

"Involve Me and I Will Understand!"

2-Day Team Challenge: **Mastering Stored Procedures & Triggers**

➤ Workshop Philosophy:

This workshop is built on the belief that true understanding comes through active participation.

*Rather than just listening, you will **explore, test, troubleshoot, and build**. The goal is not just to learn — but to deeply understand by doing and practice 😊.*

➤ Objective:

You are now in the shoes of a developer working under a project manager's request to **quickly understand and implement** SQL features in a real project. This is your **team challenge** to **deeply explore, test, and document** your understanding.

Your goal is to **understand, explain, and demonstrate** the use of **Stored Procedures and Triggers** in SQL through:

- **Clear explanation**
- **Practical application**
- **Real-life business examples**

You are expected to **try everything, fail, fix, and learn** — just like real development.

➤ Format:

- Work in **pairs** (2 trainees per team).
- At the end, we will **select the best explanation and use it as a template** for future batches. Your names will be mentioned as contributors!

➤ Deliverables:

Each pair must:

1. **Explore & Research**

- What is a Stored Procedure?

- What is a Trigger?
- Types of Triggers
- Benefits and limitations of both

2. Apply in Practice

Use a real-life case (from your own project or choose a realistic system like: E-commerce, Booking App, Library, HR System, etc.).

- Create meaningful stored procedures (e.g., auto-calculate salary, approve order, etc.)
- Create relevant triggers (e.g., prevent delete if unpaid invoice exists, log updates)

3. Explain Clearly

- Smooth, easy-to-understand explanation
- Step-by-step walkthrough of your SQL code
- Use comments in SQL to explain logic
- Clear breakdown of each part
- What challenges you faced and how you solved them
- **List of references and sources used** (blogs, articles, videos – not only ChatGPT!)
- ErrorLog.txt or section in README: Describe errors you encountered and what fixed them
- Provide screenshots or screen recordings of the output if possible

4. Upload to GitHub

- Create a repository named:
SP_Trigger_Challenge_TeamX (replace X with your team number or names)
- Include:
 - README.md explaining everything clearly
 - SQL scripts: All procedures and triggers
 - Optional but encouraged: short demo video or screenshots

➤ What to Explore and Practice:

Stored Procedures

- Create, execute, alter, and drop stored procedures
- Input/output parameters
- Using conditions, variables, loops
- Use in scenarios like:
 - Approving orders
 - Auto-generating payroll
 - Validating registrations

Triggers

- AFTER, BEFORE, INSTEAD OF triggers
- INSERT, UPDATE, DELETE triggers
- Use triggers to:
 - Log changes to audit tables
 - Prevent unwanted deletions
 - Update stock levels automatically
 - Send alerts (simulate logic)

➤ Exploration Checklist

Test Everything – Not Just Theory!

Don't just copy theory — **show me your understanding through testing.**

I want to see different cases — not only the "happy path".

You must **TRY and DOCUMENT** all these:

- What happens if procedure has a syntax error?
- What if trigger creates a loop?
- Can a procedure call another procedure?

- What errors did you face while altering a trigger?
- What if data doesn't meet the condition inside trigger?
- What happens when you DROP a table with triggers?

Try unusual inputs, break the logic, and explore unexpected cases.

Let your curiosity lead — that's how real developers grow.

Reminder:

- **Errors are part of learning** – I expect to see them in your log or notes.
- Don't skip! Even if something didn't work, include it and explain why.
- I will **review every detail**, not just the final output.
- Don't just copy theory — show me your understanding through your testing!

➤ **References:**

Every team must include a **reference list** at the end of your README.

Use **reliable sources**:

e.g., Microsoft Docs, SQLServerTutorial, StackOverflow, YouTube tutorials, blog posts, GitHub samples, etc. (not only ChatGPT!)

➤ **Submission Deadline:**

Submit GitHub repo link by: **May 28, 2025 at 4:30 or before ... not after this time !**

➤ **Recognition:**

The **best work** will become our official explanation template. You'll be **credited as contributors** in the next batch's training resources.