Booktique

Booktique is a system that implements an online book store.

Repository

The source code is available on GitLab at
https://gitlab.com/GiorgiaAuroraAdorni/3rdAssignment/.

Contributors

This project has been developed by Giorgia Adorni (806787) .

Installation

```
$ git clone https://gitlab.com/GiorgiaAuroraAdorni/3rdAssignment.git
$ cd booktique
$ docker-compose up
```

An alternative to `docker-compose up` is the command `docker-compose up --build` that
allows the rebuild of the app.
The application will be available at http://localhost:8080/
In addition, the `docker-compose.yml` file also starts an instance of pgAdmin to access the
database directly: you can reach it at http://localhost:5432/.

Docker Compose also allows to locally run the unit tests:

```
$ docker-compose run app mvn test
```

Application

Booktique is a system that implements an online book store.
The basic interface involves querying books according to the fields ISBN, title, author, publisher,
language etc. We support services for buying books. We build a personal profile page which is
used for handling customer orders. We implement a "recommendation system" for
recommending related books (prequels and sequels) based on those searched and purchased.

Entities

The model is composed of 9 entities:

- **book** - items available in the online store. The book object provides a set of information
  such as title, ISBN, authors, publisher, format etc. A unique id is used as the primary key,
  despite this the ISBN provides a perfect primary key because every book has a different
  ISBN.

- **person** - a real person. The person object provides personal information such as name, surname, fiscal code, date of birth, email and telephone number. A unique id is used as a primary key, in addition the fiscal code is marked as a natural identifier as it identifies a person in the real world.
- **author** - the author of the book. Because an author is also a person, the object inherits the same attributes of the person superclass. More information about the author is provided, for example an optional biography and the web site URL.
- **customer** – people who buy books in the online store. Every customer has a basic set of data inherits from the person superclass such as for the authors. Mandatory attributes, such as login credentials and postal address, is added to customers, but can be also provided VAT number.
- **employee** – is a person who carries out activities related to orders. An employee has the same properties of its parent class, in addition he is associated whit another employee as a supervisor. More information about the employee is given, such as the login credentials, the hire date and his postal address.
- **supplier** - organization that provides book to the online store. To the supplier object is associated a unique id as primary key, the company name and other information such as email, telephone number and postal address.
- **item** – articles selected by the customer for purchase in online store. To each item is assigned the reference to the book in the catalog that the customer want to buy, its unit price, the desired quantity and its supplier.
- **purchase** - items ordered. To every purchase is associated a hypothetically unlimited number of items, the customer, the employee who takes charge of the order, order and shipping dates, the total amount, order status and the transaction information (payment type and date).
- **address** - postal information about customer and supplier. In this object is stored the street address, the postal code, the city, region and country name.

Relationships

Between those entities ther's 14 relations, in particular:

- two self-relations: the first one is a One-to-One relation between a book and his prequel, inand in the other between the book and his sequel; the second is a Many-to-One relation between employees and supervisor (each employee is associated with only one supervisor).
- one relation Many-to-Many between books and authors, handled with lazy load.
- one inheritance hierarchy that involves 4 entities: the class person serves as a superclass for employee, author and customer subclass.

persons
🔑 id	BIGINT
created_at	TIMESTAMP(6) WITHOUT TIME ZONE
updated_at	TIMESTAMP(6) WITHOUT TIME ZONE
birth_date	DATE
email	CHARACTER VARYING(255)
fiscal_code	CHARACTER VARYING(255)
mobile_phone	CHARACTER VARYING(255)
name	CHARACTER VARYING(50)
surname	CHARACTER VARYING(50)

authors
author	BYTEA
biography	CHARACTER VARYING(255)
web_siteurl	CHARACTER VARYING(255)
id	BIGINT

customers
customer	BYTEA
password	CHARACTER VARYING(255)
username	CHARACTER VARYING(255)
vat_number	CHARACTER VARYING(255)
id	BIGINT
address_id	BIGINT

addresses
🔑 id	BIGINT
created_at	TIMESTAMP(6) WITHOUT TIME ZONE
updated_at	TIMESTAMP(6) WITHOUT TIME ZONE
address	CHARACTER VARYING(120)
building	CHARACTER VARYING(120)
city	CHARACTER VARYING(80)
country	CHARACTER VARYING(30)
postal_code	CHARACTER VARYING(10)
region	CHARACTER VARYING(30)

suppliers
🔑 id	BIGINT
created_at	TIMESTAMP(6) WITHOUT TIME ZONE
updated_at	TIMESTAMP(6) WITHOUT TIME ZONE
company_name	CHARACTER VARYING(50)
email	CHARACTER VARYING(255)
mobile_phone	CHARACTER VARYING(255)
address_id	BIGINT

purchases
🔑 id	BIGINT
created_at	TIMESTAMP(6) WITHOUT TIME ZONE
updated_at	TIMESTAMP(6) WITHOUT TIME ZONE
amount	NUMERIC(12,2)
order_date	DATE
payment_date	DATE
payment_type	CHARACTER VARYING(255)
shipping_date	DATE
status	CHARACTER VARYING(255)
customer_id	BIGINT
employee_id	BIGINT

employees
employee	BYTEA
hire_date	DATE
password	CHARACTER VARYING(255)
username	CHARACTER VARYING(255)
id	BIGINT
address_id	BIGINT
supervisor_id	BIGINT

books_authors
🔑 **book_id**	BIGINT
🔑 **authors_id**	BIGINT

books
🔑 id	BIGINT
created_at	TIMESTAMP(6) WITHOUT TIME ZONE
updated_at	TIMESTAMP(6) WITHOUT TIME ZONE
book_format	CHARACTER VARYING(255)
edition	INTEGER
isbn	CHARACTER VARYING(255)
language	CHARACTER VARYING(30)
publication_date	DATE
publisher	CHARACTER VARYING(100)
subtitle	CHARACTER VARYING(100)
title	CHARACTER VARYING(100)
prequel_id	BIGINT

purchases_items
🔑 **purchase_id**	BIGINT
🔑 **items_id**	BIGINT

items
🔑 id	BIGINT
created_at	TIMESTAMP(6) WITHOUT TIME ZONE
updated_at	TIMESTAMP(6) WITHOUT TIME ZONE
quantity_per_unit	INTEGER
unit_price	NUMERIC(12,2)
book_item_id	BIGINT
supplier_id	BIGINT