Урок номер 8.

**Классная работа.**

**1.**

Создайте класс **MyClass**. В данном классе создайте 4 приватных поля – 2 строковых и 2 целочисленных. В теле метода **Main** создайте экземпляр класса **MyClass** и проинициализируйте все поля данного класса. Выполните сериализацию экземпляра **MyClass** в XML файл и десериализацию из XML файла в новый экземпляр **MyClass**. Выведите на экран значение полей нового экземпляра **MyClass.**

При создании **MyClass** примите во внимание, что сериализоваться должны только значения одного строкового поля и одного целочисленного.

**2.**

Выполните задачу 1 с сериализацией в формате **Binary**.

**3.**

Создайте структуру **Triangle** реализующую интерфейс **IDeserializationCallback**. В теле структуры создайте:

* три public поля – обозначающие стороны треугольника;
* private поле, обозначающие периметр треугольника (значение поля должно заполняться во время инициализации полей);
* private метод, выполняющий вычисление периметра.

Реализуйте логику работы интерфейса **IDeserializationCallback** с тем, чтобы при десериализации экземпляров **Triangle** происходило вычисление значения поя периметра.

Сериализуйте экземпляр и десериализуйте.

**4.**

**Домашняя работа.**

**1.**

Создайте консольную программу-дневник, соответствующую следующим требованиям:

1. При запуске консоли пользователя просят выбрать – зарегистрирован он уже или хочет зарегистрироваться.
2. Если пользователь хочет зарегистрироваться – его просят ввести желаемый логин и пароль. Логин проверяется на уникальность - наличие в системе такого же. После регистрации пользователя снова спрашивают, как в п.1.
3. Если пользователь хочет авторизоваться и уже зарегистрирован в системе – пользователя просят ввести логин и пароль для входа.
4. После авторизации пользователю выводят текущую дату и просят ввести сегодняшнюю запись личного электронного дневника построчно. Для выхода пользователь вводит “quit”.
5. После ввода текста сегодняшней записи информация по каждому пользователю должна сохраняться так, чтобы при следующем запуске вы могли получить доступ ко всем предыдущим записям.
6. Получить доступ ко всем записям можно введя в п.4 вместо “quit” слово “data”.
7. Система должна иметь возможность поддерживать работу с несколькими пользователями (не одновременно), сохраняя их записи.

**Подумайте над возможностью разделения логики таким образом, чтобы внешний интерфейс взаимодействия впоследствии можно было бы подменить на Win Forms, WPF или Web-интерфейс, а логику сохранения в файл – подменить на логику сохранения в базу данных.**

**Проект залейте на Github.**