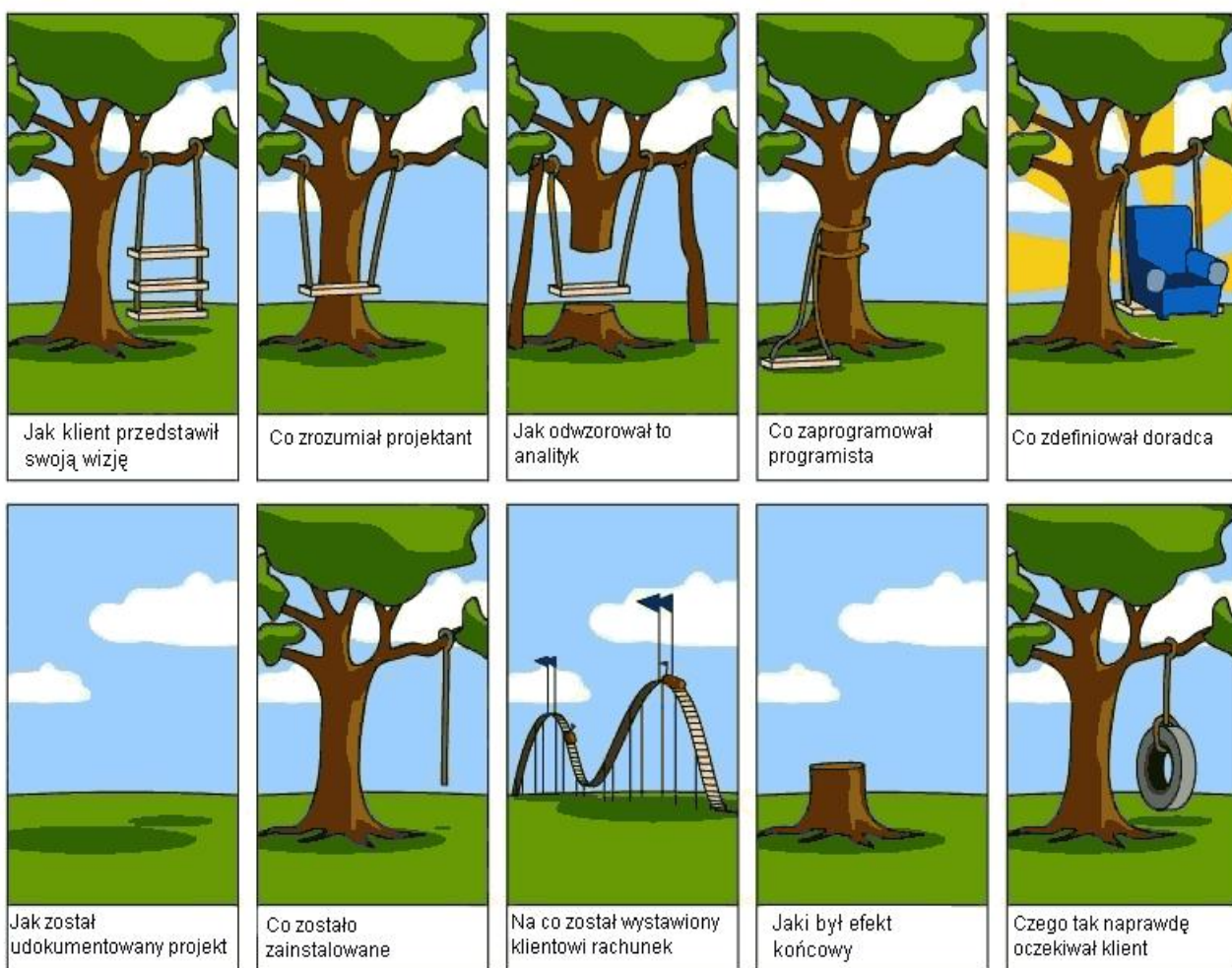


Stwórz projekt maven'owy w języku programowania java (nazwa projektu: LibrarySimulator), który będzie służył do obsługi biblioteki. Projekt ma zostać dodany do GitLab i proszę o dodanie do niego użytkownika g.szczutkowski z prawami Developer.

