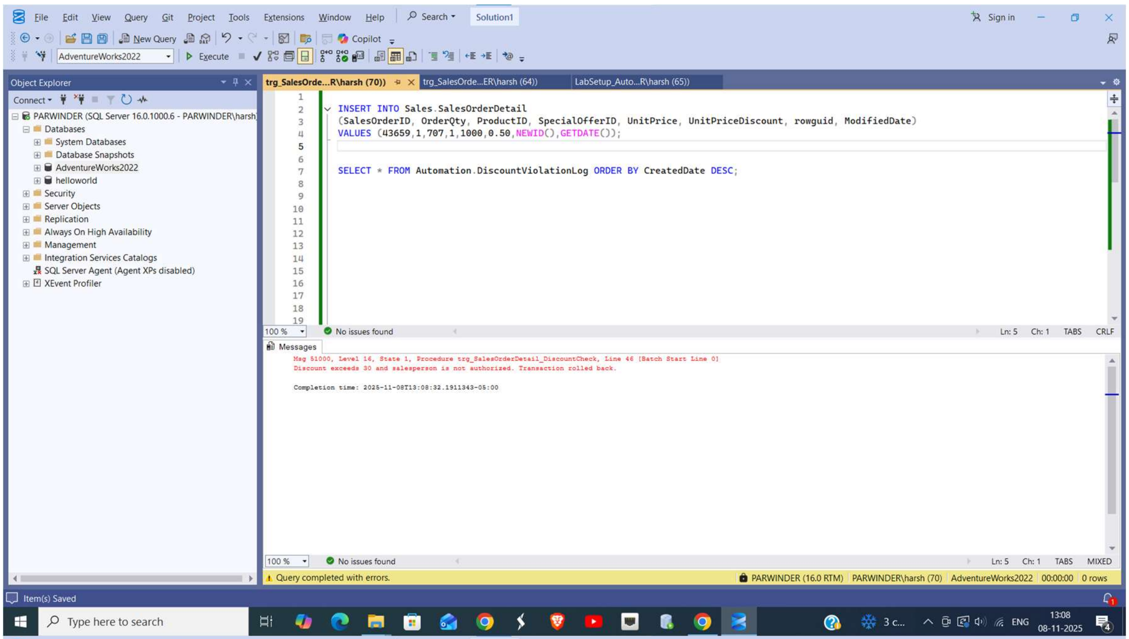
Query executed successfully.



```sql
            (SalesOrderID, SalesPersonID, ViolationMessage, AttemptedDiscount, CreatedDate)
            SELECT
                v.SalesOrderID,
                v.SalesPersonID,
                'Unauthorized discount greater than 30% detected.',
                v.AttemptedDiscount,
                GETDATE()
            FROM @Violations v;

            -- Throw error to rollback only the sales insert/update
            THROW 51000,
                'Discount exceeds 30% and salesperson is not authorized. Transaction rolled back.',
                1;
        END
    END;
    GO

    PRINT 'Trigger [Sales.trg_SalesOrderDetail_DiscountCheck] successfully.';
```

```
Trigger [Sales.trg_SalesOrderDetail_DiscountCheck] successfully.

Completion time: 2025-11-08T13:07:45.4859060-05:00
```

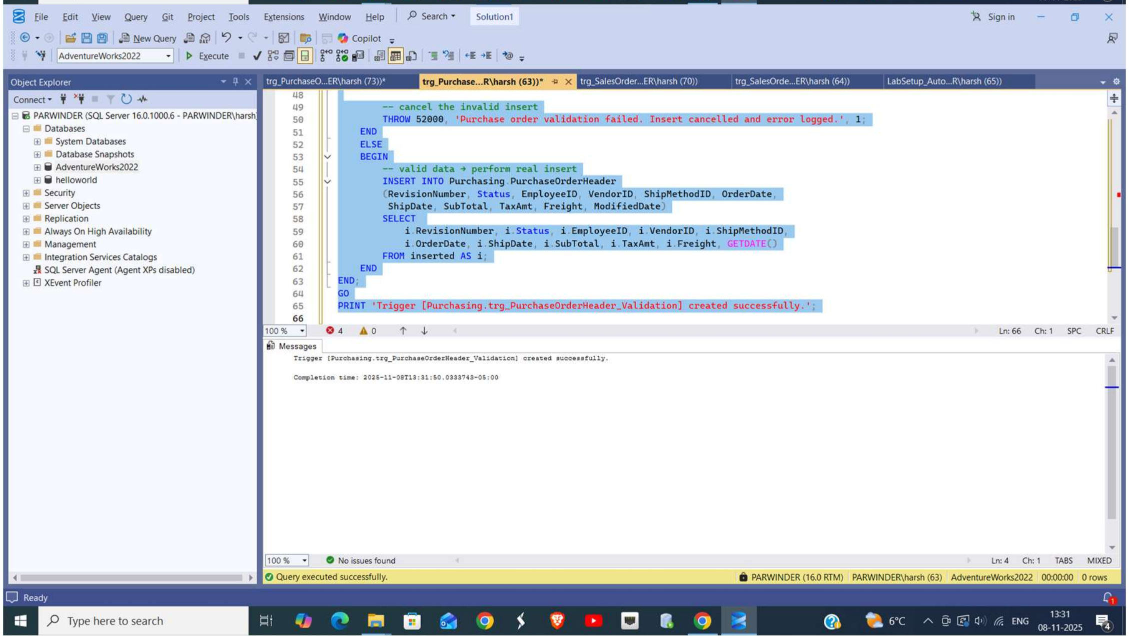
Query executed successfully.


