

№ 13 Сериализация

Задание

<https://docs.microsoft.com/en-us/dotnet/standard/serialization/>

1. Из лабораторной №4 выберите класс с наследованием и/или композицией/агрегацией для сериализации. Выполните сериализацию/десериализацию объекта используя форматы:
 - a. *Binary*,
 - b. *SOAP*,
 - c. *JSON*,
 - d. *XML*.

Запретите сериализацию одного из членов вашего класса и продемонстрируйте отсутствие данного элемента в результате работы сериализаторов

Все сериализаторы должен реализовывать общий интерфейс. Выбор и использование сериализатора следует реализовать таким образом, чтобы добавление нового сериализатора не требовало изменения существующего кода

* *Усложненное задание:*

*Создайте класс **CustomSerializer**, который обеспечивает сериализацию и десериализацию любых объектов любых типов всеми вышеперечисленными способами. Интерфейс класса и параметры методов продумайте самостоятельно.*

2. Создайте коллекцию (массив) объектов и выполните сериализацию/десериализацию – возможность сохранения и загрузки списка объектов в/из файла.

* *Усложненное задание:*

Создайте клиент и сервер на синхронных сокетах.

Нужно сериализованные данные(объект) отправить по сокету и десериализовать на стороне клиента.

3. Используя *XPath* напишите два селектора для вашего *XML* документа.
4. Используя *Linq to XML* (или *Linq to JSON*) создайте новый *xml (json)* - документ и напишите несколько запросов.

Вопросы

1. Что такое сериализация, десериализация?
2. Какие существуют форматы сериализации? Поясните структуру для каждого формата. Какие классы для работы с ними существуют в .NET?
3. Какие классы существуют в пространстве имен *System.Xml*?
4. Какие атрибуты используются для настройки XML сериализации?

5. В чем отличие BinaryFormatter или SoapFormatter?
6. Что такое сериализация контрактов данных, контракт данных?
7. Где и для чего используются атрибуты [OnSerializing], [OnSerialized], [OnDeserializing], [OnDeserialized]?
8. Что такое XPath? Приведите пример.
9. Какие возможности дает LINQ to Xml. Приведите примеры