

Supabase Migration Instructions - Migration 013

Important Context

The `customer_stats` object in your database is currently a **TABLE**, not a **VIEW**. This migration will:

1. Drop the existing TABLE
2. Create a new VIEW with the same name
3. Add credit-related columns to the stats

Step-by-Step Instructions

Option 1: Copy & Paste (Recommended for Supabase)

1. Open the SQL file

- Open `/database/SUPABASE_MIGRATION_013.sql`
- Copy the entire contents (Ctrl+A, Ctrl+C)

2. Navigate to Supabase Dashboard

- Go to your Supabase project dashboard
- Click on **SQL Editor** in the left sidebar

3. Create a new query

- Click **“New Query”** button
- Paste the copied SQL

4. Run the migration

- Click **“Run”** button (or press Ctrl+Enter)
- Wait for the success message

5. Verify the migration

- Scroll down in the same SQL editor
- Run the verification queries (included in the file)
- Confirm that:
 - `customer_stats` is now a VIEW (not a TABLE)
 - New columns exist: `credit_balance` , `total_credits_purchased` , `total_credits_used`

Option 2: Using Node.js Script

```
cd /home/ubuntu/roblox-tool
node scripts/run-migration-013.js
```

Note: Make sure your `.env` file has the correct database credentials.

Option 3: Using psql Command Line

```
psql postgresql://postgres:[PASSWORD]@[HOST]:[PORT]/postgres \
-f database/migrations/013_add_credits_to_customer_stats.sql
```

✓ What Changed

Before (TABLE)

```
customer_stats TABLE with columns:
- id
- name
- is_active
- created_at
- contact_email
- max_users
- logo_url
- total_users
- active_users
- total_searches
- last_search_at
- last_login_at
```

After (VIEW)

```
customer_stats VIEW with columns:
- id
- name
- is_active
- created_at
- contact_email
- max_users
- logo_url
- total_users
- active_users
- total_searches
- last_search_at
- last_login_at
- credit_balance 🌟 NEW
- total_credits_purchased 🌟 NEW
- total_credits_used 🌟 NEW
```

🔍 Verification Queries

After running the migration, verify it worked:

```
-- Check that it's now a VIEW
SELECT table_type
FROM information_schema.tables
WHERE table_name = 'customer_stats';
-- Expected result: 'VIEW'

-- Test the new credit columns
SELECT
    name,
    credit_balance,
    total_credits_purchased,
    total_credits_used
FROM customer_stats;
```

Troubleshooting

Error: “customer_stats is not a view”

 **Solution:** This is expected! The migration handles this by using `DROP TABLE` instead of `DROP VIEW`.

Error: “permission denied”

 **Problem:** You need admin/owner permissions in Supabase

 **Solution:** Make sure you’re logged in as the project owner

Error: “customer_credits table does not exist”

 **Problem:** The credit system tables haven’t been created yet

 **Solution:** Run the earlier migrations first:

```
node scripts/run-migration-011.js # Creates customer_credits table  
node scripts/run-migration-012.js # Creates credit_transactions table
```

Warning: “DROP TABLE will affect X dependent objects”

 **Expected:** The `CASCADE` option will handle this safely



Expected Impact

- **Downtime:** None (views are created instantly)
- **Data Loss:** None (just restructuring, not deleting customer data)
- **Performance:** Improved (VIEWs are computed on-demand)
- **Dependencies:** Any queries using `customer_stats` will continue to work



Rollback (If Needed)

If you need to rollback:

```
-- Drop the VIEW
DROP VIEW IF EXISTS customer_stats;

-- Recreate the original TABLE (without credit columns)
CREATE TABLE customer_stats AS
SELECT
    c.id,
    c.name,
    c.is_active,
    c.created_at,
    c.contact_email,
    c.max_users,
    c.logo_url,
    COUNT(DISTINCT u.id) as total_users,
    COUNT(DISTINCT CASE WHEN u.is_active THEN u.id END) as active_users,
    COUNT(sh.id) as total_searches,
    MAX(sh.searched_at) as last_search_at,
    MAX(u.last_login) as last_login_at
FROM customers c
LEFT JOIN users u ON c.id = u.customer_id
LEFT JOIN search_history sh ON c.id = sh.customer_id
GROUP BY c.id, c.name, c.is_active, c.created_at, c.contact_email, c.max_users, c.logo_url;
```

Support

If you encounter issues:

1. Check the Supabase logs for detailed error messages
 2. Verify all previous migrations have been run
 3. Ensure the `customer_credits` table exists
 4. Confirm you have admin permissions
-

Last Updated: December 1, 2025

Migration Version: 013

Status: Ready for Production 