File Edit View Query Git Project Tools Extensions Window Help
Search
Solution1
Sign in
New Query
Copilot
AdventureWorks2022
Execute
Object Explorer
Connect
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh
Databases
System Databases
Database Snapshots
AdventureWorks2022
helloworld
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler
trg_SalesOrde...R\harsh (70))
trg_SalesOrde...ER\harsh (64))
LabSetup_Auto...R\harsh (65))
100 %
No issues found
Ln: 5 Ch: 1 TABS CRLF
Messages
Ln: 5 Ch: 1 TABS MIXED
PARWINDER (16.0 RTM) PARWINDER\harsh (70) AdventureWorks2022 00:00:00 0 rows
Item(s) Saved
Type here to search
3 c...
ENG
13:08
08-11-2025

```
INSERT INTO Sales.SalesOrderDetail
(SalesOrderID, OrderQty, ProductID, SpecialOfferID, UnitPrice, UnitPriceDiscount, rowguid, ModifiedDate)
VALUES (43659,1,707,1,1000,0.50,NEWID(),GETDATE());


SELECT * FROM Automation.DiscountViolationLog ORDER BY CreatedDate DESC;
```

```
Msg 51000, Level 16, State 1, Procedure trg_SalesOrderDetail_DiscountCheck, Line 46 [Batch Start Line 0]
Discount exceeds 30 and salesperson is not authorized. Transaction rolled back.

Completion time: 2025-11-08T13:08:32.1911343-05:00
```

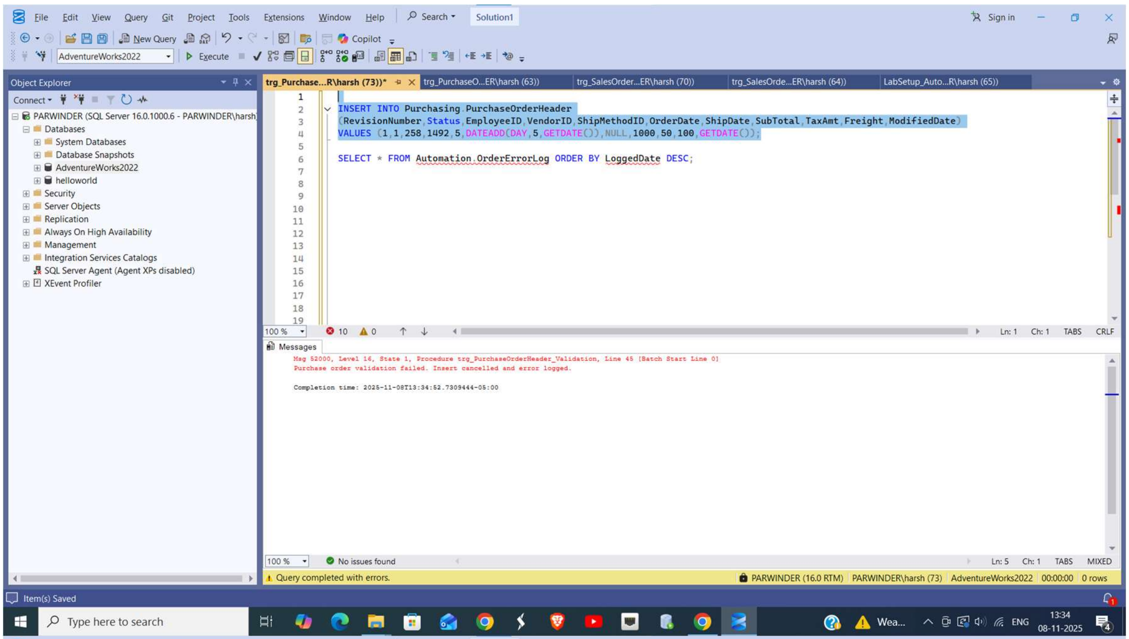
Query completed with errors.


File Edit View Query Git Project Tools Extensions Window Help
Search
Solution1
Sign in
New Query
Copilot
AdventureWorks2022
Execute
Object Explorer
Connect
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh
Databases
System Databases
Database Snapshots
AdventureWorks2022
helloworld
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler
trg_PurchaseO...ER\harsh (73))*
trg_Purchase...R\harsh (63))*
trg_SalesOrder...ER\harsh (70))
trg_SalesOrde...ER\harsh (64))
LabSetup_Auto...R\harsh (65))
100 %
4
0
Ln: 66 Ch: 1 SPC CRLF
Messages
No issues found
Ln: 4 Ch: 1 TABS MIXED
PARWINDER (16.0 RTM) PARWINDER\harsh (63) AdventureWorks2022 00:00:00 0 rows
Ready
Type here to search
6°C
ENG
13:31
08-11-2025

```

            -- cancel the invalid insert
            THROW 52000, 'Purchase order validation failed. Insert cancelled and error logged.', 1;
        END
        ELSE
        BEGIN
            -- valid data → perform real insert
            INSERT INTO Purchasing.PurchaseOrderHeader
            (RevisionNumber, Status, EmployeeID, VendorID, ShipMethodID, OrderDate,
             ShipDate, SubTotal, TaxAmt, Freight, ModifiedDate)
            SELECT
                i.RevisionNumber, i.Status, i.EmployeeID, i.VendorID, i.ShipMethodID,
                i.OrderDate, i.ShipDate, i.SubTotal, i.TaxAmt, i.Freight, GETDATE()
            FROM inserted AS i;
        END
END;
GO
PRINT 'Trigger [Purchasing.trg_PurchaseOrderHeader_Validation] created successfully.';
```

```
Trigger [Purchasing.trg_PurchaseOrderHeader_Validation] created successfully.

Completion time: 2025-11-08T13:31:50.0333743-05:00
```

Query executed successfully.

