**УВАГА! ДОКУМЕНТ НАДАНО СТУДЕНТОМ 2 КУРСУ ПОТОКУ ІНФОРМАТИКИ 2014-2015 Н.Р.**

**Лабораторна робота № 3**

Є інформаційна система (вважаємо, що вона вже існує і на даному етапі її розробляти непотрібно!). Інформація з інформаційної системи передається між клієнтами та системою у вигляді XML-документів. Необхідно забезпечити обробку цих документів. Обробка включає в себе дві задачі: аналіз вмісту документу (пошук інформації за ключовими словами та динамічну генерацію запитів) та трансформацію у файл HTML. Вхідні дані для аналізу та трансформації надаються у вигляді файлу-прикладу \*.xml. Трансформація документу в HTML-код виконується на основі XSL-документа \*.xsl. Аналіз вмісту документа повинен бути виконаний трьома способами – за допомогою SAX API, DOM API та LINQ to XML. Необхідно реалізувати всі три способи!

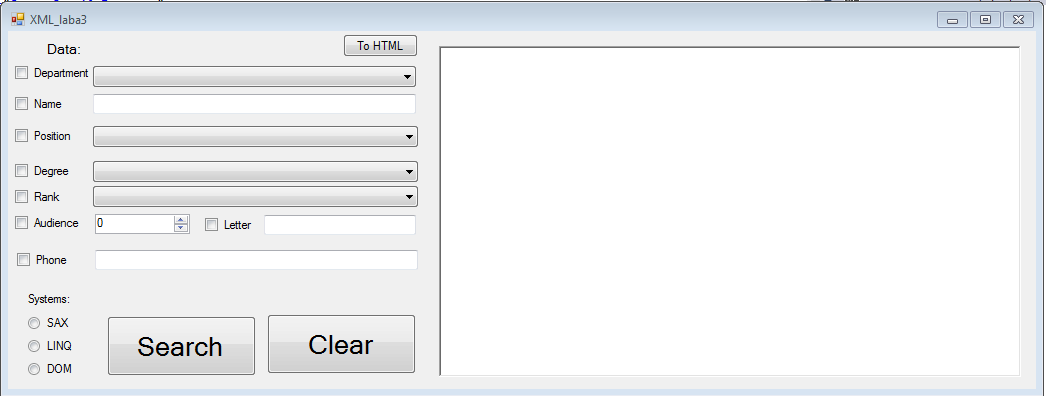
1. Створення XML документу

Прописуємо власноруч текст файлу в редакторі (Наприклад в Блокноті або Notepad++). Зберігаємо в типі \*.xml. Потрібно враховувати, що повинен існувати ОДИН головний кореневий об’єкт, який містить всі потрібні нам. В даному випадку база даних про всі кафедри містить кафедри, а кафедри містять викладачів. Атрибути вузлів створюєте на власний смак.



Враховано, що одна людина може мати багато звань. Тому створюються дочірні вузли-звання. Враховуйте такі можливості для своєї роботи.

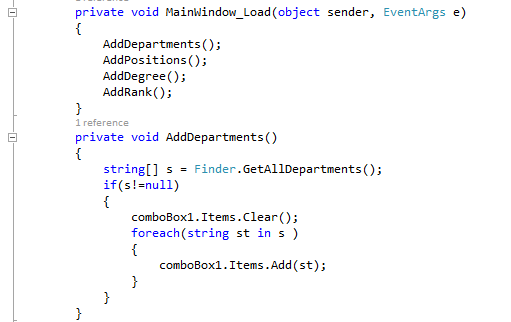
1. Створення форми



Потрібно враховувати можливість пошуку за кожним критерієм окремо і комбіновано. Тому доцільно використовувати CheckBox.

Оскільки пошук буде вестися лише одною з систем, то доцільно використовувати RadioButton.

В форму не повинно бути вшито даних. Краще на початку виконання програми прочитати всі можливі варіанти параметрів і задати в ComboBox. Так ви будете знати, з чого вибирати, без ризику зробити помилку.



Потрібно прибрати можливість вводити самому дані, а дозволити лише вибирати.

