

```
-- DROP

-- order_items
DROP TABLE IF EXISTS order_items;
DROP SEQUENCE IF EXISTS order_items_seq;
DROP TYPE IF EXISTS ORDER_ITEM_STATUS;

-- orders
DROP TABLE IF EXISTS orders;
DROP SEQUENCE IF EXISTS orders_seq;
DROP TYPE IF EXISTS ORDER_STATUS;

-- addresses
DROP TABLE IF EXISTS addresses;
DROP SEQUENCE IF EXISTS addresses_seq;

-- customers
DROP TABLE IF EXISTS customers;
DROP SEQUENCE IF EXISTS customers_seq;
DROP TYPE IF EXISTS CUSTOMER_STATUS;

-- countries
DROP TABLE IF EXISTS countries;

-- part_images
DROP TABLE IF EXISTS part_images;
DROP SEQUENCE IF EXISTS part_images_seq;

-- item_parts
DROP TABLE IF EXISTS item_parts;
DROP SEQUENCE IF EXISTS item_parts_seq;

-- item_images
DROP TABLE IF EXISTS item_images;
DROP SEQUENCE IF EXISTS item_images_seq;

-- category_images
DROP TABLE IF EXISTS category_images;
DROP SEQUENCE IF EXISTS category_images_seq;

-- categories
DROP TABLE IF EXISTS categories;
DROP SEQUENCE IF EXISTS categories_seq;
DROP TYPE IF EXISTS CATEGORY_STATUS;

-- batches
DROP TABLE IF EXISTS batches;
DROP SEQUENCE IF EXISTS batches_seq;

-- parts
DROP TABLE IF EXISTS parts;
DROP SEQUENCE IF EXISTS parts_seq;

-- images
DROP TABLE IF EXISTS images;
DROP SEQUENCE IF EXISTS images_seq;
DROP TYPE IF EXISTS MIMETYPE;
```

```

-- items
DROP TABLE IF EXISTS items;
DROP SEQUENCE IF EXISTS items_seq;

-- users
DROP TABLE IF EXISTS users;
DROP SEQUENCE IF EXISTS users_seq;

-- roles
DROP TABLE IF EXISTS roles;
DROP SEQUENCE IF EXISTS roles_seq;

-- texts
DROP TABLE IF EXISTS texts;
DROP SEQUENCE IF EXISTS texts_seq;

-- CREATE

-- texts
CREATE SEQUENCE texts_seq;
CREATE TABLE texts (
    id integer NOT NULL PRIMARY KEY DEFAULT NEXTVAL ('texts_seq'),
    table_name varchar(20) NOT NULL,
    code varchar(20) NOT NULL,
    en varchar(20) NULL,
    fr varchar(20) NULL,
    de varchar(20) NULL
);

-- roles
CREATE SEQUENCE roles_seq;
CREATE TABLE roles (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('roles_seq'),
    name VARCHAR(20) NOT NULL,
    description VARCHAR(60) NOT NULL
);

-- users
CREATE SEQUENCE users_seq;
CREATE TABLE users (
    id integer NOT NULL PRIMARY KEY DEFAULT NEXTVAL('users_seq'),
    email VARCHAR(60) NOT NULL UNIQUE,
    password VARCHAR(60) NOT NULL,
    salt VARCHAR(60) NOT NULL,
    username VARCHAR(20) NOT NULL,
    firstname VARCHAR(20) NOT NULL,
    middlename VARCHAR(20) NULL,
    lastname VARCHAR(20) NOT NULL,
    enabled BOOLEAN DEFAULT FALSE
);

```

```

-- items
CREATE SEQUENCE items_seq;
CREATE TABLE items (
    id INTEGER PRIMARY KEY DEFAULT NEXTVAL('items_seq'),
    parent INTEGER NULL,
    abstract BOOLEAN NOT NULL DEFAULT FALSE,
    customisation BOOLEAN NOT NULL DEFAULT FALSE,
    title VARCHAR(20) NOT NULL,
    blurb VARCHAR(100) NULL,
    description VARCHAR(4000) NULL,
    price NUMERIC(8, 2) NOT NULL DEFAULT 0,
    weight FLOAT NOT NULL DEFAULT 0,
    in_stock INTEGER NOT NULL DEFAULT 0,
    on_backorder INTEGER NOT NULL DEFAULT 0,
    can_backorder BOOLEAN NOT NULL DEFAULT FALSE
);

-- images
CREATE TYPE MIMETYPE AS ENUM ('jpg', 'png', 'gif', 'mp4');
CREATE SEQUENCE images_seq;
CREATE TABLE images (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT nextval('images_seq'),
    url VARCHAR(60) NOT NULL,
    name VARCHAR(20) NOT NULL,
    ordering INTEGER NOT NULL DEFAULT 0,
    mimetype MIMETYPE NOT NULL
);

-- parts
CREATE SEQUENCE parts_seq;
CREATE TABLE parts (
    id INTEGER PRIMARY KEY DEFAULT NEXTVAL('parts_seq'),
    title VARCHAR(20) NOT NULL,
    blurb VARCHAR(100) NULL,
    description VARCHAR(4000) NULL,
    price NUMERIC(8, 2) NOT NULL DEFAULT 0,
    weight FLOAT NOT NULL DEFAULT 0,
    in_stock INTEGER NOT NULL DEFAULT 0
);

-- batches
CREATE SEQUENCE batches_seq;
CREATE TABLE batches (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('batches_seq'),
    user_id INTEGER NOT NULL REFERENCES users(id),
    item_id INTEGER NOT NULL REFERENCES items(id),
    initial_size INTEGER NOT NULL DEFAULT 0,
    finished_items INTEGER NOT NULL DEFAULT 0,
    rejected_items INTEGER NOT NULL DEFAULT 0,
    created TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
);