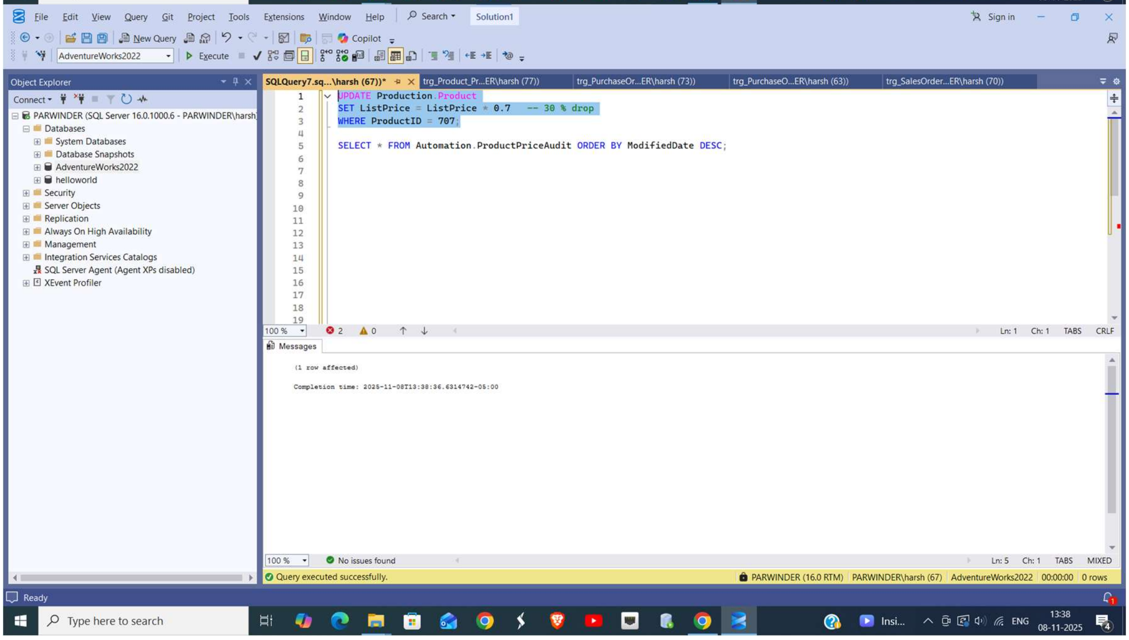


trg_Purchase...R\harsh (73))*
trg_PurchaseO...ER\harsh (63))
trg_SalesOrder...ER\harsh (70))
trg_SalesOrde...ER\harsh (64))
LabSetup_Auto...R\harsh (65))
Object Explorer
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh)
AdventureWorks2022
Item(s) Saved

```sql
INSERT INTO Purchasing.PurchaseOrderHeader
(RevisionNumber,Status,EmployeeID,VendorID,ShipMethodID,OrderDate,ShipDate,SubTotal,TaxAmt,Freight,ModifiedDate)
VALUES (1,1,258,1492,5,DATEADD(DAY,5,GETDATE()),NULL,1000,50,100,GETDATE());

SELECT * FROM Automation.OrderErrorLog ORDER BY LoggedDate DESC;
```

```
Msg 52000, Level 16, State 1, Procedure trg_PurchaseOrderHeader_Validation, Line 45 [Batch Start Line 0]
Purchase order validation failed. Insert cancelled and error logged.

Completion time: 2025-11-08T13:34:52.7309444-05:00
```

Query completed with errors.

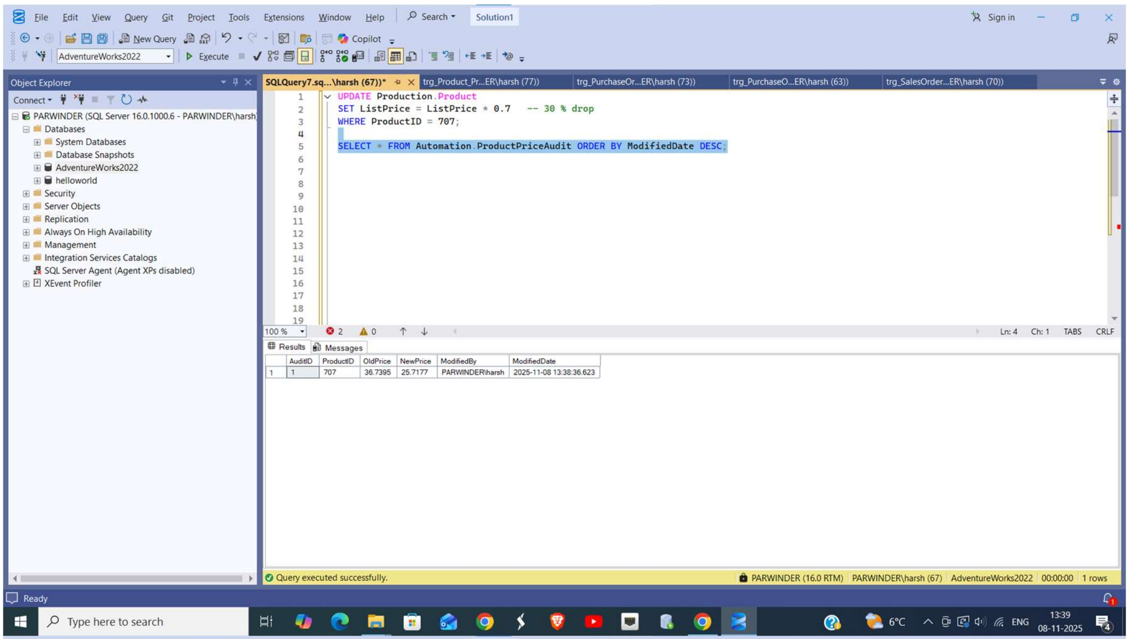


SQLQuery7.sq...\harsh (67))*
trg_Product_Pr...ER\harsh (77))
trg_PurchaseOr...ER\harsh (73))
trg_PurchaseO...ER\harsh (63))
trg_SalesOrder...ER\harsh (70))
Object Explorer
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh)
AdventureWorks2022
Ready

```sql
UPDATE Production.Product
SET ListPrice = ListPrice * 0.7   -- 30 % drop
WHERE ProductID = 707;

SELECT * FROM Automation.ProductPriceAudit ORDER BY ModifiedDate DESC;
```

```
(1 row affected)

Completion time: 2025-11-08T13:38:36.6314742-05:00
```

Query executed successfully.

