

Tema 1

LibroData – Gestiunea și livrarea de cărți electronice

Responsabil: Laur Neagu

Introducere

Scopul acestei teme este de a vă familiariza cu programarea orientată pe obiecte și cu limbajul Java. În același timp, puteți aprofunda cunoștințele cu privire la structuri de date și aveți șansa de a implementa un prim proiect de dezvoltare software similar industriei.

Descrierea problemei

Odată cu digitalizarea a mare parte din industrie, și industria distribuției cărților a fost modificată. În acest context, prezentăm mai departe cum funcționează entitățile ce modelează procesul de distribuție a cărților electronice.

Entitatea **Book** este elementul de bază al domeniului. Ea este caracterizată de următoarele:

- ID (identificator unic)
- name (numele de publicare al cărții)
- subtitle (subtitlu al cărții, opțional)
- ISBN (identificat unic al cărții, format string, e.g. 9782806231789, cunoscut de entități externe)
- pageCount (număr de pagini)
- keywords (cuvinte cheie ale cărții, separate prin ;)
- languageID (limba în care este scrisă)
- createdOn (data adăugării în sistem a cărții)
- authors (listă de autori)

Mai departe, definim Entitatea **Language**. Ea are următoarele proprietăți:

- ID (identificator unic)
- code (cod de limbă, prescurtare a limbii)
- name

Entitatea **Author**, cu următoarele caracteristici:

- ID (identificator unic)
- firstName
- lastName

Entitatea **PublishingBrand** – orice carte poate sau nu să treacă în publicare printr-un brand care se ocupă de publicare – cu următoarele proprietăți:

- ID (identificator unic)
- name
- books (listă de cărți publicate de un brand în parte)

Entitatea **EditorialGroup** – orice carte care poate sau nu să treacă înainte de publicare printr-un brand care se ocupa de editarea cărții – cu următoarele proprietăți:

- ID (identificator unic)
- name
- books (listă de cărți publicate de un brand în parte)

Mai departe prezentăm entitatea **PublishingRetailer**, care este efectiv firma care publică cărțile în magazinele online, spre exemplu – Google, Amazon, Apple, etc. Ei sunt cei care afișează cărțile pe platformele lor, și au următoarele proprietăți:

- ID (identificator unic)
- name
- publishingArtifacts (listă de entități publicabile – tip IPublishingArtifact)
- countries (listă cu țările în care respectivul retailer publică)

IPublishingArtifact este o interfață ce este implementată de către clasele **Book**, **PublishingBrand**, **EditorialGroup**.

Logica este următoarea – către **PublishingRetailer** pot fi publicate cărți direct de la autori, prin entitatea **Book**, de la **EditorialGroup** – care implică publicarea către retailer a tuturor cărților ce sunt editate la un editorial group; respectiv de la **PublishingBrand** – cărțile dintr-un brand de publicare merg automat și la retailerii unde acel PublishingBrand aparține.

Interfața **IPublishingArtifact** conține o metodă Publish() ce întoarce un String, care conține metadata respectivului artifact.

Mențiune – o carte poate fi publicată unui **PublishingRetailer** prin oricare din cele 3 canale – direct de autor, prin editorial group, respectiv prin publishing brand.

Pentru o instanță **Book1**, ea va trebui să întoarcă următoarele:

```
<xml>
  <title>NameBook1</title>
  <subtitle>SubtitleBook1</subtitle>
  <isbn>isbnBook1</isbn>
  <pageCount>pageCountBook1</pageCount>
  <keywords>keywordsBook1</keywords>
  <languageID>languageIDBook1</languageID>
  <createdOn>createdOnBook1<createdOn>
  <authors>authorsBook1<authors>
</xml>
```

Pentru o instanță *PublishingBrand1*, ea va trebui să întoarcă următoarele:

```
<xml>
  <publishingBrand>
    <ID>publishingBrandIDPublishingBrand1</ID>
    <Name>publishingBrandNamePublishingBrand1</Name>
  </publishingBrand>
  <books>
    <book>
      <title>NameBook1PublishingBrand1</title>
      <subtitle>SubtitleBook1PublishingBrand1</subtitle>
      <isbn>isbnBook1PublishingBrand1</isbn>
      <pageCount>pageCountBook1PublishingBrand1</pageCount>
      <keywords>keywordsBook1PublishingBrand1</keywords>
      <languageID>languageIDBook1PublishingBrand1</languageID>
      <createdOn>createdOnBook1PublishingBrand1</createdOn>
      <authors>authorsBook1PublishingBrand1</authors>
    </book>
    <!-- Alte cărți pentru acest publishing brand -->
  </books>
</xml>
```

