Top 10 Data Cleaning Steps

Remove Duplicates

Duplicate records can inflate metrics and distort analysis.

```sql
SELECT DISTINCT *
FROM table_name;
```

```sql
WITH CTE AS (
  SELECT *, ROW_NUMBER() OVER (PARTITION BY id_column ORDER BY created_at DESC) AS rn
  FROM table_name
)
DELETE FROM CTE WHERE rn > 1;
```

USE:

GROUP BY OR COUNT(*)

Handle NULL Values

NULLs can lead to incorrect aggregations or logic errors.

```sql
SELECT
  COALESCE(column_name, 'Default Value') AS clean_column
FROM table_name;
```

```sql
UPDATE table_name
SET column_name = 'Default'
WHERE column_name IS NULL;
```

Trim Whitespace

Extra spaces cause mismatches in joins and comparisons.

```sql
UPDATE table_name
SET column_name = TRIM(column_name);
```

USE

TRIM(), LTRIM() AND RTRIM() . .

Standardize Text Case

Inconsistent casing causes grouping and filtering issues.

```sql
UPDATE table_name
SET column_name = LOWER(column_name);
```

```sql
UPDATE table_name
SET column_name = UPPER(column_name);
```

STANDARDIZE BEFORE USING GROUP BY ON TEXT COLUMNS.

Fix Invalid Date Formats

Invalid date formats break time-based analysis.

Convert string to proper date

```sql
SELECT CAST(column_name AS DATE)
FROM table_name;
```

VALIDATE USING
ISDATE() IN SQL SERVER OR
USE REGEX IN ADVANCED DATABASES.

Validate Data Types

Text stored in numeric/date fields can cause failures in queries.

Cast to correct Data type

```sql
SELECT CAST(column_name AS INT)
FROM table_name;
```

Remove Outliers

Outliers distort averages, trends, and modeling.

Remove top / bottom 1% salaries

```sql
SELECT *
FROM table_name
WHERE salary BETWEEN
  PERCENTILE_CONT(0.01) WITHIN GROUP (ORDER BY salary)
  AND
  PERCENTILE_CONT(0.99) WITHIN GROUP (ORDER BY salary);
```

Fix Inconsistent Categories

‘NY’, ‘ny’, and ‘New York’ should all mean the same thing.

Create a mapping table for large datasets.

```sql
UPDATE table_name
SET city = 'New York'
WHERE LOWER(city) IN ('ny', 'n.y.', 'new york');
```

Thanks for Reading!

Follow for more content on SQL