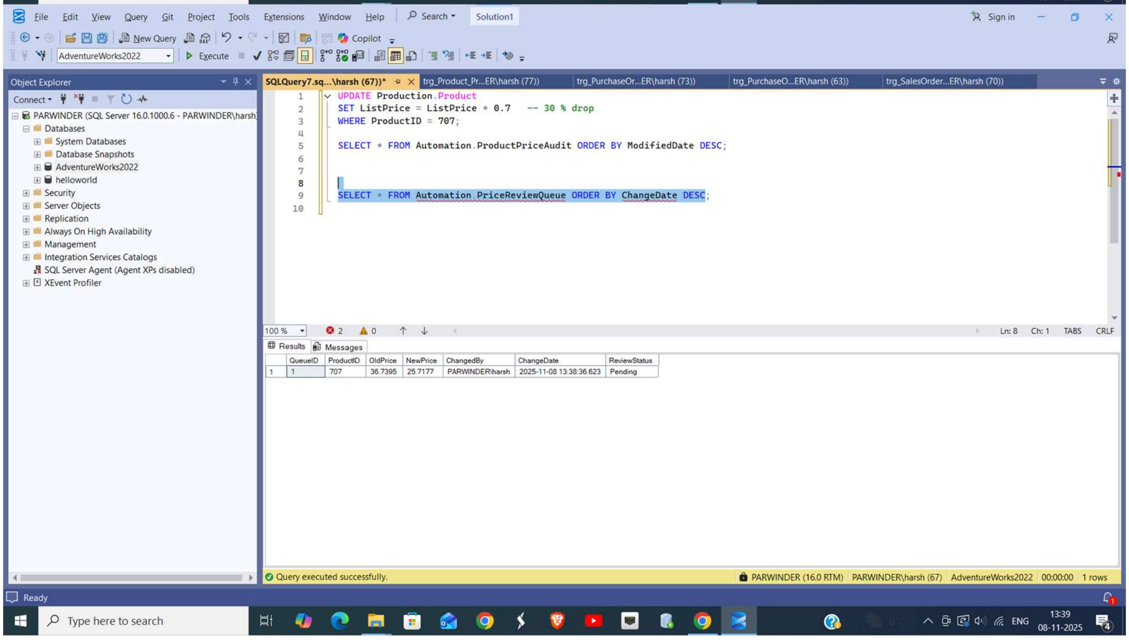


SQLQuery7.sq...\harsh (67))*
trg_Product_Pr...ER\harsh (77))
trg_PurchaseOr...ER\harsh (73))
trg_PurchaseO...ER\harsh (63))
trg_SalesOrder...ER\harsh (70))
Object Explorer
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh
Databases
System Databases
Database Snapshots
AdventureWorks2022
helloworld
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler
Results
Messages

```sql
UPDATE Production.Product
SET ListPrice = ListPrice * 0.7   -- 30 % drop
WHERE ProductID = 707;

SELECT * FROM Automation.ProductPriceAudit ORDER BY ModifiedDate DESC;
```

	AuditID	ProductID	OldPrice	NewPrice	ModifiedBy	ModifiedDate
1	1	707	36.7395	25.7177	PARWINDER\harsh	2025-11-08 13:38:36.623

Query executed successfully. PARWINDER (16.0 RTM) PARWINDER\harsh (67) AdventureWorks2022 00:00:00 1 rows

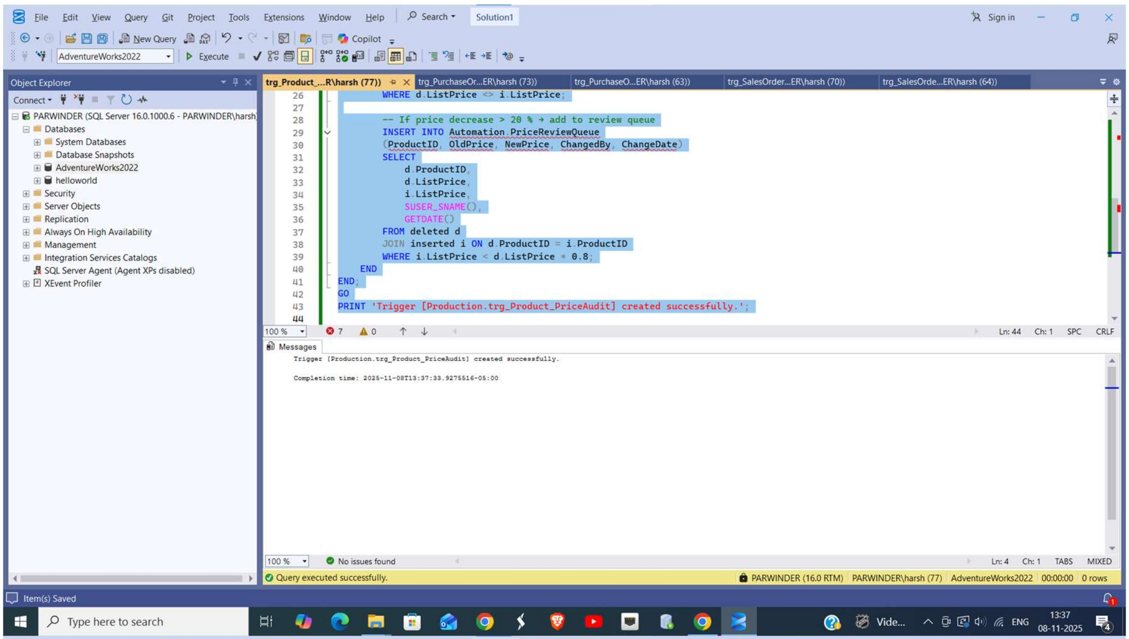


SQLQuery7.sq...\harsh (67))*
trg_Product_Pr...ER\harsh (77))
trg_PurchaseOr...ER\harsh (73))
trg_PurchaseO...ER\harsh (63))
trg_SalesOrder...ER\harsh (70))
Results
Messages

```sql
UPDATE Production.Product
SET ListPrice = ListPrice * 0.7   -- 30 % drop
WHERE ProductID = 707;

SELECT * FROM Automation.ProductPriceAudit ORDER BY ModifiedDate DESC;


SELECT * FROM Automation.PriceReviewQueue ORDER BY ChangeDate DESC;
```