```

```

-- categories
CREATE TYPE CATEGORY_STATUS AS ENUM ('good', 'bad');
CREATE SEQUENCE categories_seq;
CREATE TABLE categories (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('categories_seq'),
    parent INTEGER NULL,
    title VARCHAR(20) NOT NULL,
    blurb VARCHAR(20) NULL,
    description VARCHAR(4000) NULL,
    user_id INTEGER NOT NULL REFERENCES users(id),
    status CATEGORY_STATUS NOT NULL,
    created_on TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
);

-- category_images
CREATE SEQUENCE category_images_seq;
CREATE TABLE category_images (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('category_images_seq'),
    category_id INTEGER NOT NULL REFERENCES categories(id),
    image_id INTEGER NOT NULL REFERENCES images(id),
    quantity INTEGER NOT NULL DEFAULT 0,
    ordering INTEGER NOT NULL DEFAULT 0
);

-- item_images
CREATE SEQUENCE item_images_seq;
CREATE TABLE item_images (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('item_images_seq'),
    item_id INTEGER NOT NULL REFERENCES items(id),
    image_id INTEGER NOT NULL REFERENCES images(id),
    quantity INTEGER NOT NULL DEFAULT 0,
    ordering INTEGER NOT NULL DEFAULT 0
);

-- item_parts
CREATE SEQUENCE item_parts_seq;
CREATE TABLE item_parts (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('item_parts_seq'),
    item_id INTEGER NOT NULL REFERENCES items(id),
    part_id INTEGER NOT NULL REFERENCES parts(id),
    quantity INTEGER NOT NULL DEFAULT 0,
    ordering INTEGER NOT NULL DEFAULT 0
);

-- part_images
CREATE SEQUENCE part_images_seq;
CREATE TABLE part_images (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('part_images_seq'),
    part_id INTEGER NOT NULL REFERENCES parts(id),
    image_id INTEGER NOT NULL REFERENCES images(id),
    quantity INTEGER NOT NULL DEFAULT 0,
    ordering INTEGER NOT NULL DEFAULT 0
);

```

```

-- countries
CREATE TABLE countries (
    code CHAR(2) NOT NULL PRIMARY KEY,
    en VARCHAR(20) NOT NULL,
    fr VARCHAR(20) NOT NULL,
    de VARCHAR(20) NOT NULL
);

-- customers
CREATE TYPE CUSTOMER_STATUS AS ENUM ('good', 'bad');
CREATE SEQUENCE customers_seq;
CREATE TABLE customers (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('customers_seq'),
    email VARCHAR(60) NOT NULL,
    password VARCHAR(60) NOT NULL,
    salt VARCHAR(60) NOT NULL,
    firstname VARCHAR(20) NOT NULL,
    middlename VARCHAR(20) NULL,
    lastname VARCHAR(20) NOT NULL,
    created TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
    modified TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
    status CUSTOMER_STATUS NOT NULL DEFAULT 'good'
);

-- addresses
CREATE SEQUENCE addresses_seq;
CREATE TABLE addresses (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('addresses_seq'),
    customer INTEGER NOT NULL REFERENCES customers(id),
    current BOOLEAN NOT NULL DEFAULT FALSE,
    billing BOOLEAN NOT NULL DEFAULT FALSE,
    shipping BOOLEAN NOT NULL DEFAULT FALSE,
    address1 VARCHAR(60) NOT NULL,
    address2 VARCHAR(60) NULL,
    city VARCHAR(20) NOT NULL,
    state VARCHAR(20) NOT NULL,
    zip VARCHAR(20) NOT NULL,
    country_id CHAR(2) NOT NULL REFERENCES countries(code),
    ordering INTEGER NOT NULL DEFAULT 0
);

```

```
-- orders
CREATE TYPE ORDER_STATUS AS ENUM('good', 'bad');
CREATE SEQUENCE orders_seq;
CREATE TABLE orders (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('orders_seq'),
    customer_id INTEGER NOT NULL REFERENCES customers(id),
    shipping_address INTEGER NOT NULL REFERENCES addresses(id),
    billing_address INTEGER NOT NULL REFERENCES addresses(id),
    created TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
    status ORDER_STATUS NOT NULL DEFAULT 'good'
);

-- order_items
CREATE TYPE ORDER_ITEM_STATUS AS ENUM('good', 'bad');
CREATE SEQUENCE order_items_seq;
CREATE TABLE order_items (
    id INTEGER NOT NULL PRIMARY KEY DEFAULT NEXTVAL('order_items_seq'),
    order_id INTEGER NOT NULL REFERENCES orders(id),
    item_id INTEGER NOT NULL REFERENCES items(id),
    quantity INTEGER NOT NULL DEFAULT 0,
    ordering INTEGER NOT NULL DEFAULT 0,
    status ORDER_ITEM_STATUS NOT NULL DEFAULT 'good'
);
```