Absolutely! Here's a **cleaned-up and finalized version** of your schema with **warranty tracking integrated properly**, including logic to calculate warranty based on delivery date. It includes:

* Adjustments to the schema
* Trigger and/or stored procedure to insert warranty on delivery
* Optional view for tracking warranty

**✅ 1. SCHEMA CHANGES**

**🔹 Add Warranty Period to Material**

ALTER TABLE Material

ADD Warranty\_Period\_Months INT DEFAULT 12; -- Default 12 months warranty

**✅ 2. WARRANTY TABLE (Already Present)**

Just make sure this is exactly as below:

CREATE TABLE Warranty (

Warranty\_ID INT IDENTITY(1,1) PRIMARY KEY,

Material\_ID INT NOT NULL,

Serial\_Number NVARCHAR(255) NOT NULL,

Start\_Date DATE NOT NULL,

End\_Date DATE NOT NULL,

Status NVARCHAR(50) CHECK (Status IN ('Under Warranty', 'Expired', 'Replaced')),

FOREIGN KEY (Material\_ID) REFERENCES Material(Material\_ID),

FOREIGN KEY (Serial\_Number) REFERENCES Stock(Serial\_Number)

);

**✅ 3. STORED PROCEDURE: Add Warranty on Delivery**

This procedure inserts a new warranty record when an item is delivered.

CREATE PROCEDURE InsertWarrantyOnDelivery

AS

BEGIN

INSERT INTO Warranty (Material\_ID, Serial\_Number, Start\_Date, End\_Date, Status)

SELECT

ci.Material\_ID,

ci.Serial\_Number,

c.Challan\_Date,

DATEADD(MONTH, ISNULL(m.Warranty\_Period\_Months, 12), c.Challan\_Date),

'Under Warranty'

FROM Challan\_Items ci

JOIN Challan c ON ci.Challan\_ID = c.Challan\_ID

JOIN Material m ON ci.Material\_ID = m.Material\_ID

WHERE m.Requires\_Serial\_Number = 'Yes'

AND ci.Serial\_Number IS NOT NULL

AND NOT EXISTS (

SELECT 1 FROM Warranty w

WHERE w.Serial\_Number = ci.Serial\_Number

);

END;

🔁 You should call this procedure **after every delivery (Challan entry)**.

**✅ 4. VIEW: Track Warranty Status**

Create a view for easily checking warranty info:

CREATE VIEW vw\_Warranty\_Status AS

SELECT

w.Warranty\_ID,

m.Materials\_Name,

w.Serial\_Number,

w.Start\_Date,

w.End\_Date,

DATEDIFF(DAY, GETDATE(), w.End\_Date) AS Days\_Remaining,

w.Status

FROM Warranty w

JOIN Material m ON w.Material\_ID = m.Material\_ID;

**✅ 5. SCHEDULED JOB (Optional): Expire Warranty**

To update warranty status daily:

UPDATE Warranty

SET Status = 'Expired'

WHERE End\_Date < GETDATE()

AND Status = 'Under Warranty';

You can automate this using a **SQL Agent Job**.

**✅ Summary of Changes**

| **Component** | **Description** |
| --- | --- |
| Warranty\_Period\_Months | New column in Material to define warranty length |
| InsertWarrantyOnDelivery | Procedure to insert warranty during delivery |
| vw\_Warranty\_Status | View to monitor warranty status |
| Optional Agent Job | To auto-update Expired warranties |

Would you also like a trigger to run the warranty insert *automatically* after inserting into Challan\_Items, or do you plan to handle it from your application/backend logic?