

Форматы XML и JSON. Файлы конфигурации



#### После урока обязательно



Повторите этот урок в видео формате на <a href="ITVDN.com">ITVDN.com</a>



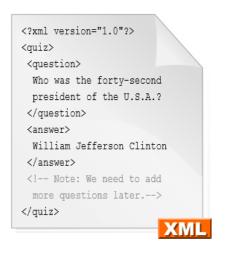
Проверьте как Вы усвоили данный материал на TestProvider.com



Форматы XML и JSON. Файлы конфигурации



#### eXtensible Markup Language



Текстовый формат, предназначенный для хранения структурированных данных для обмена информацией между программами, а также для создания на его основе более специализированных языков разметки.

Стандарты XML <a href="http://www.w3.org/standards/xml/">http://www.w3.org/standards/xml/</a>



#### Структура XML документа

#### Документ XML:

- объявление XML первая строка документа, в которой указывается версия, кодировка текста и другие зависимости
- корневой элемент является основным телом документа, у каждого документа может быть только один корневой элемент
- теги вложенные элементы документа, могут иметь атрибуты и содержимое
- комментарии текст, размещенный в теге <!-- -->



- - System.XML [4.0.0.0]

  - {} System.Xml
    - ▶ <sup>4</sup> NameTable
    - → 

      M XmlAttribute
    - > 3 XmlAttributeCollection
    - XmlCDataSection
    - XmlCharacterData
    - > 🛠 XmlComment

    - > M XmlDocument
    - > 🤻 XmlDocumentFragment
    - > **%** XmlDocumentType
    - > 🥸 XmlElement
    - XmlEntity
    - XmlEntityReference
    - > 🥸 XmlException
    - > **XmlImplementation**
    - XmlLinkedNode
    - → 

      <sup>†</sup> XmlNamedNodeMap
    - > **%** XmlNamespaceManager
    - > 🏂 XmlNameTable

    - XmlNodeChangedEventArgs
    - XmlNodeList
    - XmlNodeReader
    - XmlNotation
    - > **%** XmlParserContext
    - XmlProcessingInstruction
    - XmlQualifiedName
  - > March March
  - → 

    <sup>†</sup>

    XmlResolver
  - XmlSecureResolver
  - XmlSignificantWhitespace
  - → 

    <sup>\*</sup>

    XmlText
  - → 

    M XmlTextReader

    M XmlTextReade
  - XmlTextWriter

  - XmlValidatingReader
  - XmlWhitespace
  - XmlWriter
  - ▶ ⋘ XmlWriterSettings

#### System.Xml

Пространство имен, в котором

содержатся классы, необходимые для

работы с XML файлами.

- System.XML [4.0.0.0]

  - ↓ { } System.Xml.Serialization.Configuration

  - System.Xml.Xsl



#### System.Xml.XPath

- {} System.Xml.XPath
  - > **XPathDocument**
  - > **XPathException**
  - > **XPathExpression**
  - > 🔧 XPathItem
  - → ★ XPathNavigator

  - IXPathNavigable

  - > 

    ## XmlDataType
  - > 

    ## XmlSortOrder
  - ▶ XPathNamespaceScope
  - XPathNodeType
  - > J XPathResultType

Пространство имен, позволяющее использовать возможности языка XPath (XML Path Language).

Содержит класс **XPathDocument**, позволяющий создавать экземпляры **XPathNavigator**.

**XPathNavigator** — предоставляет курсор-ориентированную модель для просмотра и редактирования XML данных.



#### JSON

JSON основан на двух структурах данных:

- Коллекция пар ключ/значение объект.
- Упорядоченный список значений массив

Объект начинается с открывающей фигурной скобки и заканчивается закрывающей фигурной скобкой. Каждое имя сопровождается двоеточием, пары ключ/значение разделяются запятой.

Массив начинается с открывающей квадратной скобки и заканчивается закрывающей квадратной скобкой. Значения разделены запятой.

Значение может быть строкой в двойных кавычках, числом, **true**, **false**, **null**, объектом или массивом. Эти структуры могут быть вложенными.

