PostgreSQL Database Documentation

Database Information

- **Name**: migration_test
- **Version**: PostgreSQL 15 (adjust based on your installed version)
- **Host**: localhost
- **Port**: 5432
- **User**: test_user

Tables Overview

The database consists of 3 main tables forming a simple blog-like data structure:

Table Name	Description	Number of Records
users	Store user account information	3
user_profiles	Store extended user profile data	3
posts	Store blog posts created by users	4

Schema Details

Table: users

Primary table storing basic user information.

Columns:

Column Name	Data Type	Constraints	Description
id	SERIAL	PRIMARY KEY	Auto-incrementing user ID
name	VARCHAR(100)	NOT NULL	User's full name
email	VARCHAR(100)	NOT NULL, UNIQUE	User's email address
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Account creation timestamp

Indexes:

- Primary Key: id
- Unique Index: email

Sample Data:

id	name	email	created_at
1	John Doe	john@example.com	2023-05-14 10:00:00
2	Jane Smith	jane@example.com	2023-05-14 10:00:00
3	Bob Johnson	bob@example.com	2023-05-14 10:00:00

Table: user_profiles

Extended profile information for users.

Columns:

Column Name	Data Type	Constraints	Description
id	SERIAL	PRIMARY KEY	Auto-incrementing profile ID
user_id	INTEGER	FOREIGN KEY	Reference to users.id
bio	TEXT		User biography text
avatar	VARCHAR(255)		Path to user's avatar image

Indexes:

- Primary Key: id
- Foreign Key: user_id references users(id)

Sample Data:

id	user_id	bio	avatar
1	1	Software developer with 5 years experience	john.jpg
2	2	UX Designer and researcher	jane.jpg
3	3	DevOps engineer	bob.jpg

Table: posts

Blog posts created by users.

Columns:

Column Name	Data Type	Constraints	Description
id	SERIAL	PRIMARY KEY	Auto-incrementing post ID
user_id	INTEGER	FOREIGN KEY	Reference to users.id
title	VARCHAR(200)	NOT NULL	Post title
content	TEXT		Post content
published	BOOLEAN	DEFAULT false	Publication status
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Post creation timestamp

Indexes:

- Primary Key: id
- Foreign Key: user_id references users(id)

Sample Data:

id	user_id	title	content	published	created_at
1	1	PostgreSQL Basics	PostgreSQL is a powerful, open-source...	true	2023-05-14 10:00:00
2	1	Migration Strategies	When migrating data between databases...	true	2023-05-14 10:00:00
3	2	UX Design Principles	User experience is critical for application...	true	2023-05-14 10:00:00
4	3	DevOps Best Practices	Continuous integration and delivery pipelines...	false	2023-05-14 10:00:00

Relationships

The database has the following relationships:

1. **users ↔ user_profiles**: One-to-one relationship
   - A user can have one profile
   - A profile belongs to one user
   - Foreign key: `user_profiles.user_id` references `users.id`
2. **users ↔ posts**: One-to-many relationship
   - A user can have multiple posts
   - A post belongs to one user
   - Foreign key: `posts.user_id` references `users.id`

SQL to Recreate Schema

sql

```sql
-- Users table
CREATE TABLE users (
  id SERIAL PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  email VARCHAR(100) UNIQUE NOT NULL,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- User profiles table
CREATE TABLE user_profiles (
  id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(id),
  bio TEXT,
  avatar VARCHAR(255)
);

-- Posts table
CREATE TABLE posts (
  id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(id),
  title VARCHAR(200) NOT NULL,
  content TEXT,
  published BOOLEAN DEFAULT false,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Sample Data Insertion

sql

```sql
-- Insert sample users
INSERT INTO users (name, email) VALUES
  ('John Doe', 'john@example.com'),
  ('Jane Smith', 'jane@example.com'),
  ('Bob Johnson', 'bob@example.com');

-- Insert sample profiles
INSERT INTO user_profiles (user_id, bio, avatar) VALUES
  (1, 'Software developer with 5 years experience', 'john.jpg'),
  (2, 'UX Designer and researcher', 'jane.jpg'),
  (3, 'DevOps engineer', 'bob.jpg');

-- Insert sample posts
INSERT INTO posts (user_id, title, content, published) VALUES
  (1, 'PostgreSQL Basics', 'PostgreSQL is a powerful, open-source object-relational da
  (1, 'Migration Strategies', 'When migrating data between databases, there are severa
  (2, 'UX Design Principles', 'User experience is critical for application adoption...
  (3, 'DevOps Best Practices', 'Continuous integration and delivery pipelines...', fal
```