Soft Delete Pg
You're viewing user-generated content that may be unverified or unsafe. Report

Edit with ChatGPT

Soft Delete in PostgreSQL

Slide 1: Title Slide

- **Title:** Soft Delete in PostgreSQL
- **Subtitle:** Managing Data Without Permanent Deletion
- **Your Name | Date**

Slide 2: What is Soft Delete?

- Soft delete means **marking a record as deleted** instead of actually removing it from the database.
- Commonly used in applications where **data recovery** is needed.
- Achieved by adding a `is_deleted` **(BOOLEAN) column**.
- Example:

```sql
ALTER TABLE users ADD COLUMN is_deleted BOOLEAN DEFAULT FALSE;
```

Slide 3: Hard Delete vs. Soft Delete

Feature	Hard Delete	Soft Delete
Data is removed?	☑ Yes	✘ No (Hidden)
Can be recovered?	✘ No	☑ Yes
Affects foreign keys?	☑ Yes (breaks relations)	✘ No
Suitable for logging?	✘ No	☑ Yes

Slide 4: Implementing Soft Delete in PostgreSQL

1. **Add an `is_deleted` column:**

```sql
ALTER TABLE users ADD COLUMN is_deleted BOOLEAN DEFAULT FALSE;
```

2. **Modify DELETE queries to update `is_deleted` instead:**

```sql
UPDATE users SET is_deleted = TRUE WHERE id = 2;
```

3. **Filter out deleted records in SELECT queries:**

```sql
SELECT * FROM users WHERE is_deleted = FALSE;
```

Slide 5: Soft Delete in Node.js (Express + PostgreSQL)

```javascript
app.delete("/users/:id", async (req, res) => {
  const userId = req.params.id;
  try {
    const result = await pool.query(
      "UPDATE users SET is_deleted = TRUE WHERE id = $1 RETURNING *",
      [userId]
    );
    res.json({ message: "User soft deleted", user: result.rows[0] });
  } catch (error) {
    res.status(500).json({ message: "Internal error" });
  }
});
```

Slide 6: Restoring Soft Deleted Data

- Since data is not **actually deleted**, we can restore it:

```sql
UPDATE users SET is_deleted = FALSE WHERE id = 2;
```

Slide 7: Using PostgreSQL Views for Soft Delete

- To make queries easier, create a **VIEW** that excludes deleted records:

```sql
CREATE VIEW active_users AS
SELECT * FROM users WHERE is_deleted = FALSE;
```

- Now, simply query the view:

```sql
SELECT * FROM active_users;
```

Slide 8: Advantages of Soft Delete

☑ Prevents accidental data loss
☑ Maintains foreign key relationships
☑ Allows for data restoration
☑ Useful for logging and audits

Slide 9: When to Use Soft Delete?

☑ User accounts (Deactivation instead of deletion)
☑ E-commerce (Restoring deleted products)
☑ Logging & Auditing systems