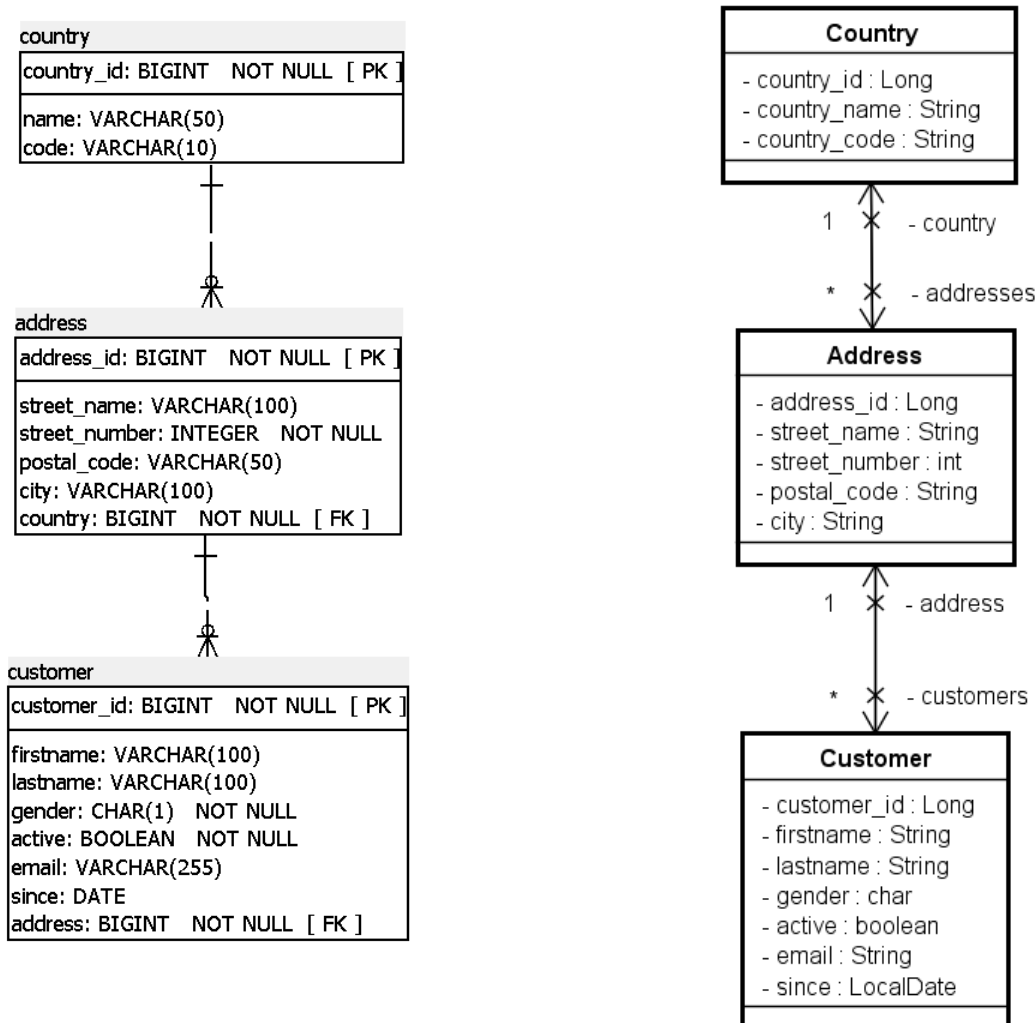


Example 103 JPA CustomerDB

Create the database `customerdb` according to the given ER- and Class-diagram with a JPA mapping.



Implement a console application, that reads testdata from the files `customers.xml` resp. `customer.json`, creates the Pojos and persists them to the database. After successful import the number of imported datasets is printed to the console:

```
Countries imported: 6
Addresses imported: 1000
Customers imported: 1000
```

Mapping constraints for the Pojo-classes:

beans.Country.java

- ❑ JPA-mapping according to the given diagrams
- ❑ The primary-key is created by a sequence
- ❑ The associations to class **Address** is bidirectional. Use a **List**-Collection
- ❑ Create the following **NamedQueries**
 - **findByName** to get a country specified by name
 - **findAll** to get all countries
 - **countAll** to get the number of all different countries

beans.Address.java

- ❏ JPA-mapping according to the given diagrams
- ❏ The primary-key is generated in the database
- ❏ The associations to the classes *Country* and *Customer* are bidirectional. Use *List-Collections*
- ❏ Create the **NamedQuery** `countAll` to get the number of different addresses

beans.Customer.java

- ❏ JPA-mapping according to the given diagrams
- ❏ The primary-key is generated in the database
- ❏ The association to class **Address** is bidirectional.
- ❏ Create the following **NamedQueries**
 - **findYears** to get all different years from property **since**
 - **countAll** to get the number of all customers
 - **findFromCountry** to get all active or inactive customers of a specified country

console.DataImport.java

- ❏ **open()** - Open the connection to database **customerdb**
- ❏ **close()** - Close the connection to the database
- ❏ **importXML()** - Read data from file **customers.xml**, create the Pojos and persist them to the database **customerdb**. No duplicate countries are allowed. Make sure that all bidirectional associations are setup!
- ❏ **importJSON()** - Read data from file **customers.xml**, create the Pojos and persist them to the database **customerdb**. No duplicate countries are allowed in the database. Make sure that all bidirectional associations are setup!
- ❏ **main()** - Menu-driven console application: implement menu-items for the XML- and the JSON-import and menu-items to test all queries.