ComboBox -> Properties->DropDownStyle->DropDownList



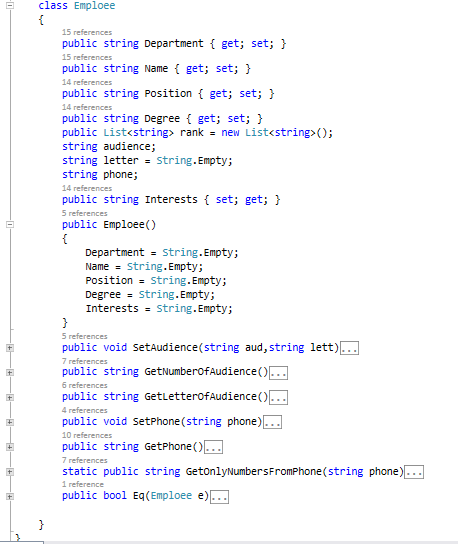
Форма не повинна працювати з файлами чи робити операції над даними. Для цього слід виділити окремий клас (Finder в даному випадку) і також клас, який буде слугувати контейнером для обміну інформацією між формою та моделлю(Emploee (трошки не дописав - буває) в даному випадку). За таких умов формі все рівно, звідки беруться дані, її завдання:

* Передати параметри пошуку
* Отримати дані та відобразити їх

Тоді запит форми буде виглядати приблизно так:



«Контейнер» виглядає так:



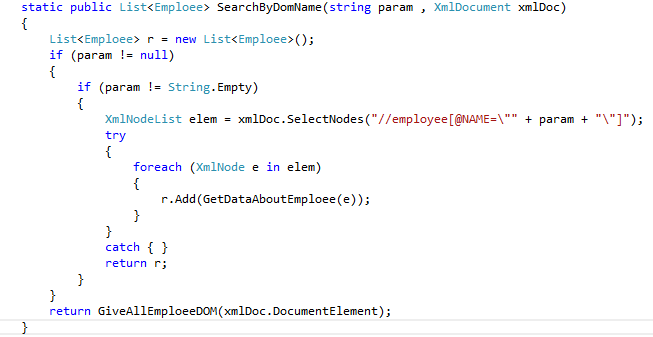
1. Пошук за допомогою DOM

В пошук за допомогою цієї системи я поклав принцип: шукаю за кожним параметром окремо, а потім беру перетин результатів. Такий підхід допустимий лише при не дуже великих об’ємах інформації.

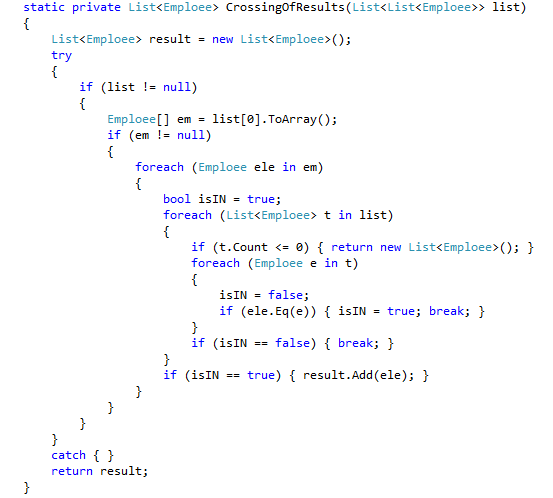
Так виглядає головна функція пошуку:



Всі фільтри абсолютно ідентичні. Ось приклад одного з них:



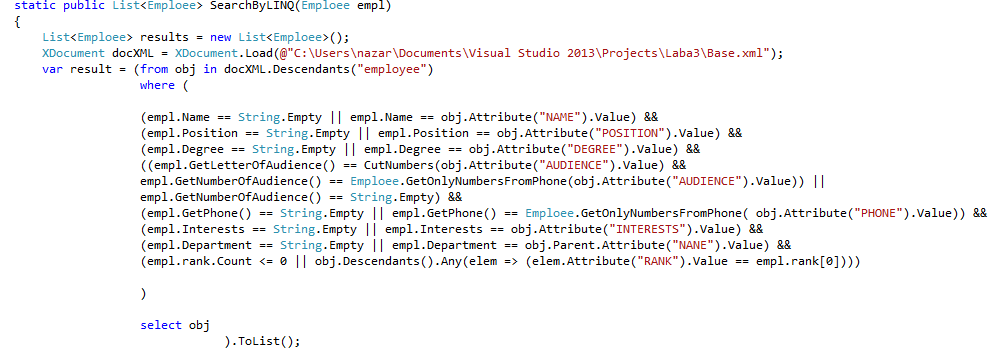
Тепер потрібно перетнути результати:



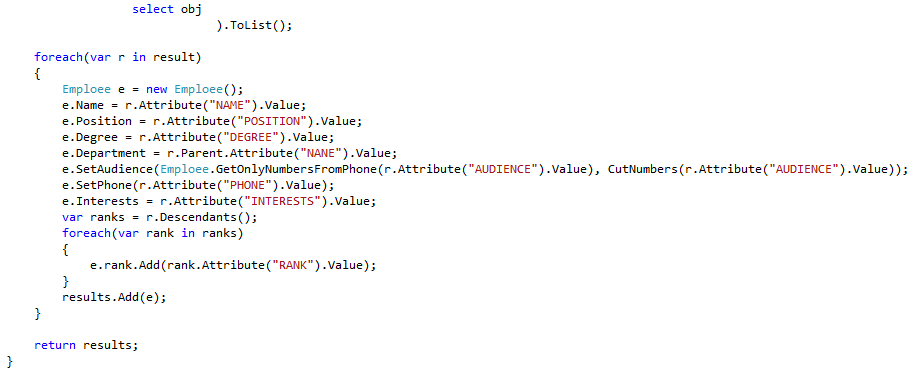
Результат передається формі.

