

Звіт

по 2 лабораторній роботі
з програмування

Виконав студент групи К-24
Теодор Романус

Електронний архів факультету

Первинна постановка задачі:

Таблиці містять таку інформацію: П.І.П. автора, назва матеріалу, факультет (департамент, відділення), кафедра, вид матеріалу, обсяг, дата створення та ін.

Розробити інформаційну систему за допомогою технології ADO.NET з Entity Framework, яка б включала підсистему введення нової інформації та підсистему одержання довідок та пошуку інформації (окремо динаміка змін).

Увесь код лабораторної можна знайти [тут](#).

В лабораторній роботі наявний увесь функціонал 1 лаби:

1. Підтримка бази даних.
2. Візуалізація даних у вигляді однієї таблиці.
3. Можливість фільтрувати дані за факультетом, кафедрою, категорією. Пошук.
4. Додавання автора.
5. Додавання публікації.

Увесь функціонал портований з DataSet на Entities. Наприклад, запит на отримання, фільтрування і пошук даних з БД тепер виглядає так:

```

var arc = this.ctx.Archives.Future();
var pub = this.ctx.Publications.Future();
var cat = this.ctx.Categories.Future();
var fac = this.ctx.Faculties.Future();
var dep = this.ctx.Departments.Future();
var Query = (from a in arc
              join p in pub on a.PubID equals p.ID
              join c in cat on a.CategoryID equals c.ID
              join f in fac on a.FacultyID equals f.ID
              join d in dep on a.DepartmentID equals d.ID
select new
{
    Publication = p.Name,
    Authors = this.ctx.AuthorsOfPub(p.ID).First().Authors,
    Category = c.Name,
    Faculty = f.Name,
    Department = d.Name,
    Amount = a.Amount,
    PubDate = a.PubDate,
    ID = a.ID,
    PubID =a.PubID,
    FacultyID=a.FacultyID,
    CategoryID=a.CategoryID,
    DepartmentID=a.DepartmentID
});
if(category_id>0)
    Query = Query.Where(x => x.CategoryID==category_id);
if (faculty_id > 0)
    Query = Query.Where(x => x.FacultyID == faculty_id);
if (department_id > 0)
    Query = Query.Where(x => x.DepartmentID == department_id);

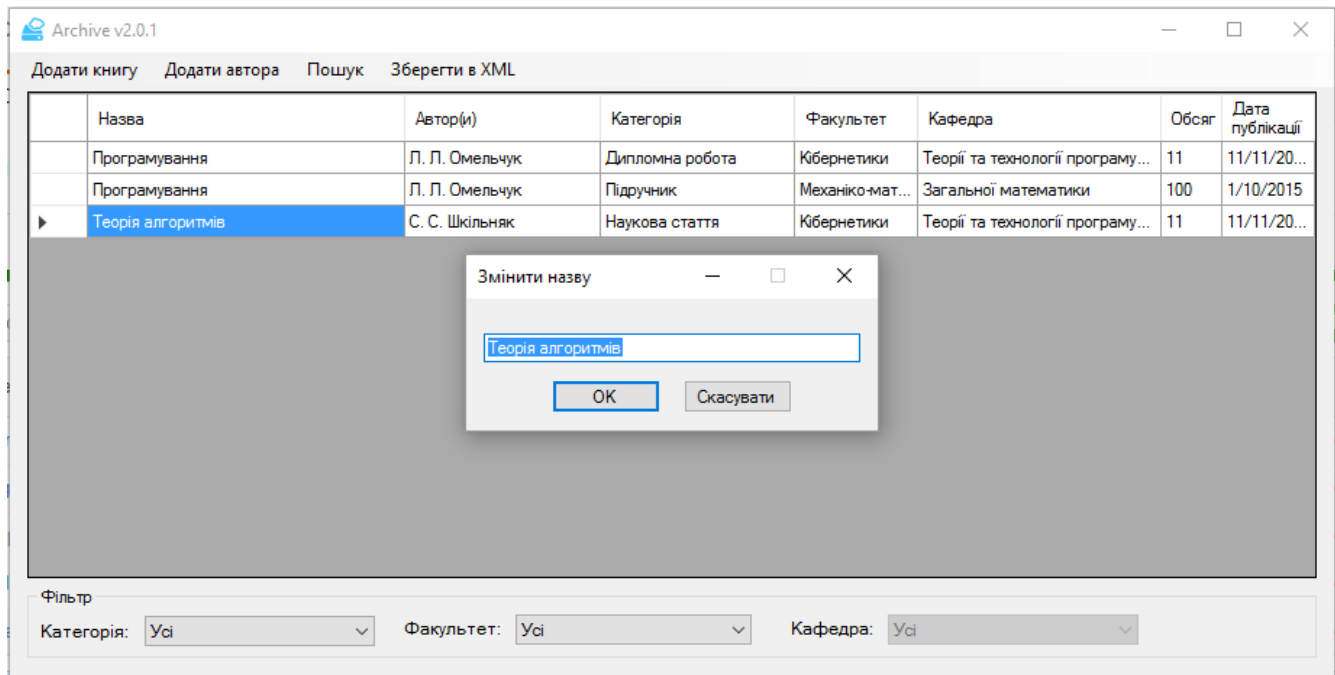
this.bindingSource.DataSource = Query.OrderBy(x=>x.Publication).ToList();

