1. Soft Deletes Required: E-commerce Platforms

In an e-commerce ecosystem, we almost never "hard delete" an order or a product that has been sold.

- **The Scenario:** A customer cancels an order or a merchant stops selling a specific item.
- **Why Soft Delete?**
    - **Data Integrity & Auditing:** If a customer bought a "Blue T-shirt" in 2023, you cannot hard delete that product record in 2024. If you did, the customer's order history would link to a non-existent ID, breaking the database's referential integrity.
    - **Analytics:** Business intelligence teams need to see "deleted" or "cancelled" orders to calculate churn rates and lost revenue.
    - **Accidental Deletion Recovery:** If an employee accidentally deletes a category of products, a soft delete (setting a deleted_at timestamp) allows for near-instant restoration.

2. Hard Deletes Preferred: GDPR/Privacy-First Systems

When dealing with Personal Identifiable Information (PII) or high-volume transient data, hard deletes are often the standard.

- **The Scenario:** A user requests "The Right to be Forgotten" under GDPR on a social media platform, or a system handles high-frequency session tokens.
- **Why Hard Delete?**
    - **Legal Compliance:** Regulations like GDPR and CCPA often require that when a user requests account deletion, their data must be physically removed from the storage medium within a certain timeframe. Keeping a "soft deleted" copy could result in heavy fines.
    - **Storage & Performance:** In systems that generate millions of temporary records (like session tokens, verification codes, or IoT sensor logs), keeping "soft deleted" rows would bloat the database indexes and significantly slow down query performance.
    - **Security:** If a database is compromised, soft-deleted data is still there for an attacker to steal. Hard deleting sensitive data minimizes the "blast radius" of a security breach.