

# Improved PostgreSQL Schemas

## 1. users Table

```
CREATE TABLE users ( id SERIAL PRIMARY KEY, name VARCHAR(100) NOT NULL, email VARCHAR(100) UNIQUE NOT NULL, password_hash TEXT NOT NULL, role VARCHAR(10) NOT NULL DEFAULT 'user' CHECK (role IN ('user', 'admin')), created_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP, updated_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP );
```

- Added `DEFAULT 'user'` for role.
  - Used `TIMESTAMP WITH TIME ZONE` for better timezone handling.
  - Added `updated_at` column for tracking changes.
- 

## 2. products Table

```
CREATE TABLE products ( id SERIAL PRIMARY KEY, title VARCHAR(100) NOT NULL, description TEXT, price NUMERIC(10,2) NOT NULL CHECK (price >= 0), stock INT NOT NULL CHECK (stock >= 0), status VARCHAR(20) NOT NULL DEFAULT 'in-stock' CHECK (status IN ('in-stock', 'out-of-stock', 'sold')), created_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP, updated_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP );
```

- Added `CHECK` on `price` and `stock` to prevent negative values.
  - Added `DEFAULT 'in-stock'` for `status`.
  - Added `updated_at` for modification tracking.
- 

## 3. cart\_items Table

```
CREATE TABLE cart_items ( id SERIAL PRIMARY KEY, user_id INT NOT NULL REFERENCES users(id) ON DELETE CASCADE, product_id INT NOT NULL REFERENCES products(id) ON DELETE CASCADE, quantity INT NOT NULL CHECK (quantity > 0), created_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP, updated_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP, UNIQUE(user_id, product_id) );
```

- Made `user_id` and `product_id` NOT NULL to enforce presence.
  - Added timestamps for audit.
  - `UNIQUE(user_id, product_id)` to prevent duplicates.
- 

## 4. orders Table

```
CREATE TABLE orders ( id SERIAL PRIMARY KEY, user_id INT NOT NULL REFERENCES users(id)
ON DELETE CASCADE, total_price NUMERIC(10,2) NOT NULL CHECK (total_price >= 0), status
VARCHAR(20) NOT NULL DEFAULT 'pending' CHECK (status IN ('pending', 'in-transit',
'delivered', 'cancelled', 'returned')), created_at TIMESTAMP WITH TIME ZONE DEFAULT
CURRENT_TIMESTAMP, updated_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP );
```

- Added NOT NULL constraints.
  - Added DEFAULT 'pending' for status.
  - CHECK on total\_price to avoid negative values.
  - Added updated\_at for tracking order updates.
- 

## 5. order\_items Table

```
CREATE TABLE order_items ( id SERIAL PRIMARY KEY, order_id INT NOT NULL REFERENCES
orders(id) ON DELETE CASCADE, product_id INT NOT NULL REFERENCES products(id), quantity
INT NOT NULL CHECK (quantity > 0), price NUMERIC(10,2) NOT NULL CHECK (price >= 0) );
```

- Made foreign keys NOT NULL.
  - Added CHECK on quantity and price.
- 

## 6. returns Table

```
CREATE TABLE returns ( id SERIAL PRIMARY KEY, order_id INT NOT NULL REFERENCES
orders(id) ON DELETE CASCADE, user_id INT NOT NULL REFERENCES users(id), reason TEXT,
status VARCHAR(20) NOT NULL DEFAULT 'requested' CHECK (status IN ('requested',
'approved', 'rejected', 'processed')), created_at TIMESTAMP WITH TIME ZONE DEFAULT
CURRENT_TIMESTAMP, updated_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP );
```

- Made order\_id and user\_id NOT NULL.
- Added DEFAULT 'requested' for status.
- Added timestamps.