Более подробно: https://www.json.org/json-ru.html



#### **JSON**

```
"Name": "Alexander",
"Age": 27,
"Work Experience": [
    "Company": "Microsoft",
    "Position": "Software Developer",
    "Start": "2015-12-10T00:00:00",
    "End": "2018-09-13T00:00:00"
  },
    "Company": "MySoftwareCompany",
    "Position": "Software Developer",
    "Start": "2018-10-08T00:00:00",
    "End": "2020-11-08T00:00:00"
```



#### System.Text.Json

- System.Text.Json
  - ▲ () System.Text.Json

    - ▶ ♣ JsonDocument

    - D 🔩 JsonException
    - ▶ ★ JsonNamingPolicy

    - D 🔩 JsonSerializer
    - ▶ ★ JsonSerializerOptions

    - ▶ JsonValueKind

    - Utf8JsonWriter

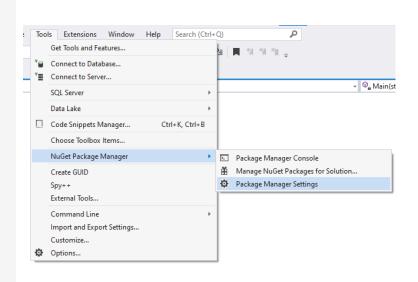
Пространство имен для работы с документами в формате JSON.

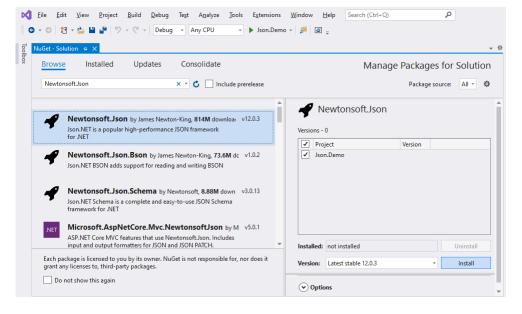


#### Newtonsoft. Json



Для начала работы необходимо установить Nuget пакет под названием Newtonsoft. Json.







■ ■ Newtonsoft.Json

▶ () Microsoft.CodeAnalysis▲ () Newtonsoft.Json

▷ ■ ConstructorHandling
 ▷ ■ DateFormatHandling
 ▷ ■ DateParseHandling
 ▷ ■ DateTimeZoneHandling
 ▷ ■ DefaultJsonNameTable

▶ ■ DefaultValueHandling▶ ■ FloatFormatHandling

▶ ■ FloatParseHandling
 ▶ ■ Formatting
 • □ IArrayPool<T>
 • □ IJsonLineInfo
 ▶ ↑ JsonArrayAttribute
 ▶ ↑ JsonConstructorAttribute

▶ ★ JsonConverterAttribute

▶ ♣ JsonConverterCollection

JsonlgnoreAttribute

JsonObjectAttribute

JsonPropertyAttribute

JsonNameTable

JsonReader

▶ ⁴s JsonTextReader
 ▶ ⁴s JsonTextWriter
 ▶ ♂ JsonToken

JsonExtensionDataAttribute

▶ ★ JsonConvert

JsonConverter

JsonException

▷ ★ JsonConverter<T>

#### Microsoft. Extensions. Configuration

- Microsoft.Extensions.Configuration

  - ▲ () Microsoft.Extensions.Configuration
    - ▶ ★ ChainedBuilderExtensions
    - ChainedConfigurationProvider
    - 🕨 🔩 ChainedConfigurationSource
    - ConfigurationBuilder
    - 🕨 🔩 ConfigurationKeyComparer
    - ConfigurationProvider
    - D 🔩 ConfigurationReloadToken
    - ConfigurationRoot
    - ConfigurationSection
    - ▶ ★ MemoryConfigurationBuilderExtensions
    - StreamConfigurationProvider
    - ▶ ★ StreamConfigurationSource

Пространство имен для работы с файлами конфигурации.

Для начала работы необходимо установить Nuget пакет под названием Microsoft.Extensions.Configuration.



Q&A



#### Информационный видеосервис для разработчиков программного обеспечения