```

Додатковий функціонал:

6. Редагування назви книги.

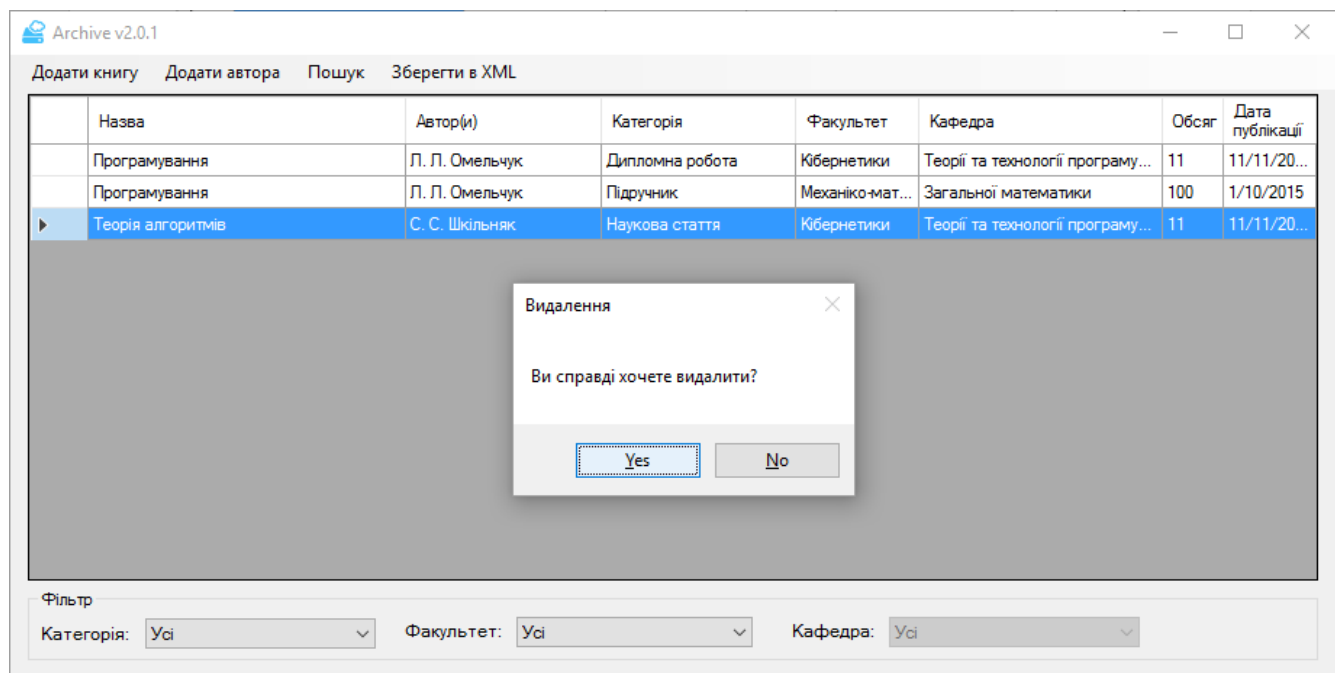
На вимогу викладача додано можливість змінювати назву книги. Відкриття вікна редагування здійснюється подвійним натисканням на поле назви:



Після введення нової назви і перевірки, чи книга з новою назвою відсутня в БД, відбувається Update-запит за допомогою LINQ.