1. Пошук за допомогою LINQ

В даному випадку не важко написати запит, який фільтрує по всім параметрам одночасно.



Але потрібно пам’ятати, що пошук не відбувається одразу. Тому існування контейнеру грає нам на руку. Переписуємо результати в прийнятному для форми вигляді:



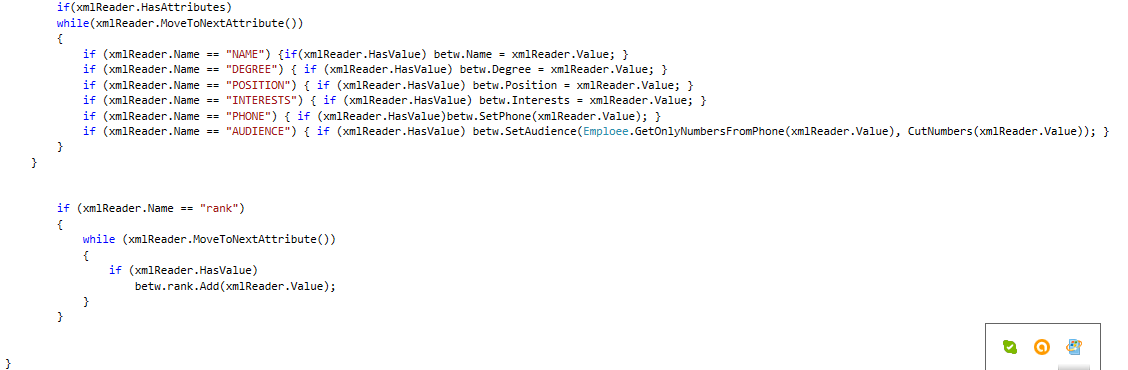
Пошук реалізовано єдиним методом.

1. Пошук за допомогою SAX

В даному випадку я вирішив дістати з xml-документу всіх робочих, а потім відфільтрувати за параметром.

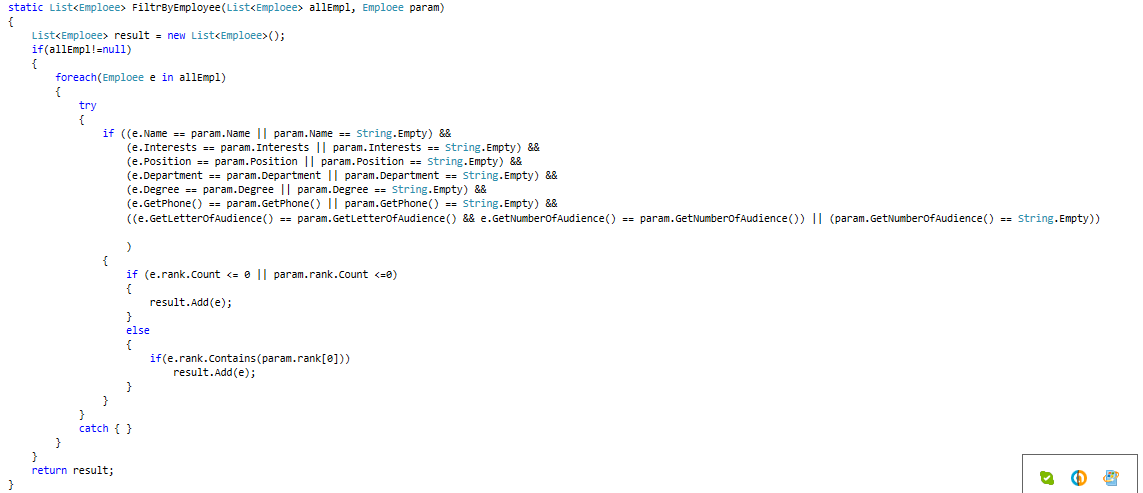
Зчитування має наступний вигляд:





Тепер потрібно відфільтрувати і повернути результати.

Краще створити окрему функцію-фільтр:



Тоді залишається записати в попередню функцію:



Форма отримує результат.

