

Napisz program do zarządzania Biblioteką.

Uwaga: aplikacja może z pozoru wyglądać na trudną, ale gotowe implementacje niektórych metod były na ćwiczeniach.

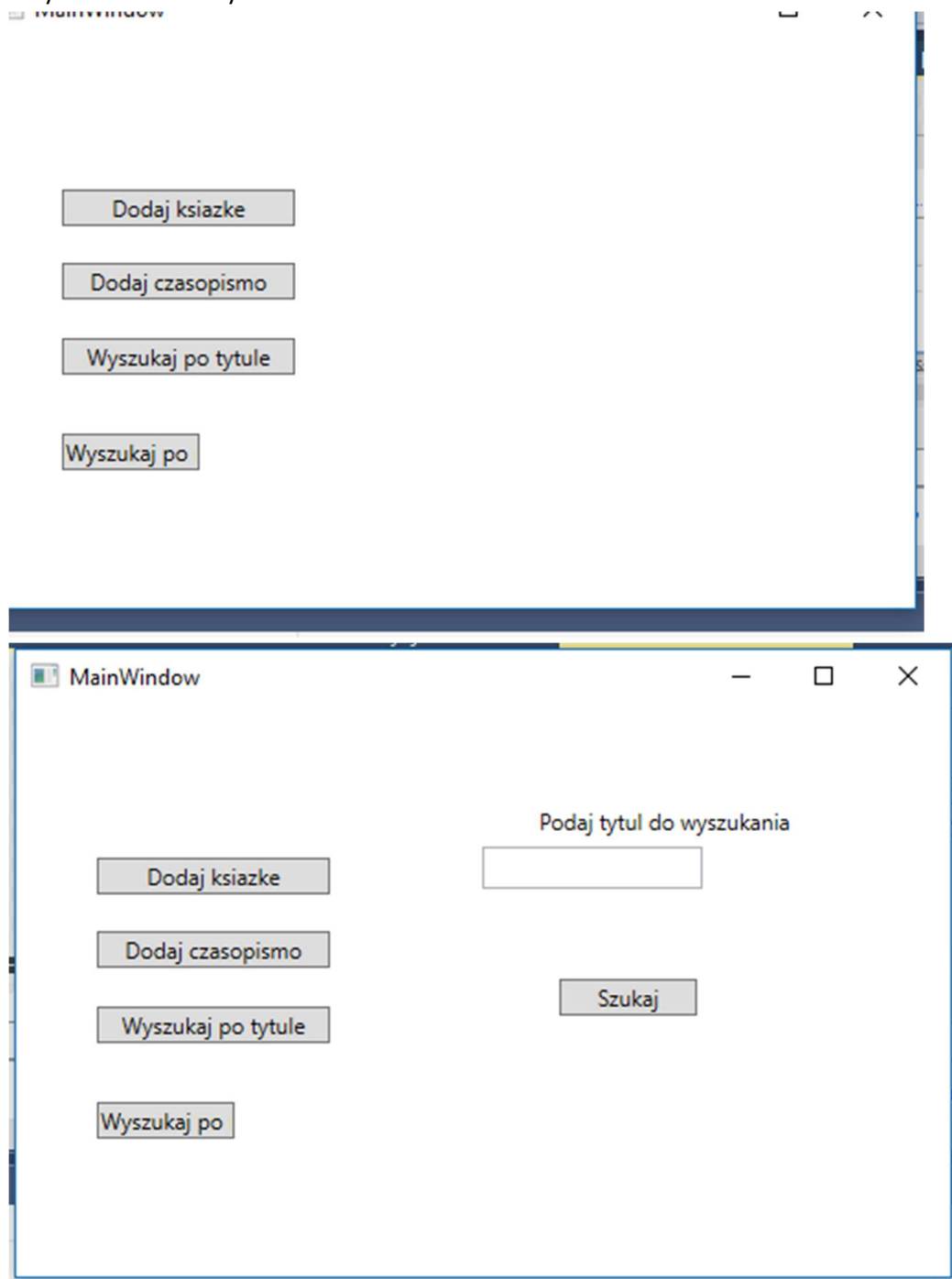
1. Stwórz klasę Autor z polami imie i nazwisko (mod. protected). Dodaj w klasie konstruktor domyślny i parametryczny (kolejność parametrów: imie, nazwisko).
2. Stwórz abstrakcyjną klasę Pozycja. Dodaj w niej pola tytuł (string), id (int), wydawnictwo (string), rokWydania (int) - wszystkie pola z mod. protected.
3. W klasie Pozycja dodaj konstruktor domyślny i parametryczny (kolejność parametrów tytuł, id, wydawnictwo, rokWydania).
4. Stwórz klasę Książka dziedziczącą z klasy Pozycja. Dodaj w niej prywatne pole liczbaStron (int). Następnie dodaj pole autor typu Autor (przechowujące obiekty z klasy Autor). Dodaj konstruktor domyślny i parametryczny (kolejność parametrów: imie(Autor), nazwisko(Autor), tytuł, id, wydawnictwo, rokWydania, liczbaStron). Następnie przesłoń metodę ToString(), która zwróci wszystkie informacje o książce (dowolnie czy w jednym wierszu, czy w kilku).
5. Stwórz klasę Czasopismo dziedziczącą z klasy Pozycja. Dodaj w niej prywatne pole numer (int). Stwórz konstruktor domyślny i parametryczny (kolejność parametrów: tytuł, id, wydawnictwo, rokWydania, numer). Przesłoń metodę ToString(), tak aby zwracała wszystkie informacje o czasopiśmie.
6. Stwórz klasę Katalog. Stwórz pole listaPozycji typu List<Pozycja> przechowujący pozycje z katalogu.
7. Stwórz w klasie Katalog konstruktor domyślny.
8. Następnie w klasie Katalog dodaj metodę DodajPozycje(pozycja), która będzie dodawać na listę książkę lub czasopismo. Następnie przeciąż metodę DodajPozycja następująco:
