

Jazyk C# 2

4. seminář

Jakub Večeřa

Univerzita Palackého v Olomouci

3. 3. 2025

Reakce na úkoly

- Ošetření výjimek – nepřípravené disky, zápis, čtení
- Efektivita zápisu – kolik bajtů je potřeba pro zápis dané matice
 - ▶ Velikost bool?
 - ▶ Velikost int?
 - ▶ Bitové posuny

Čtení XML (1/2)

- Známe formát XML?
- Vstup:

```
1 <?xml version="1.0" ?>
2 <items>
3   <item att1="foo" att2="bar">
4     <number>25</number>
5     <str>Some string</str>
6   </item>
7   <item att1="bar" att2="baz">
8     <number>5</number>
9     <str>Another string</str>
10  </item>
11  <item att1="xgf" att2="abc">
12    <number>2255</number>
13    <str>The End!</str>
14  </item>
15 </items>
```

Čtení XML (2/2)

- Práce s DOM (Document Object Model)
- „Stromový přístup“

Čtení XML (2/2)

- Práce s DOM (Document Object Model)
- „Stromový přístup“

```
1 string filePath = @"E:\5\in.xml";
2 XmlDocument doc = new XmlDocument();
3 doc.Load(filePath); // Nacteni dokumentu
4 XmlNodeList nodeList = doc.GetElementsByTagName("item"); // Vybrani elementu "item"
5 foreach (XmlNode node in nodeList) {
6     Console.WriteLine(node.Name);
7     Console.WriteLine("Data: ");
8     foreach (XmlNode child in node.ChildNodes) { // Potomci elementu
9         Console.Write(child.Name);
10        Console.WriteLine(child.Value);
11    }
12    Console.WriteLine("Atributy: ");
13    foreach (XmlAttribute att in node.Attributes) { // Atributy elementu
14        Console.Write(att.Name + ": ");
15        Console.WriteLine(att.Value);
16    }
17 }
```

Zápis XML

- Podobný přístup - budování stromu

```
1 XmlDocument doc = new XmlDocument();
2 XmlDeclaration xmlDeclaration = doc.CreateXmlDeclaration("1.0", "UTF-8", null);
3 XmlElement root = doc.DocumentElement;
4 doc.InsertBefore(xmlDeclaration, root);
5
6 XmlElement item = doc.CreateElement("item");
7 item.SetAttribute("att1", "some value");
8 item.SetAttribute("att2", "another value");
9
10 XmlElement number = doc.CreateElement("number");
11 number.InnerText = 8876.ToString();
12 item.AppendChild(number);
13
14 XmlElement str = doc.CreateElement("str");
15 str.InnerText = "Yet another string";
16 item.AppendChild(str);
17
18 doc.AppendChild(item);
19 XmlTextWriter xmlTextWriter = new XmlTextWriter(@"E:\5\out.xml", Encoding.UTF8);
20 xmlTextWriter.Formatting = Formatting.Indented;
21 doc.WriteContentTo(xmlTextWriter);
22 xmlTextWriter.Close();
```

Serialize

- Ruční zápis a načítání je zdlouhavé
- Vytvoříme si novou třídu modelující `item` z XML

```
1 public class Item
2 {
3     public int Number { get; set; }
4     public string Str { get; set; }
5     public int[] Numbers { get; set; }
6 }
```

- Serializace:

```
1 Item i = new Item() { Number = 42, Str = "abcdefg", Numbers =
    new int[]{ 5, 4, 3, 2, 1 } };
2 TextWriter tw = new StreamWriter(@"E:\5\serialize.xml");
3 XmlSerializer sr = new XmlSerializer(typeof(Item));
4 sr.Serialize(tw, i);
5 tw.Close();
```

Deserializace

- Opak serializace, tedy načtení xml

```
1 FileStream f = new FileStream(@"E:\5\serialize.xml",  
    FileMode.Open);  
2 XmlSerializer serializer = new XmlSerializer(typeof(Item));  
3 Item item = (Item)serializer.Deserialize(f);  
4 Console.WriteLine(item);  
5 f.Close();
```

- Jednoduché a bez práce

Formát JSON

- Známe formát JSON?
- Skvěle vysvětleno na: <https://www.json.org/json-cz.html>

Formát JSON

- Známe formát JSON?
- Skvěle vysvětleno na: <https://www.json.org/json-cz.html>

```
1 { "studentPredmetu":  
2   [  
3     { "jmeno": "Vojtech", "prijmeni": "BARNET", "titulPred": null, "nazevSp": "Informacni  
         technologie"},  
4     { "jmeno": "David", "prijmeni": "CINKL", "titulPred": null, "nazevSp": "Informatika"},  
5     { "jmeno": "Tobias", "prijmeni": "CVRCEK", "titulPred": null, "nazevSp": "Informacni  
         technologie"},  
6     { "jmeno": "Tomas", "prijmeni": "DAVIES", "titulPred": null, "nazevSp": "Informatika"},  
7   ]  
8 }
```

Serializace do JSON

- Namespace `System.Text.Json`

- Serializace:

```
1 string SerData = JsonSerializer.Serialize(data);
```

- Výstup:

```
1 { "Number":42, "Str":"abcdefg", "Numbers":[5,4,3,2,1] }
```

Deserializace z JSON

- Úplně stejný přístup:

```
1 string SerData =  
    "{\"Number\":42,\"Str\":\"abcdefg\",\"Numbers\":[5,4,3,2,1]}\";";  
2 Item data = JsonSerializer.Deserialize<Item>(SerData);;  
3 Console.WriteLine(deserialized);
```

Úkol (1/2)

- (Ručně – pomocí XMLDocument) načíst XML se seznamem studentů předmětu a (ručně – vybudovat řetězec)zapsat do JSON ty, kteří jsou ve 2. ročníku
 - ▶ <https://www.inf.upol.cz/downloads/jcs2/studentiPredmetu.xml>
- Načíst JSON s daty počty nakažených
<https://www.inf.upol.cz/downloads/jcs2/nakaza.json>
 - ▶ Zdroj
<https://onemocneni-aktualne.mzcr.cz/api/v2/covid-19/nakaza.json>
(již nedostupné)
 - ▶ Vypočítat 7denní klouzavý průměr z přírůstků
 - ▶ Vypočítané zapsat do XML v obdobné formě
- Deserializovat XML s předměty katedry a vyserializovat do JSON prvních 20 předmětů
 - ▶ <https://www.inf.upol.cz/downloads/jcs2/predmetyKatedry.xml>
- Hint:
 - ▶ *Edit* → *Paste special* je váš pomocník

Úkol (2/2)

- Náznak volání

```
1 readStudentsWriteSecondYear("C:\studentiPredmetu.xml",  
    "C:\druhaci.json");  
2 readCovidWriteAverages("C:\nakaza.json", "C:\nakazaAverages.xml");  
3 deserializeSubjectsWriteFirstN("C:\predmetyKatedry.xml",  
    "C:\firstNSubjects.xml", 20);  
4 Console.WriteLine("DONE!!!!");  
5 Console.ReadLine();
```