1. Трансформація

Найголовніше на цьому етапі правильно написати файл трансформації, враховуючи структуру свого xml.

Приклад xsl:



Логіка трансформування. Для кожного працівника створюється таблиця з даними, вона міститься в загальній таблиці кафедри. Для кожної кафедри створюється така таблиця.

Звернути увагу на:

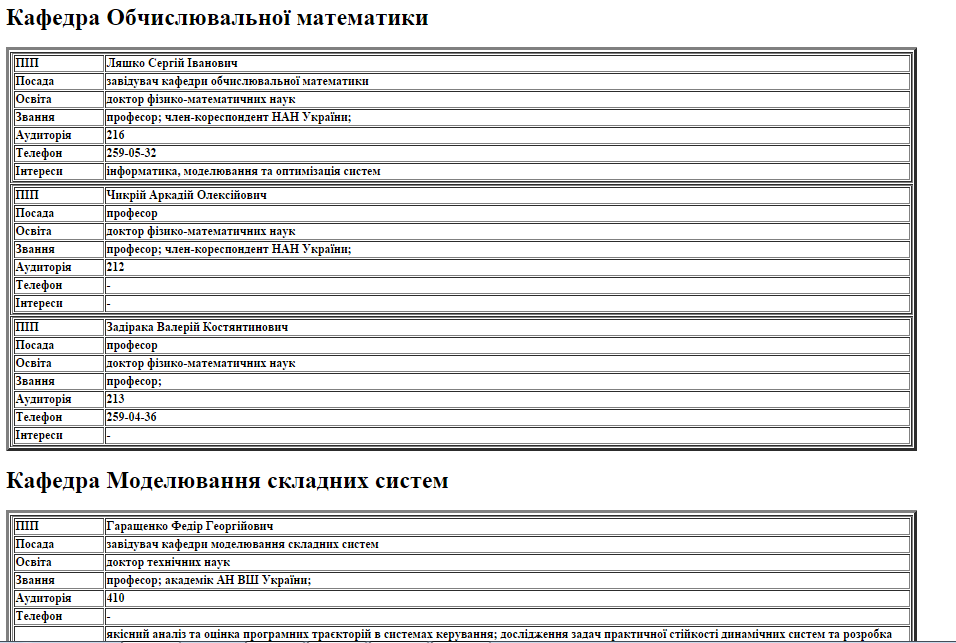
- співвідношення між розмірами комірок

(без співвідношення розміри комірок залежать від кількості даних, а для різних працівників вона, як правило, різна – вийде «крива» таблиця)

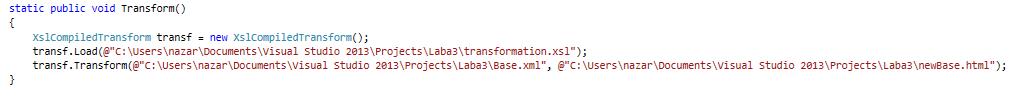
* вирівнювання

(без нього дані будуть в центрі комірки – теж не гарно).

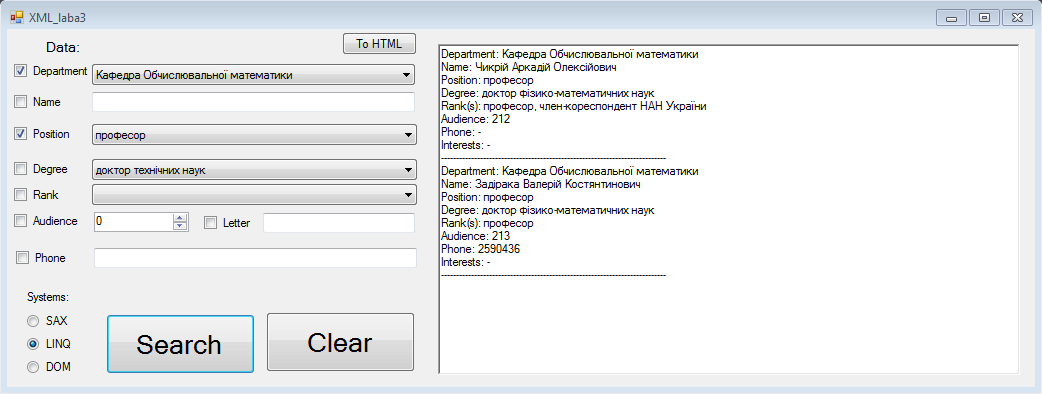
Результат:



Функція конвертації оформлюється так:



1. Вигляд програми в робочому стані:



**Увага:** Як бачимо, «Освіта» не врахована, оскільки ми не поставили «галочку»