  - jeśli parametry są następujące (string imie, string nazwisko, string tytuł, int id, string wydawnictwo, int rokWydania, int liczbaStron), to dodawana do katalogu jest książka o odpowiednich parametrach.
  - jeśli parametry są następujące (string tytuł, string wydawnictwo, int rokWydania, int id, int numer), to dodawane jest czasopismo z odpowiednimi parametrami.
9. Stwórz Interfejs IZarządzanieKatalogiem. Dodaj w niej deklarację metod typu string: WyszukajPoTytule(string), WyszukajPoId(int),
10. Podepnij interfejs IZarządzanieKatalogiem do klasy Katalog. Dodaj implementację metod. Wyszukiwanie po tytule ma wyszukać wszystkie pozycje w katalogu o podanym tytule, następnie ich opis (poprzez metodę ToString() ma być zwrócony jako string) – uwzględnij, że może być więcej niż jedna pozycja o tym samym tytule. Wyszukiwanie po id zwraca pierwszą pozycje o podanym id.
11. W klasie Katalog dodaj publiczną metodę Test typu void. Implementacja ma być następująca:

```
DodajPozycje("Gazeta Olsztyńska", "Edytor", 1992, 7, 200);  
DodajPozycje("Gazeta Wyborcza", "Agora", 2010, 2, 123);  
DodajPozycje("Henryk", "Sienkiewicz", "Krzyżacy", 23, "Znak", 2010, 220);  
DodajPozycje("Henryk", "Sienkiewicz", "Krzyżacy", 24, "Znak", 2011, 220);
```
12. Stwórz aplikację WPF do obsługi katalogu. W klasie MainWindow stwórz prywatne pole

```
private Katalog biblioteka = new Katalog();
```

W konstruktorze MainWindow dla pola biblioteka wywołaj metodę Test().

Przykładowe screeny:



Punktacja:

Zrobienie punktów 1-3 – dwa punkty.

Zrobienie punktów 4-5 – dwa punkty.

Zrobienie punktów 6-7 – jeden punkt.

Zrobienie punktu 8 – jeden punkt.

Zrobienie punktów 9-11 – dwa punkty.

Zrobienie punktu 12 – dwa punkty.