

Folder Person

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Homeworke0811

{

class Person

{

public string Name { get; set; }

public int Age { get; set; }

public string Specialization { get; set; }

public string City { get; set; }

public string Mood { get; set; }

public int HaveNiceDay { get; set; }

public Person(string name, int age, string specialization)

{

Name = name;

Age = age;

Specialization = specialization;

}

}

}

Folder xml

<?xml version="1.0" encoding="utf-8"?>

<people>

<person name="Миша">

<specialization>Coder</specialization>

<age>18</age>

<city>Вологда(Vologda)</city>

<mood>Bad</mood>

<haveNiceDay>0</haveNiceDay>

</person>

<person name="Тюлень">

<specialization>Coder</specialization>

<age>18</age>

<city>Vologda</city>

<mood>Great</mood>

<haveNiceDay>1</haveNiceDay>

</person>

</people>

Folder task

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Xml;

namespace Homeworke0811

{

class task

{

static void Main(string[] args)

{

XmlDocument xDoc = new XmlDocument();

xDoc.Load("people.xml");

// получим корневой элемент

XmlElement? xRoot = xDoc.DocumentElement;

task a = new task();

XmlElement personElem = xDoc.CreateElement("person");

XmlAttribute nameAttr = xDoc.CreateAttribute("name");

XmlElement specializationElem = xDoc.CreateElement("specialization");

XmlElement ageElem = xDoc.CreateElement("age");

XmlElement cityElem = xDoc.CreateElement("city");

XmlElement moodElem = xDoc.CreateElement("mood");

XmlElement havenicedayElem = xDoc.CreateElement("haveniceday");

XmlText nameText = xDoc.CreateTextNode("Максим");

XmlText specText = xDoc.CreateTextNode("Database Admin");

XmlText ageText = xDoc.CreateTextNode("18");

XmlText cityText = xDoc.CreateTextNode("Vologda");

XmlText moodText = xDoc.CreateTextNode("Greate");

XmlText havenicedayText = xDoc.CreateTextNode("0");

nameAttr.AppendChild(nameText);

specializationElem.AppendChild(specText);

ageElem.AppendChild(ageText);

cityElem.AppendChild(cityText);

moodElem.AppendChild(moodText);

havenicedayElem.AppendChild(havenicedayText);

personElem.Attributes.Append(nameAttr);

personElem.AppendChild(specializationElem);

personElem.AppendChild(ageElem);

personElem.AppendChild(cityElem);

personElem.AppendChild(moodElem);

personElem.AppendChild(havenicedayElem);

xRoot?.AppendChild(personElem);

xDoc.Save("people.xml");

if (xRoot != null)

{

// обход всех узлов в корневом элементе

foreach (XmlElement xnode in xRoot)

{

// получаем атрибут name

XmlNode? attr = xnode.Attributes.GetNamedItem("name");

Console.WriteLine(attr?.Value);

// обходим все дочерние узлы элемента user

foreach (XmlNode childnode in xnode.ChildNodes)

{

// если узел - company

if (childnode.Name == "age")

{

Console.WriteLine($"Age: {childnode.InnerText}");

}

// если узел age

if (childnode.Name == "specialization")

{

Console.WriteLine($"Specialization: {childnode.InnerText}");

}

if (childnode.Name == "city")

{

Console.WriteLine($"City: {childnode.InnerText}");

}

if (childnode.Name == "mood")

{

Console.WriteLine($"Mood: {childnode.InnerText}");

}

if (childnode.Name == "haveniceday")

{

if (childnode.InnerText == "1")

{

Console.WriteLine("Have Nice Day:)");

}

else

{

Console.WriteLine("Haven't Nice Day:(");

}

}

}

}

}

}

}

}