	QueueID	ProductID	OldPrice	NewPrice	ChangedBy	ChangeDate	ReviewStatus
1	1	707	36.7395	25.7177	PARWINDER\harsh	2025-11-08 13:38:36.623	Pending

Query executed successfully. PARWINDER (16.0 RTM) PARWINDER\harsh (67) AdventureWorks2022 00:00:00 1 rows

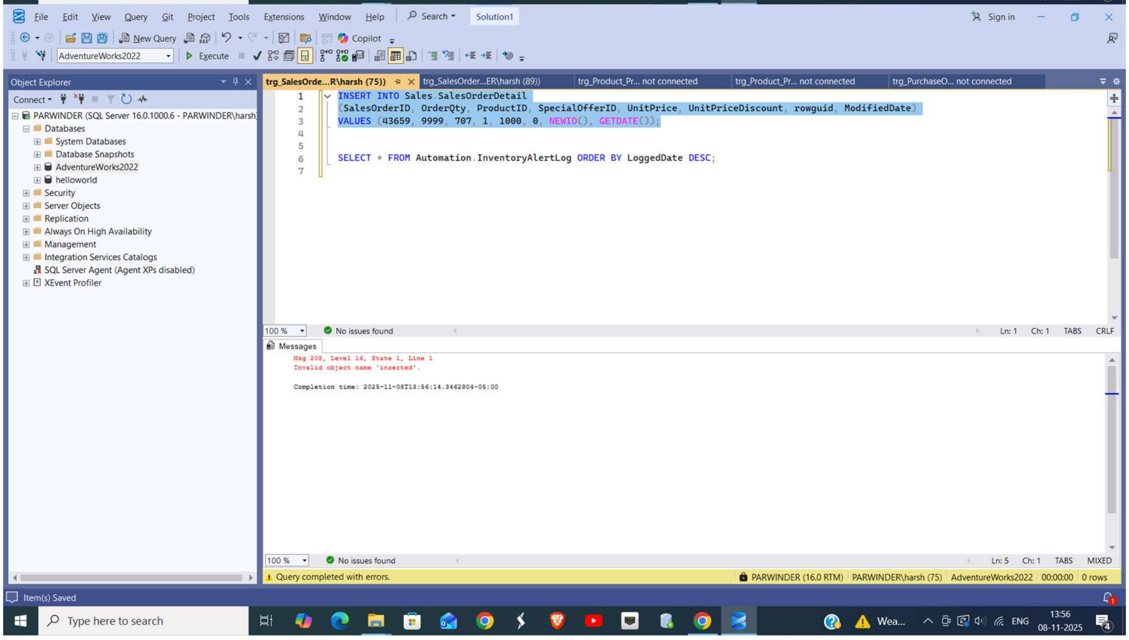


File Edit View Query Git Project Tools Extensions Window Help
Search
Solution1
Sign in
New Query
AdventureWorks2022
Execute
Copilot
Object Explorer
Connect
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh)
Databases
System Databases
Database Snapshots
AdventureWorks2022
helloworld
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler
trg_Product_...R\harsh (77))
trg_PurchaseOr...ER\harsh (73))
trg_PurchaseO...ER\harsh (63))
trg_SalesOrder...ER\harsh (70))
trg_SalesOrde...ER\harsh (64))
100 %
7
0
Ln: 44 Ch: 1 SPC CRLF
Messages
No issues found
Ln: 4 Ch: 1 TABS MIXED
PARWINDER (16.0 RTM) PARWINDER\harsh (77) AdventureWorks2022 00:00:00 0 rows
Item(s) Saved
Type here to search
Vide...
ENG
13:37
08-11-2025

```sql
            WHERE d.ListPrice <> i.ListPrice;

            -- If price decrease > 20 % → add to review queue
            INSERT INTO Automation.PriceReviewQueue
            (ProductID, OldPrice, NewPrice, ChangedBy, ChangeDate)
            SELECT
                d.ProductID,
                d.ListPrice,
                i.ListPrice,
                SUSER_SNAME(),
                GETDATE()
            FROM deleted d
            JOIN inserted i ON d.ProductID = i.ProductID
            WHERE i.ListPrice < d.ListPrice * 0.8;
        END
END;
GO
PRINT 'Trigger [Production.trg_Product_PriceAudit] created successfully.';
```

```
Trigger [Production.trg_Product_PriceAudit] created successfully.

Completion time: 2025-11-08T13:37:33.9275516-05:00
```

Query executed successfully.

