# Приложение «Личная база данных фильмов»

## База данных

Содержит таблицы Movie, Country, Person.

### Входные данные

Входной файл movies.csv. Кодировка файла windows-1251. Первую строку следует пропустить так она содержит имена колонок. Следующие строки содержат информацию об отдельном фильме. Отдельные части разделяются точкой с запятой.

Описание полей в порядке их появление:

1. Название фильма
   * Всегда в наличии
2. Год выпуска фильма или дата выпуска в формате «день.месяц.год»
   * Может отсутствовать
   * Для фильмов только с годом предполагать, что имеется в виду 1 января данного года.
3. Страна
   * Может отсутствовать
4. Режиссер фильма
   * Может отсутствовать
5. Сценарист.
   * Может отсутствовать
   * Обратите внимание, что поля сценарист и режиссер могут содержать одно и тое же имя. В базе данных это должна быть одна запись!
6. Рейтинг фильма на сайте imdb.com. Число с дробной частью. Разделитель запятая.
   * Может отсутствовать
7. Ссылка на описание фильма на сайте imdb.com
   * Может отсутствовать

## Слой доступа к базе данных

Отдельный проект с использованием Entity Framework.

## Программа импорта данных

Консольное приложение. Читает данные из файла описанного в разделе «Входные данные» и вносит их в базу данных. Кодировка входного файла windows-1251 (см. класс StreamReader, Encoding.GetEncoding()).

## WCF сервис

Предоставляет три операции:

1. Добавить фильм – void Add (Movie movie)
   1. Может возбудить исключение если произошла ошибка работы с БД. Код метода не должен обрабатывать данные исключения, а позволить им уйти клиенту.
2. Удалить фильм – void Delete (string title)
   1. Если фильм с таким именем не существует, то метод «молча» завершает исполнение
   2. Если найдено несколько фильмов с таким названием, то удаляются все
3. Получить список фильмов – List<Movie> GetList(string title)
   1. Если аргумент title == null или содержит пустую строку, то возвращаем первые N фильмов отсортированных по названию. (String.IsNullOrEmpty).
   2. Если аргумент title != null , то возвращаем фильмы в имени которых содержится заданная строка. (LINQ & Enumerable.Where, String.Contains)
   3. Если ничего не найдено, то метод должен вернуть пустую коллекцию.
   4. В любом случаях метод не должен возвращать больше чем 125 фильмов. Данное значение должно быть объявлено как константа. (LINQ & Enumerable.Take (count))

Сервис должен работать через http протокол на порту 8899.

## Хост для WCF сервиса

Реализуется в виде консольного приложения. Конфигурацию сервиса описываем в файле конфигурации приложения (app.config). За основу можно взять пример из главы №3 книги Pro WCF. Practical Microsoft SOA Implementation.

Так как операция создания http хоста требует прав администратора, то добавьте в данный проект файл манифеста с такой строкой: <requestedExecutionLevel level="requireAdministrator" uiAccess="false" />. Это гарантирует корректную работу приложения под Windows Vista и выше.

## Графический интерфейс

Программа использующая WPF или Windows Forms. Должна получать данные от WCF сервиса и отобразить их на экране.

## Структура решения

* Проект №1 – Доступ к БД.
* Проект №2 – Программа импорта данных в БД. Использует проект №1 для добавления данных в БД
* Проект №3 – WCF сервис. Использует проект №1 для реализации своей фукнциональности.
* Проект №4 – Хост для WCF сервиса.
* Проект №5 – Программа на WPF/WinForms. Содержит Service Reference на сервис из проекта №3.

## Для крутых перцев и перчинок ☺

Пункты перечисленные здесь являются необязательными. Выполняйте их только если вы полностью сделали основную часть задания.

* Убедитесь, что ваша программа нечуствительна к SQL Injection атакам.
* Добавьте возможность перейти по ссылке указанной в информации о фильме. При этом должен открыться браузер по умолчанию.
* Добавьте возможность иметь несколько режиссеров и сценаристов для одного фильма
* Добавьте постер(ы) фильма

# Советы

Если вы работаете под Windows Vista или Windows 7, то настройте запуск Visual Studio с правами администратора.