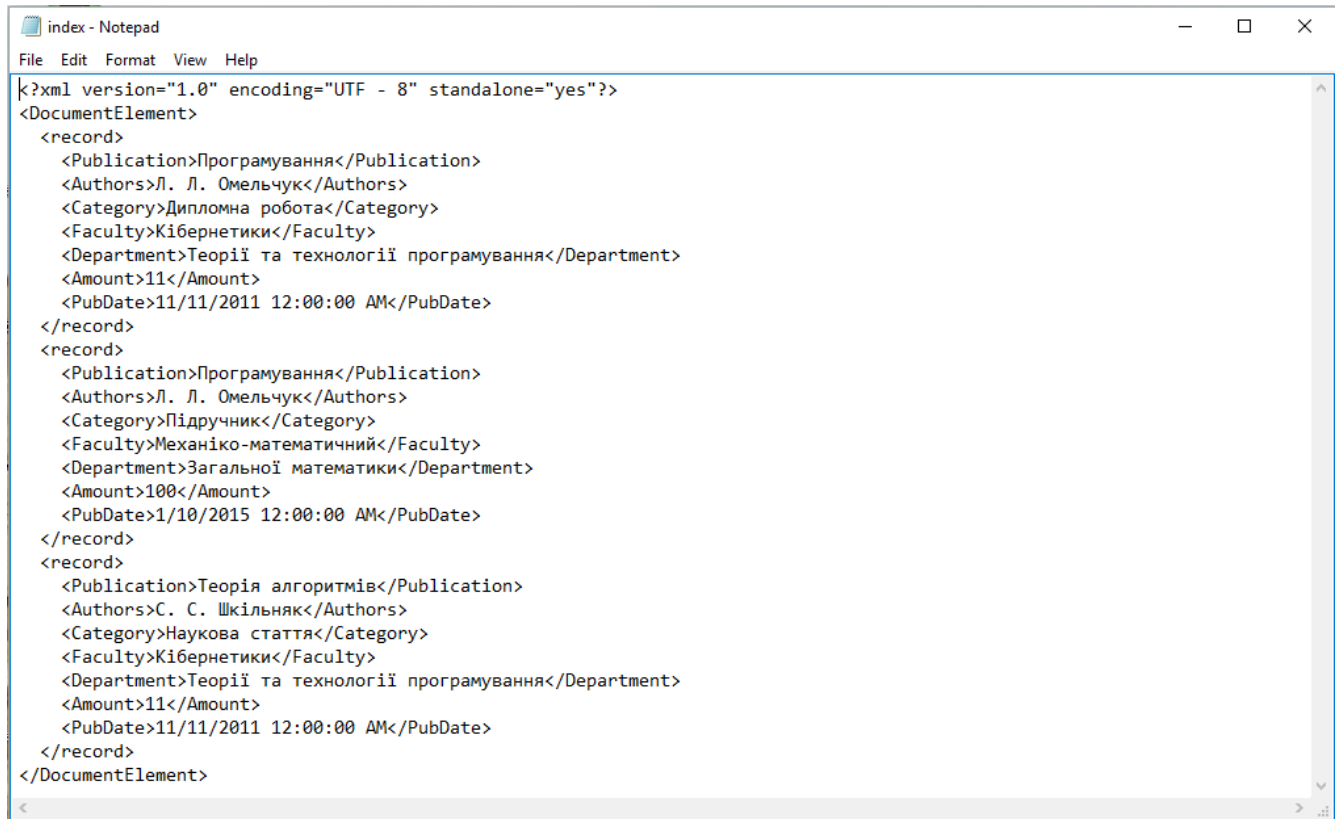
7. Видалення книги.

Після виділення публікації, її також можна видалити шляхом натискання на клавішу Delete:



8. Збереження результатів в XML.

Для перетворення даних, відображених в таблиці, використовувався універсальний спосіб. Відкриті поля, які на момент натискання клавіші «Зберегти в XML» знаходились в DataGridView (тобто без прихованих ID), копіювались в DataTable, після цього використовувався метод DataTable.WriteXML():



```
index - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="UTF - 8" standalone="yes"?>
<DocumentElement>
  <record>
    <Publication>Програмування</Publication>
    <Authors>Л. Л. Омельчук</Authors>
    <Category>Дипломна робота</Category>
    <Faculty>Кибернетики</Faculty>
    <Department>Теорії та технології програмування</Department>
    <Amount>11</Amount>
    <PubDate>11/11/2011 12:00:00 AM</PubDate>
  </record>
  <record>
    <Publication>Програмування</Publication>
    <Authors>Л. Л. Омельчук</Authors>
    <Category>Підручник</Category>
    <Faculty>Механіко-математичний</Faculty>
    <Department>Загальної математики</Department>
    <Amount>100</Amount>
    <PubDate>1/10/2015 12:00:00 AM</PubDate>
  </record>
  <record>
    <Publication>Теорія алгоритмів</Publication>
    <Authors>С. С. Шкільняк</Authors>
    <Category>Наукова стаття</Category>
    <Faculty>Кибернетики</Faculty>
    <Department>Теорії та технології програмування</Department>
    <Amount>11</Amount>
    <PubDate>11/11/2011 12:00:00 AM</PubDate>
  </record>
</DocumentElement>
```