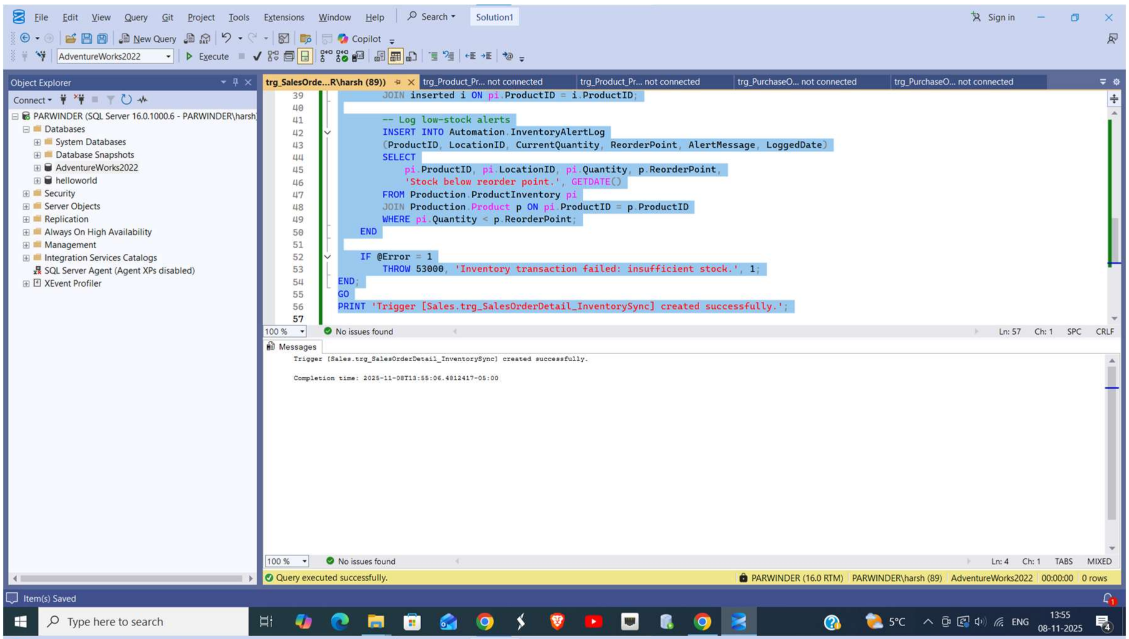


File Edit View Query Git Project Tools Extensions Window Help
Search
Solution1
Sign in
New Query
AdventureWorks2022
Execute
Copilot
Object Explorer
Connect
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh)
Databases
System Databases
Database Snapshots
AdventureWorks2022
helloworld
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler
trg_SalesOrde...R\harsh (75))
trg_SalesOrder...ER\harsh (89))
trg_Product_Pr... not connected
trg_Product_Pr... not connected
trg_PurchaseO... not connected
100 %
No issues found
Ln: 1 Ch: 1 TABS CRLF
Messages
Ln: 5 Ch: 1 TABS MIXED
PARWINDER (16.0 RTM) PARWINDER\harsh (75) AdventureWorks2022 00:00:00 0 rows
Item(s) Saved
Type here to search
Wea...
ENG
13:56
08-11-2025

```sql
INSERT INTO Sales.SalesOrderDetail
(SalesOrderID, OrderQty, ProductID, SpecialOfferID, UnitPrice, UnitPriceDiscount, rowguid, ModifiedDate)
VALUES (43659, 9999, 707, 1, 1000, 0, NEWID(), GETDATE());


SELECT * FROM Automation.InventoryAlertLog ORDER BY LoggedDate DESC;
```

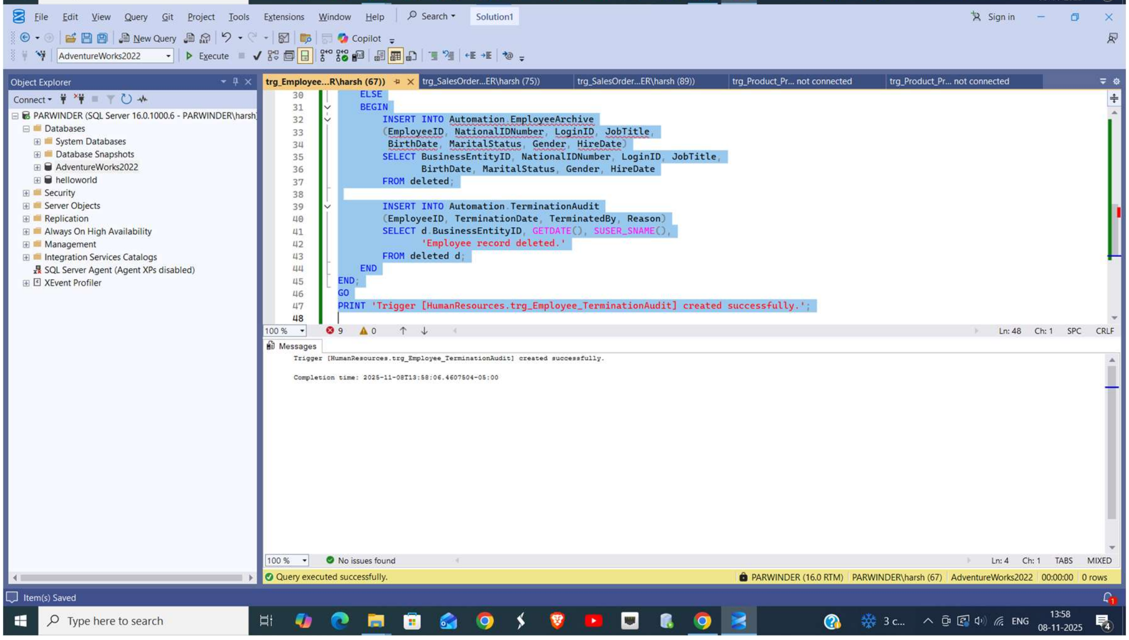
```
Msg 208, Level 16, State 1, Line 1
Invalid object name 'inserted'.

Completion time: 2025-11-08T13:56:14.3462804-05:00
```

Query completed with errors.


trg_SalesOrde...R\harsh (89))
trg_Product_Pr... not connected
trg_Product_Pr... not connected
trg_PurchaseO... not connected
trg_PurchaseO... not connected
Object Explorer
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh)
Databases
System Databases
Database Snapshots
AdventureWorks2022
helloworld
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler
No issues found
Messages
Query executed successfully.
PARWINDER (16.0 RTM) PARWINDER\harsh (89) AdventureWorks2022 00:00:00 0 rows
Item(s) Saved
Type here to search
13:55
08-11-2025

