

Projet de Concepts Objet

2019-2020

Présentation

L’initiative *open data* (**données ouvertes**) vise à obtenir, d’entités publiques comme privées (**services publics, collectivités, entreprises, ...**), le libre accès et usage aux données numériques qu’elles produisent. Celles-ci doivent être diffusées de manière (semi-)structurée sous une licence libre garantissant l’accès, le partage et la modification sans restriction technique, juridique ou financière¹. Ces données peuvent alors être valorisées pour l’ensemble de la collectivité (travaux de recherche, développement de services, informations pour les citoyens, sources pour les journalistes, ...).

Dans ce projet, nous nous intéresserons à un jeu de données, mis à disposition par la mairie de Paris, qui compile les collections de l’ensemble des bibliothèques de la ville. Ces données devront être structurées de manière à mettre en place un système efficace de recherche, d’emprunts et d’échanges dans les collections de chaque bibliothèque.

Jeu de données

Les données sont hébergées sur le site <https://opendata.paris.fr>. On considérera un sous-ensemble exporté sous la forme d’un tableur au format **CSV**.

Format CSV

Un fichier **CSV** (*Comma-Separated Values*) est un format de texte permettant de représenter un tableur. Les colonnes sont délimitées entre elles par un caractère donné (par exemple le **séparateur** point-virgule ‘;’). Les lignes correspondent aux lignes du fichier (finies par ‘\n’). Enfin, le texte à l’intérieur d’une cellule peut contenir le séparateur à condition que ce texte soit entouré de guillemets.

1. Dans la limite où cette redistribution ou modification doit respecter ces mêmes libertés. Un exemple de licence libre est l’*Open Database License* (ODbL).

Exemple

Le fichier CSV suivant :

N° notice;	ISBN;	EAN;	Titre
890137;	2-218-74082-6;	9782218740824;	L'assommoir
1092460;	;	5034504998872;	Radio Ga Ga
239359;	"2800126159; 2800126957"; ;		Gaston 15

Représente le tableau ci-dessous :

N° notice	ISBN	EAN	Titre
890137	2-218-74082-6	9782218740824	L'assommoir
1092460		5034504998872	Radio Ga Ga
239359	"2800126159;2800126957"		Gaston 15

Structure des données

Une collection est composée de **documents** identifiés de manière **unique** par un **numéro de notice**. Un document est caractérisé par son **type** (livre, CD audio, DVD, ...). De plus, différentes **métadonnées** peuvent lui être associées (titre, éditeur, date de publication, nom et prénom de l'auteur, ...). Un document peut aussi faire partie d'une **série**.

D'autres systèmes internationaux permettent de retrouver un document :

- L'ISBN (*International Standard Book Number*) identifiant une unique édition d'un **livre** (10 ou 13 caractères souvent séparés par des tirets)
- L'EAN (*European Article Numbering*) identifiant un unique produit de consommation (8 ou 10 caractères souvent associé à leur code barre)

Attention toutefois car plusieurs EAN et ISBN peuvent identifier un même document.



FIGURE 1 – Exemple d'ISBN et d'EAN sur un livre

Enfin, les dernières colonnes du tableau donnent pour chaque bibliothèque le **nombre d'exemplaires** en stock d'un document.

Modélisation

Vous devrez modéliser la collection dans des classes différentes représentant chaque type de document². De plus, vous devrez distinguer les documents qui font partie d'une série des autres types de documents (de manière à retrouver la série dont ils font partie).

Différents acteurs peuvent interagir avec cette collection. Les bibliothèques hébergent un ensemble de documents et peuvent s'en échanger entre elles. Les utilisateurs enregistrés dans une bibliothèque peuvent quant à eux emprunter et rendre des documents dans la limite du quota qui leur est imposé. Pour ce projet, on n'impose aucune limite de temps quant à la durée de l'emprunt. En revanche, pour emprunter un document, celui-ci doit être disponible dans la bibliothèque.

Une attention particulière devra être portée au modèle objet proposé (de manière à factoriser ce qui peut l'être), à l'encapsulation des données, à la gestion des erreurs, et à la documentation du code.

Fonctionnalités demandées

Votre application devra permettre de charger un potentiel jeu de données passé en paramètre et d'en construire un modèle initial :

```
java -jar biblio.jar [data.csv]
```

L'application doit également permettre interactivement d'**ajouter** :

1. une nouvelle bibliothèque dans le réseau;
2. un nouveau document dans le réseau, en prenant soin de vérifier que l'ISBN et EAN n'ont pas déjà été affectés à un ouvrage existant;
3. un nouvel utilisateur, associé à une bibliothèque, avec un quota maximum de documents qu'il peut emprunter.

L'application doit permettre de **consulter** (pour une bibliothèque ou l'ensemble du réseau) :

4. l'ensemble des documents;
5. pour une série, les documents triés par date de publication;
6. les documents d'un même auteur (par nom, prénom ou les deux);
7. la recherche d'un livre par son ISBN;
8. La recherche d'un document par son EAN;
9. Le nombre de documents de chaque type publiés dans un intervalle de temps.

L'application doit enfin permettre l'**emprunt** et la **remise** de documents entre les différents acteurs (utilisateurs et bibliothèques) dans la limite des stocks

2. Vous pouvez ignorer la sous catégorie pour limiter le nombre de classes dans votre modèle (ex. *Livre* pour *Livre jeunesse* ou *livre adulte*, *DVD* pour *DVD-vidéo tous publics*, ...).

et quota. Pour gérer les différents cas d’erreurs, vous créerez des exceptions pertinentes.

Rendu

Ce travail est à réaliser et à rendre **par groupe de 2 personnes**.

En premier lieu, vous aurez à réaliser un diagramme UML présentant les relations entre les classes de votre projet. Cet UML sera à rendre **avant le mardi 05 mai** (23h59 – UTC-12:00).

Pour le rendu final, vous aurez à réaliser en plus de votre programme, un diagramme UML présentant la structure définitive de votre projet, en justifiant les éventuelles différences avec le premier.

Chaque rendu devra être transmis par courrier électronique aux **deux** chargés de TP (Eugénie Brasier <eugenie.brasier@inria.fr> et Quentin Bouillaguet <quentin.bouillaguet@u-psud.fr>) avec comme titre d’email:
[Polytech > Et3] rendu projet MonPrénom MonNom

Le rendu final sera à envoyer **avant le mardi 19 mai** (23h59 – UTC-12:00) sous la forme d’une archive **.zip** contenant les fichiers et dossiers suivants:

- **src/** : dossier contenant les sources de votre programme
- **doc/** : dossier contenant la JavaDoc générée depuis vos sources
- **biblio.jar** : fichier jar **exécutable** de votre programme
- **uml/** : dossier contenant vos diagrammes UML