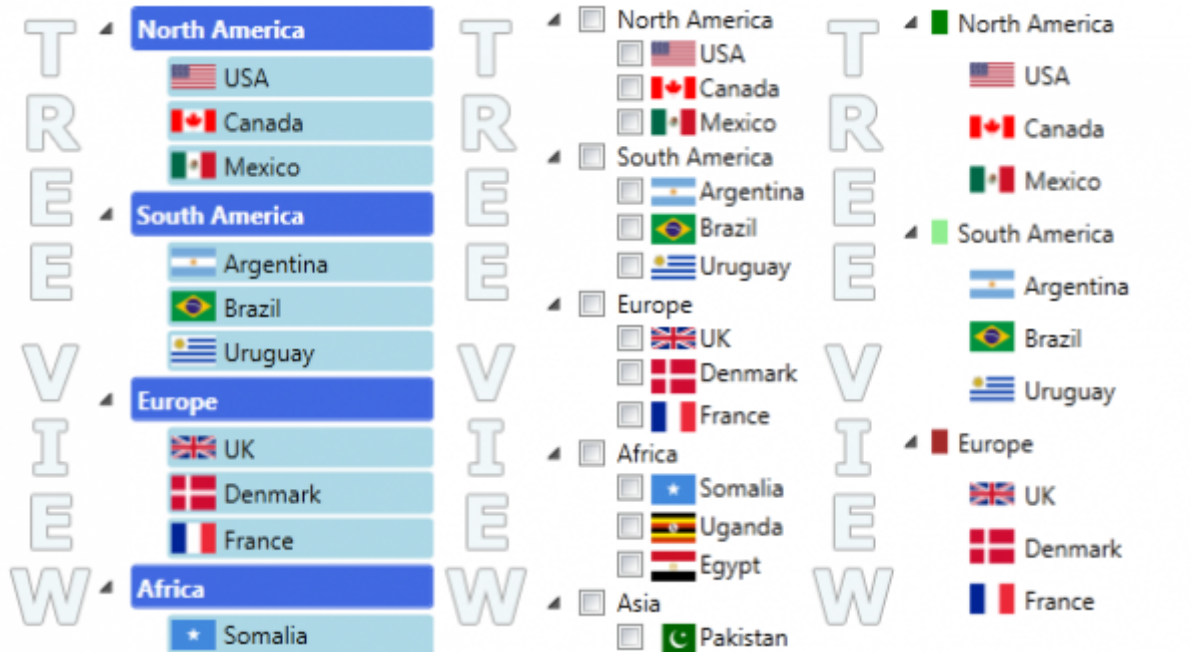
Inhaltsverzeichnis

TreeView 1
Einfacher Datenbaum erstellen 1
Angepasster Datenbaum erstellen 2
xxx Datenbaum erstellen 3
DataTable and DataSet 4
From IList<T> to DataTable 4

TreeView



Dieser Artikel beschreibt die Verwendung des TreeView-Steuerelements, das von WPF bereitgestellt wird. Auf diesem Weg will ich Ihnen aufzeigen, wie man einen einfachen Tree, eine Vorlage und eine Bindung dazu erstellt. Dieser Bericht konzentriert sich hauptsächlich darauf, wie man Inhalte in einem Treeview anzeigt. Diese Vorstellung ist eine Abschrift aus dem folgenden **Code Project** Bericht. Basic Understanding of Tree View in WPF

Einfacher Datenbaum erstellen

Um einen einfachen Datenbaum zu erstellen benötigen man lediglich eine Treeview Komponente und etwas XAML Code.

```
1. #XAML
2. <TreeView>
3.     <TreeViewItem Header="North America">
```

```
        <TreeViewItem Header="USA"></TreeViewItem>
        <TreeViewItem Header="Canada"></TreeViewItem>
        <TreeViewItem Header="Mexico"></TreeViewItem>
    </TreeViewItem>
    <TreeViewItem Header="South America">
        <TreeViewItem Header="Argentina"></TreeViewItem>
        <TreeViewItem Header="Brazil"></TreeViewItem>
        <TreeViewItem Header="Uruguay"></TreeViewItem>
    </TreeViewItem>
</TreeView>
```

Dieser Datenbaum lässt sich auch mit C# Code erstellen. Dazu muss ein Treeview mit einem Namen erstellt werden. An dieses benannte Treeview Objekt kann dann benutzt werden um weiter Treeviewitem hinzu zu fügen.

```
#XAML
<TreeView Name="tvMain">
</TreeView>
```

```
#C#
TreeViewItem treeItem = null;

// North America
treeItem = new TreeViewItem();
treeItem.Header = "North America";

treeItem.Items.Add(new TreeViewItem() { Header = "USA" });
treeItem.Items.Add(new TreeViewItem() { Header = "Canada" });
treeItem.Items.Add(new TreeViewItem() { Header = "Mexico" });

tvMain.Items.Add(treeItem);
```

Angepasster Datenbaum erstellen

Wenn Sie einige andere Steuerelemente mit dem Inhalt z.B. Checkbox, Bild, etc. hinzufügen möchten, dann können Sie Ihren Baum ohne großen Aufwand gestalten. Sie müssen nur das HeaderTemplate des TreeViewItem anpassen. Sie können auch eine Klasse erstellen, die von TreeViewItem abgeleitet ist, und den Header entsprechend Ihren Anforderungen ändern.

```
#XAML
<TreeView>

</TreeView>
```

---

xxx Datenbaum erstellen

```
#XAML
<TreeView>

</TreeView>
```

DataTable and DataSet

Das folgende Beispiel soll ein List Objekt in eine Datentabelle umwandeln. Ein generische Lösungsansatz ermöglicht es alle Eigenschaften und nur deren Eigenschaften, in eine Datentabelle umzuwandeln. Da die Eingenschaften auch NULL sein können muss dies bei der Konvertierung berücksichtigt werden.

The following example is to convert a List object into a data table. A generic approach allows to convert all properties and only their properties into a data table. Since the properties can also be NULL, this must be taken into account during the conversion.

Example A Example B

From IList<T> to DataTable

```
1. public DataTable ConvertToDataTable<T>(IList<T> data)
2.     {
3.         PropertyDescriptorCollection properties =
4.            TypeDescriptor.GetProperties(typeof(T));
5.         DataTable table = new DataTable();
6.         foreach (PropertyDescriptor prop in properties)
7.             table.Columns.Add(prop.Name,
   Nullable.GetUnderlyingType(prop.PropertyType) ?? prop.PropertyType);
8.         foreach (T item in data)
9.         {
10.            DataRow row = table.NewRow();
11.            foreach (PropertyDescriptor prop in properties)
12.                row[prop.Name] = prop.GetValue(item) ?? DBNull.Value;
13.            table.Rows.Add(row);
14.        }
15.        return table;
16.
17.    }
```

From:
https://jmz-elektronik.ch/dokuwiki/ - **Bücher & Dokumente**

Permanent link:
https://jmz-elektronik.ch/dokuwiki/doku.php?id=start:visualstudio2019:programmieren:dotnet:component:programmierbeispiele

Last update: **2022/12/19 16:32**