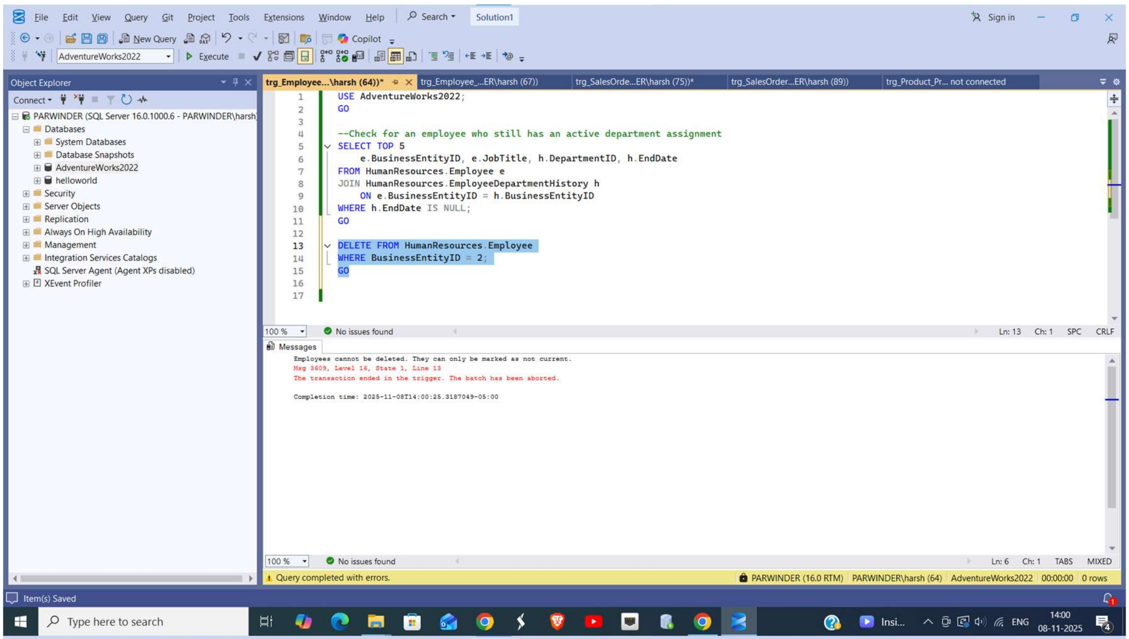
```sql
        JOIN inserted i ON pi.ProductID = i.ProductID;

        -- Log low-stock alerts
        INSERT INTO Automation.InventoryAlertLog
        (ProductID, LocationID, CurrentQuantity, ReorderPoint, AlertMessage, LoggedDate)
        SELECT
            pi.ProductID, pi.LocationID, pi.Quantity, p.ReorderPoint,
            'Stock below reorder point.', GETDATE()
        FROM Production.ProductInventory pi
        JOIN Production.Product p ON pi.ProductID = p.ProductID
        WHERE pi.Quantity < p.ReorderPoint;
    END

    IF @Error = 1
        THROW 53000, 'Inventory transaction failed: insufficient stock.', 1;
END;
GO
PRINT 'Trigger [Sales.trg_SalesOrderDetail_InventorySync] created successfully.';
```

```
Trigger [Sales.trg_SalesOrderDetail_InventorySync] created successfully.

Completion time: 2025-11-08T13:55:06.4812417-05:00
```


trg_Employee...R\harsh (67))
trg_SalesOrder...ER\harsh (75))
trg_SalesOrder...ER\harsh (89))
trg_Product_Pr... not connected
trg_Product_Pr... not connected
Object Explorer
Messages
Query executed successfully.
PARWINDER (16.0 RTM) PARWINDER\harsh (67) AdventureWorks2022 00:00:00 0 rows
Item(s) Saved
Type here to search
13:58
08-11-2025

```sql
    ELSE
    BEGIN
        INSERT INTO Automation.EmployeeArchive
        (EmployeeID, NationalIDNumber, LoginID, JobTitle,
         BirthDate, MaritalStatus, Gender, HireDate)
        SELECT BusinessEntityID, NationalIDNumber, LoginID, JobTitle,
               BirthDate, MaritalStatus, Gender, HireDate
        FROM deleted;

        INSERT INTO Automation.TerminationAudit
        (EmployeeID, TerminationDate, TerminatedBy, Reason)
        SELECT d.BusinessEntityID, GETDATE(), SUSER_SNAME(),
               'Employee record deleted.'
        FROM deleted d;
    END
END;
GO
PRINT 'Trigger [HumanResources.trg_Employee_TerminationAudit] created successfully.';
```

```
Trigger [HumanResources.trg_Employee_TerminationAudit] created successfully.

Completion time: 2025-11-08T13:58:06.4607504-05:00
```



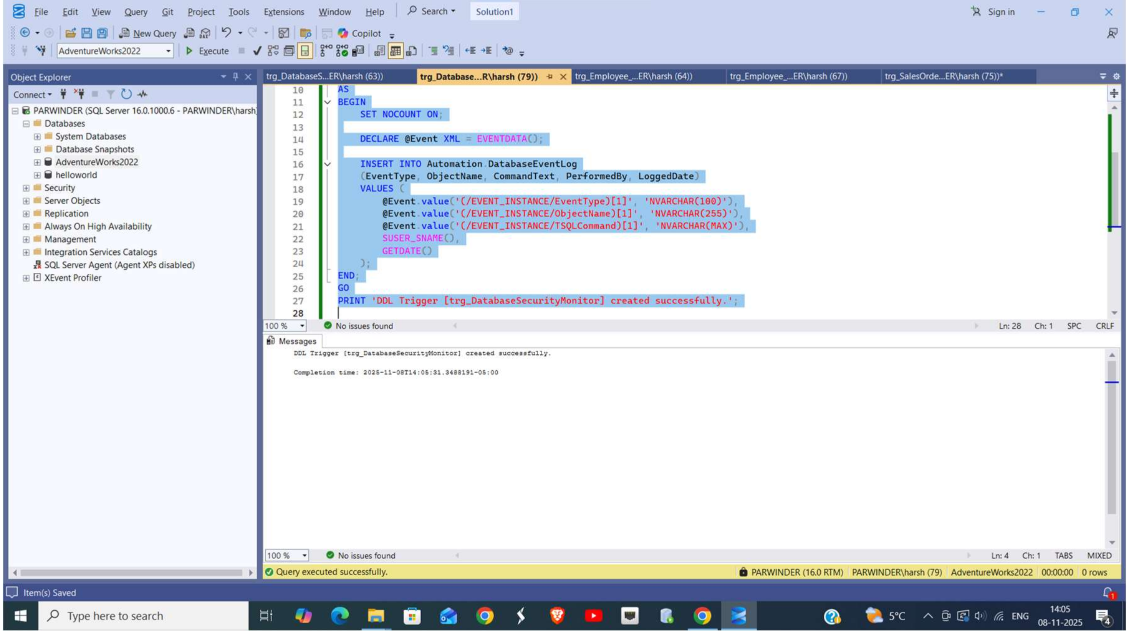
trg_Employee...\harsh (64))*
trg_Employee_...ER\harsh (67))
trg_SalesOrde...ER\harsh (75))*
trg_SalesOrder...ER\harsh (89))
trg_Product_Pr... not connected
Object Explorer
PARWINDER (SQL Server 16.0.1000.6 - PARWINDER\harsh)
Databases
System Databases
Database Snapshots
AdventureWorks2022
helloworld
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler

```sql
USE AdventureWorks2022;
GO

--Check for an employee who still has an active department assignment
SELECT TOP 5
    e.BusinessEntityID, e.JobTitle, h.DepartmentID, h.EndDate
FROM HumanResources.Employee e
JOIN HumanResources.EmployeeDepartmentHistory h
    ON e.BusinessEntityID = h.BusinessEntityID
WHERE h.EndDate IS NULL;
GO

DELETE FROM HumanResources.Employee
WHERE BusinessEntityID = 2;
GO
```

