**What’s the difference between full, differential and transactional back up?  
1. Full Backup**

* **What it does:** Backs up the **entire database** (data + part of transaction log).
* **Usage:** It’s the **base** backup required before any differential or log backups.
* **Recovery:** Can be used alone to restore the database **up to the time of the full backup**.
* **Size:** Largest among the three.

**Example Use Case:** Weekly full backups every Sunday.

**2. Differential Backup**

* **What it does:** Backs up **all changes since the last full backup** (not the last differential).
* **Usage:** Requires the **last full backup** to restore. Faster to create than full, but grows over time.
* **Recovery:** You need the **full backup + the latest differential backup**.
* **Size:** Medium (grows as more changes happen since the last full).

**Example Use Case:** Daily differential backups between weekly full backups.

**3. Transaction Log Backup**

* **What it does:** Backs up all **changes recorded in the transaction log** since the last log backup.
* **Usage:** Allows **point-in-time recovery**.
* **Recovery:** You need:
  + Last full backup
  + Optional differential
  + **All log backups** up to the point you want to restore
* **Size:** Small (usually), very frequent.

**Example Use Case:** Log backups every 15 minutes for high-availability systems.

| **Backup Type** | **Day** | **Notes** |
| --- | --- | --- |
| Full Backup | Sunday | Complete base backup |
| Differential | Monday | Changes since Sunday |
| Differential | Tuesday | Changes since Sunday |
| Log Backup | Every hour | Fine-grained changes tracking |

**What is permission and What’s the difference between grant and deny and**

**used on what level?**

Difference Between GRANT, DENY, and REVOKE

| **Command** | **What it does** | **Overrides** | **Purpose** |
| --- | --- | --- | --- |
| **GRANT** | Gives a user permission to perform an action | Nothing | Allow access |
| **DENY** | Explicitly blocks a user from performing an action | GRANT | Block access even if granted elsewhere |
| **REVOKE** | Removes a previous GRANT or DENY | N/A | Neutralize previous permission |

**What’s sql profiler and when using it?**

With SQL Profiler, you can:

* See which queries are being executed
* Monitor stored procedures, T-SQL commands, and errors
* Track login/logoff activity
* Measure query performance (duration, reads/writes, CPU time)
* Catch slow or expensive queries
* Detect deadlocks and blocking issues
* Analyze security issues or unauthorized access

**What is trigger and why use it and on what level and what makes it different from normal Stord procedure**

Triggers are used to:

* Enforce business rules automatically  
  (e.g., no negative salaries)
* Audit changes (who updated what, when)
* Log changes into history tables
* Prevent invalid operations
* Cascade actions (like automatically updating related rows)