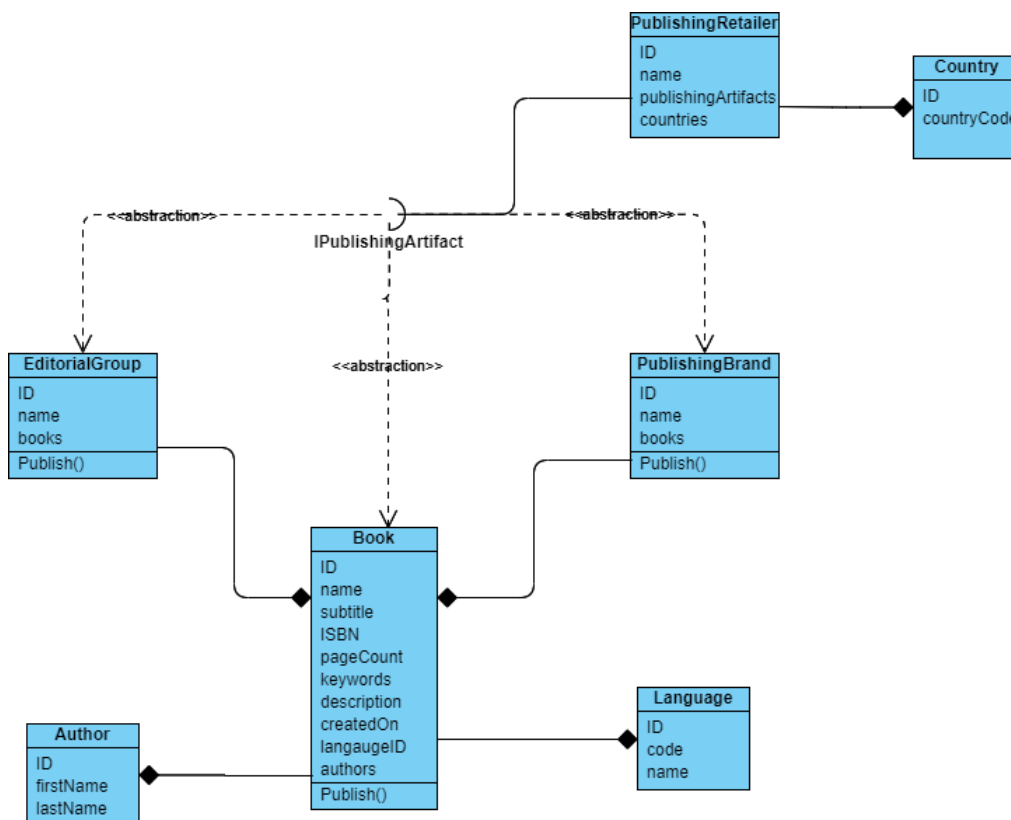
Pentru o instanță *EditorialGroup1*, metoda Publish() va trebui să întoarcă următoarele:

```
<xml>
  <editorialGroup>
    <ID>editorialGroupIDEditorialGroup1</ID>
    <Name>editorialGroupName EditorialGroup1</Name>
  </editorialGroup>
  <books>
    <book>
      <title>NameBook1EditorialGroup1</title>
      <subtitle>SubtitleBook1EditorialGroup1</subtitle>
      <isbn>isbnBook1EditorialGroup1</isbn>
      <pageCount>pageCountBook1EditorialGroup1</pageCount>
      <keywords>keywordsBook1EditorialGroup1</keywords>
      <languageID>languageIDBook1EditorialGroup1</languageID>
      <createdOn>createdOnBook1EditorialGroup1</createdOn>
      <authors>authorsBook1EditorialGroup1</authors>
    </book>
    <!-- Alte cărți pentru acest editorial group -->
  </books>
</xml>
```

Entitatea **Countries**, care are următoarele proprietăți:

- ID (identificator unic)
- countryCode (cod pentru țară, din două litere – e.g., RO pentru România)

O vedere generală a entităților din Librodata este disponibilă mai jos, în Figură 1. Diagrama claselor.



Figură 1. Diagrama claselor

În contextul acestei teme, vi s-au pregătit câteva seturi de date pe baza cărora să inițializăm biblioteca virtuală de cărți. Acestea se vor importa printr-un set de fișiere de inițializare – fișiere de intrare.

Primiți așadar următoarele fișiere, separatorul este “###”:

- *books.in* – inițializarea cărților în sistem, header-ul fișierului sunt proprietățile clasei **Book**, fără proprietatea **authors**
- *languages.in* – inițializarea limbilor în sistem, header-ul fișierului sunt proprietățile clasei **Language**
- *authors.in* – inițializarea autorilor în sistem, header-ul fișierului sunt proprietățile clasei **Authors**
- *books-authors.in* – inițializarea asocierilor dintre cărți și autori, headerul fișierului sunt id-urile cărții și , respectiv al autorului
- *editorial-groups.in* – inițializarea grupurilor de editare din sistem, header-ul fișierului sunt proprietățile clasei **EditorialGroup**