Messages

```
Employees cannot be deleted. They can only be marked as not current.
Msg 3609, Level 16, State 1, Line 13
The transaction ended in the trigger. The batch has been aborted.

Completion time: 2025-11-08T14:00:25.3187049-05:00
```

Query completed with errors.
PARWINDER (16.0 RTM) | PARWINDER\harsh (64) | AdventureWorks2022 | 00:00:00 | 0 rows
14:00
08-11-2025



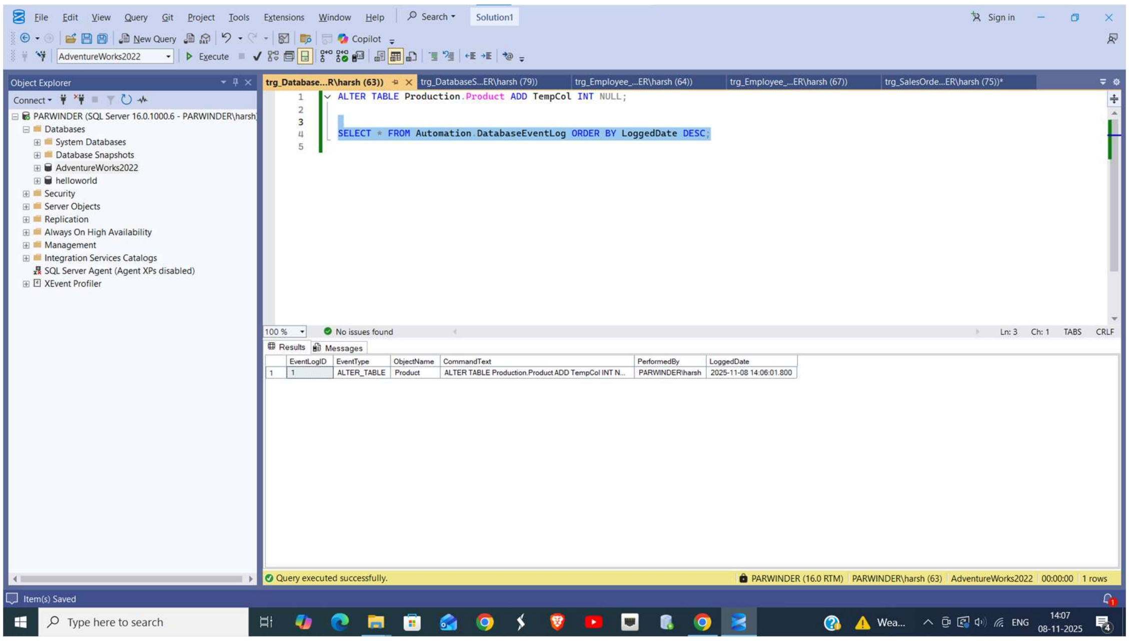
trg_DatabaseS...ER\harsh (63))
trg_Database...R\harsh (79))
trg_Employee_...ER\harsh (64))
trg_Employee_...ER\harsh (67))
trg_SalesOrde...ER\harsh (75))*

```sql
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @Event XML = EVENTDATA();

    INSERT INTO Automation.DatabaseEventLog
    (EventType, ObjectName, CommandText, PerformedBy, LoggedDate)
    VALUES (
        @Event.value('(/EVENT_INSTANCE/EventType)[1]', 'NVARCHAR(100)'),
        @Event.value('(/EVENT_INSTANCE/ObjectName)[1]', 'NVARCHAR(255)'),
        @Event.value('(/EVENT_INSTANCE/TSQLCommand)[1]', 'NVARCHAR(MAX)'),
        SUSER_SNAME(),
        GETDATE()
    );
END;
GO
PRINT 'DDL Trigger [trg_DatabaseSecurityMonitor] created successfully.';
```

Messages

```
DDL Trigger [trg_DatabaseSecurityMonitor] created successfully.

Completion time: 2025-11-08T14:05:31.3488191-05:00
```

Query executed successfully.
PARWINDER (16.0 RTM) | PARWINDER\harsh (79) | AdventureWorks2022 | 00:00:00 | 0 rows
14:05
08-11-2025



```sql
ALTER TABLE Production.Product ADD TempCol INT NULL;


SELECT * FROM Automation.DatabaseEventLog ORDER BY LoggedDate DESC;

```

No issues found

	EventLogID	EventType	ObjectName	CommandText	PerformedBy	LoggedDate
1	1	ALTER_TABLE	Product	ALTER TABLE Production.Product ADD TempCol INT N...	PARWINDER\harsh	2025-11-08 14:06:01.800

Query executed successfully.

PARWINDER (16.0 RTM) | PARWINDER\harsh (63) | AdventureWorks2022 | 00:00:00 | 1 rows