

Migration 015 Fix Guide

Problem

When trying to apply migration 015 to add credit balance to the `customer_stats` view, you may encounter this error:

```
ERROR: 42703: column c.email does not exist
LINE 9: c.email,
HINT: Perhaps you meant to reference the column "u.email".
```

Root Cause

The error occurs because the migration script incorrectly references `c.email`, but the `customers` table does not have an `email` column.

Database Schema Facts

CUSTOMERS table has:

- `id`
- `name`
- `is_active`
- `created_at`
- `updated_at`
- `contact_email` ← This is the correct column name
- `max_users`
- `logo_url`

USERS table has:

- `id`
- `username`
- `email` ← Email is in the USERS table, not CUSTOMERS
- `customer_id`
- `role`
- etc.

Solution

The Fix

Use `c.contact_email` instead of `c.email`

The correct column name in the `customers` table is `contact_email`, not `email`.

Files Updated

1. `SUPABASE_MIGRATION_015_FIXED.sql` - New comprehensive fixed version with:
 - Correct column names (`c.contact_email`)
 - Detailed comments explaining the schema
 - Validation queries to run before and after

- Troubleshooting section
 - Rollback instructions
2. `015_add_credits_to_customer_stats.sql` - Updated with:
- Better documentation
 - Clear comments on correct column usage
 - Inline notes highlighting the correct column names
3. `TEST_MIGRATION_015.sql` - New test file with:
- Pre-migration validation queries
 - Schema verification tests
 - Dry run of the view creation
 - Expected results documentation

How to Apply the Fix

Option 1: Using Supabase SQL Editor (Recommended for Supabase)

1. First, run validation tests:

```
sql
-- Check that customers table has contact_email (NOT email)
SELECT column_name
FROM information_schema.columns
WHERE table_schema = 'public'
AND table_name = 'customers'
AND column_name IN ('email', 'contact_email');
-- Expected: Should return 'contact_email'
```

2. Copy the entire content of `SUPABASE_MIGRATION_015_FIXED.sql`

3. Paste into Supabase SQL Editor

4. Run the migration

5. Verify with the provided validation queries

Option 2: Using Node.js Script (Local PostgreSQL)

```
cd /home/ubuntu/roblox-tool
npm run migrate:015
```

The script will:

- Read `015_add_credits_to_customer_stats.sql`
- Execute the migration
- Validate the results

Option 3: Manual PostgreSQL Command Line

```
psql -U postgres -d your_database -f database/migrations/
015_add_credits_to_customer_stats.sql
```

Verification Steps

After applying the migration, run these queries to verify:

1. Check View Columns

```
SELECT column_name, data_type
FROM information_schema.columns
WHERE table_schema = 'public'
    AND table_name = 'customer_stats'
ORDER BY ordinal_position;
```

Expected columns:

- id
- name
- is_active
- created_at
- contact_email (NOT 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)
- **last_purchase_at** (new)

2. Test the View

```
SELECT
    id,
    name,
    contact_email,
    credit_balance,
    total_credits_purchased,
    total_credits_used
FROM customer_stats
LIMIT 5;
```

Expected: Should return customer data with credit information (0 for customers without credits)

3. Verify Credit Data Accuracy

```

SELECT
    c.name,
    cs.credit_balance,
    cc.balance as actual_balance,
    CASE
        WHEN cs.credit_balance = COALESCE(cc.balance, 0) THEN '✅ Match'
        ELSE '❌ Mismatch'
    END as status
FROM customer_stats cs
JOIN customers c ON c.id = cs.id
LEFT JOIN customer_credits cc ON cc.customer_id = c.id
LIMIT 10;

```

Expected: All rows should show '✅ Match'

Common Errors and Solutions

Error: “column c.email does not exist”

Cause: Using `c.email` instead of `c.contact_email`

Solution: Make sure you’re using the FIXED version of the migration that uses `c.contact_email`

Error: “relation customer_credits does not exist”

Cause: The `customer_credits` table hasn’t been created yet

Solution: Run earlier migrations first:

```
npm run migrate:004 # Creates credit system tables
```

Error: “column cc.last_purchase_at does not exist”

Cause: Your `customer_credits` table is missing the `last_purchase_at` column

Solution: Check which version of migration 004 was run. The `004_create_credit_system_tables.sql` version includes `last_purchase_at`. You may need to alter the table:

```

ALTER TABLE customer_credits
ADD COLUMN IF NOT EXISTS last_purchase_at TIMESTAMP WITH TIME ZONE;

```

Credit balance shows 0 for all customers

Cause: No credit data has been created yet

Check:

```
SELECT * FROM customer_credits;
```

Solution: This is normal if no credits have been purchased. The view will correctly show 0 for customers without credit records.

Rollback

If you need to rollback to the previous version without credit fields:

```
DROP VIEW IF EXISTS customer_stats;

CREATE VIEW 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;
```

Important Notes

- Always use `c.contact_email` when referencing customer email from customers table
- Use `u.email` if you need individual user emails from users table
- The migration uses LEFT JOIN for customer_credits, so customers without credit records will show 0 for credit fields
- The view includes COALESCE to handle NULL values from LEFT JOIN
- Credit balance is calculated automatically by the customer_credits table's balance column

Testing Checklist

Before marking migration 015 as complete:

- [] Validation queries run without errors
- [] customer_stats view exists
- [] View has all expected columns including credit fields
- [] Sample query returns data without errors
- [] Credit balance matches actual balance in customer_credits table
- [] Customers without credit records show 0 for credit fields
- [] contact_email field is accessible (NOT email)
- [] No “column does not exist” errors in application logs

Files Reference

- **Fixed Migration (Supabase):** `database/migrations/SUPABASE_MIGRATION_015_FIXED.sql`

- **Updated Migration (PostgreSQL):** database/migrations/015_add_credits_to_customer_stats.sql
- **Test Queries:** database/migrations/TEST_MIGRATION_015.sql
- **Migration Runner Script:** scripts/run-migration-015.js
- **Application Guide:** database/migrations/APPLY_MIGRATION_015.md

Summary

The key fix is simple but critical:

✗ Wrong	✓ Correct
c.email	c.contact_email

The customers table has `contact_email`, not `email`. The FIXED migration files now use the correct column name and include extensive documentation to prevent this error in the future.