- *publishing-brands.in* - inițializarea grupurilor de publicare din sistem, header-ul fișierului sunt proprietățile clasei **PublishingBrand**
- *editorial-groups-books.in* – inițializarea asocierilor dintre grupuri de editare și autori, headerul fișierului sunt id-urile grupului de editare și , respectiv al autorului
- *publishing-brands-books.in* – inițializarea asocierilor dintre grupuri de publicare și autori, headerul fișierului sunt id-urile grupului de publicare și , respectiv al autorului
- *publishing-retailers.in* – inițializarea retailerilor din sistem, header-ul fișierului sunt proprietățile clasei **PublishingRetailer**
- *countries.in* – inițializarea țărilor din sistem, header-ul fișierului sunt proprietățile clasei **Country**
- *publishing-retailers-countries.in* – inițializarea asocierilor dintre retailer și țări, headerul fișierului sunt id-urile retailerului și, respectiv, al țării
- *publishing-retailers-books.in* - inițializarea asocierilor dintre retailer și cărți, headerul fișierului sunt id-urile retailerului și, respectiv, al cărții
- *publishing-retailers-editorial-groups.in* - inițializarea asocierilor dintre retailer și grupuri de editare, headerul fișierului sunt id-urile retailerului și, respectiv, al grupului de editare
- *publishing-retailers-publishing-brands.in* - inițializarea asocierilor dintre retailer și grupuri de publicare, headerul fișierului sunt id-urile retailerului și, respectiv, al grupului de publicare

Ca dezvoltator software al acestui sistem, vi se cere implementarea următoarelor funcționalități:

- *getBooksForPublishingRetailerID(int publishingRetailerID)* - Cărțile publicate de către un retailer – metodă care întoarce o listă de cărți și primește ca parametru un publishing retailer ID – va trebui să verifice atât entitatea **Book**, cât și **EditorialGroup** și **PublishingBrand** și să afișeze cărțile unice
- *getLanguagesForPublishingRetailerID(int publishingRetailerID)* – Limbile în care sunt publicate cărțile unui retailer – metodă care întoarce o listă de limbi și primește ca parametru un publishing retailer ID
- *getCountriesForBookID(int bookID)* – Țările în care o carte a ajuns – metodă care întoarce o listă de țări și primește ca parametru un book ID
- *getCommonBooksForRetailerIDs(int retailerID1, int retailerID2)* – Cărți comune între retaileri – metodă care întoarce o listă de cărți comune între retaileri și primește ca parametru două ID-uri de retaileri
- *getAllBooksForRetailerIDs (int retailerID1, int retailerID2)* - Cărți ale retailerilor (union) – metodă care întoarce o listă de cărți între retaileri (uniunea celor două liste) și primește ca parametru două ID-uri de retaileri

Recomandare – definiți aceste metode într-o clasă separată, pe care o puteți numi Administration.

Pentru citirea din fișier, regăsiți mai jos un snippet de cod prin care puteți face asta:

```
try (BufferedReader br = new BufferedReader(new FileReader(file))) {
    String line;
    while ((line = br.readLine()) != null) {
        // process the line.
    }
}
```

PUNCTAJ:

DOMENIU	TASK	PUNCTAJ	TOTAL
OOP – PART 1	Definire clasă <i>Book</i> Definire clasă <i>Language</i> Definire clasă <i>Author</i> Definire clasă <i>EditorialGroup</i> Definire clasă <i>PublishingBrand</i> Definire clasă <i>PublishingRetailer</i> Definire clasă <i>Country</i> Definire clasă <i>IPublishingArtifact</i>	3 p 1 p 1 p 1 p 1 p 1 p 1 p 1 p	10 PUNCTE
OOP – PART 2	Implementare Publish() <i>Book</i> Implementare Publish() <i>EditorialGroup</i> Implementare Publish() <i>PublishingBrand</i>	5 p 5 p 5 p	15 PUNCTE
Java - BASIC	Initializare <i>books.in</i> Initializare <i>languages.in</i> Initializare <i>authors.in</i> Initializare <i>books-authors.in</i> Initializare <i>editorial-groups.in</i> Initializare <i>publishing-brands.in</i> Initializare <i>editorial-groups-books.in</i> Initializare <i>publishing-brands-books.in</i> Initializare <i>publishing-retailers.in</i> Initializare <i>countries.in</i> Initializare <i>publishing-retailers-countries.in</i> Initializare <i>publishing-retailers-books.in</i> Initializare <i>publishing-retailers-editorial-groups.in</i> Initializare <i>publishing-retailers-publishing-brands.in</i>	2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p 2.5 p	35 PUNCTE
Java – Data Structures	<i>getBooksForPublishingRetailerID()</i> <i>getLanguagesForPublishingRetailerID()</i> <i>getCountriesForBookID()</i> <i>getCommonBooksForRetailerIDs()</i> <i>getAllBooksForRetailerIDs()</i> Testare fiecare metodă cu câte 5 exemple (5 * 5 = 25 apeluri)	7 p 7 p 7 p 7 p 7 p 5 p	40 PUNCTE

BONUS:

Implementare optimă a metodelor *getCommonBooksForRetailerIDs()* și *getAllBooksForRetailerIDs()* – complexitate $O(n+m)$ – 10 pct