Projekt podzielony jest na dwa etapy. Etap pierwszy musi zostać ukończony do końca pierwszego semestru, etap drugi do końca drugiego semestru.

Nie jest wymagana obsługa programu z poziomu menu – projekt będzie obsługiwany poprzez wykonanie metod wymienionych w części 'Wymagania co do nazewnictwa klas i ich metod', stąd bardzo istotne jest, aby **nazewnictwo, modyfikatory dostępu oraz ilość i typ argumentów tych metod się zgadzały**.

Jeżeli wymaganie lub cokolwiek innego związanego z projektem jest niejasne, proszę pisać do mnie w celu wyjaśnienia – **nie domyślamy się jak coś ma działać** – wyjaśnianie wymagań jest bardzo istotne i normalne w branży IT więc pytajcie. Poniżej wyjaśnienie dlaczego powinniście pytać.



W dokumencie podane są przykłady wypisywania danych na ekranie lub eksportowania danych do pliku tekstowego. Proszę aby w projekcie **użyć tych samych separatorów co w dokumencie** np. **ID24[Traveler-03/2017; National Geographic-01/2016]** – nawiasy kwadratowe zawierają wszystkie magazyny przypisane do użytkownika, tytuł magazynu oddzielony jest myślnikiem od numeru, a poszczególne magazyny oddzielone są średnikiem.

Etap 1 - wymagania:

1. Możliwość dodawania nowych użytkowników biblioteki. Użytkownicy dzielą się na dwa typy:

- Student - może wypożyczyć maksymalnie 4 książki w jednym momencie

- Wykładowa - może wypożyczyć do 10 książek jednocześnie

Każdy użytkownik musi posiadać imię i nazwisko i numer karty w bibliotece. Może być dwóch użytkowników z tym samym imieniem i nazwiskiem ale numer karty w bibliotece musi być unikalny. Numer karty jest generowany automatycznie w momencie dodawania użytkownika do biblioteki – nie ma znaczenia jego format.

2. Możliwość wypisania wszystkich użytkowników biblioteki. Każdy wiersz dotyczy innego użytkownika i zawiera w następującej kolejności imię, nazwisko, numer karty w bibliotece oraz typ (oddzielone średnikami). Typ może mieć wartość S jeżeli użytkownikiem jest student lub L gdy użytkownikiem jest wykładowca

np.

```
Grzegorz;Szczutkowski;21;L
Piotr;Kowalczyk;13;S
Maria;Nowak;11;S
```

Etap 2 – wymagania:

1. Możliwość dodawania do zbioru w bibliotece dwóch rodzajów elementów:

- Książka – każda musi posiadać tytuł i autora
- Magazyn – każda musi posiadać tytuł i numer magazynu

2. Jeżeli dodawana jest istniejąca książka/magazyn to powinna się zwiększyć ilość wszystkich dostępnych egzemplarzy tej książki/magazynu.

3. Metoda dodawania Książek i Magazynów powinna mieć możliwość dodania jednego jak i kilku rodzajów elementów niezależnie od typu(książka, magazyn).

4. Możliwość wypożyczania Książek i Magazynów na rzecz użytkownika, jeśli w bibliotece znajduje się wystarczająca liczba egzemplarzy do wypożyczenia i użytkownik ma możliwość wypożyczenia kolejnego egzemplarza.

