1. Remove Duplicate Rows

```sql
Copy code
DELETE FROM table_name
WHERE id NOT IN (
    SELECT MIN(id)
    FROM table_name
    GROUP BY column1, column2, column3
);
```

---

2. Trim Whitespace from String Columns

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

---

3. Replace NULL Values with Default Values

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

---

4. Standardize Case in String Columns

Convert all text to uppercase:

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

Convert all text to lowercase:

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

---

5. Remove Non-Numeric Characters from a Column

If the column contains mixed characters but should only have numbers:

```sql
Copy code
UPDATE table_name
SET column_name = REGEXP_REPLACE(column_name, '[^0-9]', '');
```

(*Note: `REGEXP_REPLACE` syntax may vary depending on your database system.*)

---

6. Correct Common Typos

```sql
Copy code
UPDATE table_name
SET column_name = REPLACE(column_name, 'teh', 'the')
WHERE column_name LIKE '%teh%';
```

---

7. Identify and Remove Outliers

For example, if the `age` column should only contain values between 0 and 120:

```sql
Copy code
DELETE FROM table_name
WHERE age < 0 OR age > 120;
```

---

8. Fix Date Formatting Issues

Convert date strings to a standard format (e.g., `YYYY-MM-DD`):

```sql
Copy code
UPDATE table_name
SET date_column = TO_DATE(date_column, 'MM/DD/YYYY');
```

---

9. Find and Remove Rows with Non-ASCII Characters

```sql
Copy code
DELETE FROM table_name
WHERE column_name NOT LIKE '%[ -~]%';
```

(*Note: Adjust syntax for your database's pattern matching.*)

---

10. Combine and Normalize Data from Multiple Columns

Concatenate first name and last name:

```sql
Copy code
UPDATE table_name
SET full_name = CONCAT(first_name, ' ', last_name);
```

11. Remove Rows with Missing Key Data

```sql
Copy code
DELETE FROM table_name
WHERE key_column IS NULL OR key_column = '';
```

12. Normalize Phone Numbers

Format all phone numbers to `(XXX) XXX-XXXX`:

```sql
Copy code
UPDATE table_name
SET phone_number = REGEXP_REPLACE(phone_number, '(\d{3})(\d{3})(\d{4})',
'(\1) \2-\3');
```

13. Split a Column into Multiple Columns

Split a full name into first and last name:

```sql
Copy code
UPDATE table_name
SET first_name = SUBSTRING_INDEX(full_name, ' ', 1),
    last_name = SUBSTRING_INDEX(full_name, ' ', -1);
```

14. Remove Rows with Duplicate Column Pairs

```sql
Copy code
DELETE FROM table_name
WHERE (column1, column2) IN (
    SELECT column1, column2
    FROM table_name
    GROUP BY column1, column2
    HAVING COUNT(*) > 1
```

```
);
```

---

15. Standardize Date Formats Across Rows

Ensure all dates follow the `YYYY-MM-DD` format:

```sql
Copy code
UPDATE table_name
SET date_column = DATE_FORMAT(date_column, '%Y-%m-%d');
```

---

16. Fix Inconsistent Casing in a Column

Capitalize the first letter of each word:

```sql
Copy code
UPDATE table_name
SET column_name = INITCAP(column_name);
```

(*Note: `INITCAP` may vary or require equivalent functions depending on your database.*)

---

17. Identify Missing Data

Find rows with NULL or empty values:

```sql
Copy code
SELECT *
FROM table_name
WHERE column_name IS NULL OR column_name = '';
```

---

18. Replace Specific Values in a Column

```sql
Copy code
UPDATE table_name
SET column_name = 'Correct Value'
WHERE column_name = 'Incorrect Value';
```

---

19. Remove Special Characters from a Column

```sql
```

```
Copy code
UPDATE table_name
SET column_name = REGEXP_REPLACE(column_name, '[^a-zA-Z0-9 ]', '');
```

---

20. Flag Rows with Inconsistent Data

Add a flag for rows that don't meet criteria:

```sql
Copy code
UPDATE table_name
SET flag = 'Invalid'
WHERE column_name IS NULL OR column_name NOT LIKE 'ExpectedPattern%';
```

These SQL snippets can be adapted for most database systems, such as MySQL, PostgreSQL, Oracle, or SQL Server.