5. Możliwość wypisania na ekranie wszystkich Magazynów z biblioteki. Każdy wiersz dotyczy jednego numeru i zawiera w następującej kolejności tytuł, numer, ilość wszystkich egzemplarzy oraz ilość dostępnych egzemplarzy (oddzielone średnikami)

np.

```
Traveler;03/2017;30;17
National Geographic;01/2016;15;13
```

6. Możliwość wypisania na ekranie wszystkich Książek z biblioteki. Każdy wiersz dotyczy jednego tytułu i zawiera w następującej kolejności tytuł, autora, ilość wszystkich egzemplarzy oraz ilość dostępnych egzemplarzy (oddzielone średnikami)

np.

```
Ogniem i mieczem;H. Sienkiewicz;20;7
Tożsamość Bourne'a;R. Ludlum;40;33
```

7. Możliwość dodawania listy książek/magazynów z pliku tekstowego. W każdym wierszu w pliku powinien być tytuł, autor/numer, ilość egzemplarzy do dodania i typ (B – książka, M – magazyn) oddzielone średnikiem. Jeśli są już w bibliotece jakieś książki z pliku tekstowego to wtedy zwiększamy liczbę egzemplarzy dla tych pozycji. Poniżej przykładowa zawartość pliku z książkami:

```
Ogniem i mieczem;H. Sienkiewicz;20;B
Tożsamość Bourne'a;R. Ludlum;10;B
Gra o tron;George'a R.R. Martin;15;B
National Geographic;01/2016;15;M
```

8. Możliwość zapisania do nowego pliku numerów kart tych użytkowników biblioteki, którzy mają pożyczone egzemplarze i dla każdego użytkownika wypisane tytuły i autora książek lub tytuły i

numery magazynu, które posiada. Jeżeli ktoś posiada kilka takich samych egzemplarzy książki lub magazynu to powinien on być wyświetlony kilka razy
np.

ID5[Ogniem i mieczem-H. Sienkiewicz; Ogniem i mieczem-H. Sienkiewicz; Traveler-03/2017; National Geographic-01/2016]
ID24[Traveler-03/2017; National Geographic-01/2016]

Wymagania co do nazewnictwa klas i ich metod:

class Library	
public Library()	Konstruktor klasy obiektu typu Library
public void addUserToLibrary(User... users)	Metoda do dodawania Studentów oraz Nauczycieli do bazy biblioteki (punkt 1).
public void addItemToLibrary(Item.. item)	Metoda do dodawania Książek oraz Magazynów do bazy biblioteki (punkt 2, 3, 4).
public boolean rentItemToUser(Item item, User user)	Metoda do wypożyczania Książki lub Magazynu na rzecz Studenta lub Nauczyciela. Metoda zwraca 'true' jeżeli akcja wykonana zostanie poprawnie oraz 'false' gdy z jakiegoś powodu nie można wykonać akcji (punkt 5).
public void printListOfMagazines()	Metoda wypisująca na ekranie listę wszystkich magazynów z biblioteki z ilością wszystkich oraz dostępnych egzemplarzy(punkt 6).
public void printListOfBooks()	Metoda wypisująca na ekranie listę wszystkich książek z biblioteki z ilością wszystkich oraz dostępnych egzemplarzy(punkt 7).
public void printListOfUsers()	Metoda wypisująca na ekranie listę wszystkich użytkowników biblioteki z numerem karty i typem(punkt 8).
public void importItemsFromFile(String csvFile)	Metoda do importowania z pliku książek lub magazynów (punkt 9).
public void exportUsersWithItemsToFile(String csvFile)	Metoda do zapisywania w pliku użytkowników z wypożyczonymi egzemplarzami książek i magazynów (punkt 10).

class Magazine extends Item	
public Magazine(String number, String title)	Konstruktor tworzący obiekt typu Magazine (punkt 1).

class Book extends Item	
public Book(String author, String title)	Konstruktor tworzący obiekt typu Book (punkt 1).

class Student extends User	
public Student(String firstName, String lastName)	Konstruktor tworzący obiekt typu Student (punkt 1).

class Lecturer extends User	
public Lecturer(String firstName, String lastName)	Konstruktor tworzący obiekt typu Lecturer (punkt 1).