ATTRIBUTE

Vincent Uhlmann
IT-Akademie Dr. Heuer GmbH

ATTRIBUTE

- Attribute ermöglichen es uns, Metadaten zu unserem Code hinzuzufügen
- Sie können z.B. für Assemblys, Typen, Methoden oder auch Eigenschaften verwendet werden

- Diese Metadaten sind in verschiedenen Anwendungsfällen hilfreich
- Zum Beispiel um dem Compiler Anweisungen zu geben
- Um durch Reflection an bestimmte Informationen zu gelangen
- …

ATTRIBUTE

- Um ein Attribut anzuwenden, muss es in eckigen Klammern über der Klasse, Eigenschaft, Methode... stehen

```
[JsonPropertyName("name")]
public string Name { get; set; } = string.Empty;
```

- Es ist möglich mehr als ein Attribut hinzuzufügen

```
[JsonPropertyName("alter"), Column]
public string Alter { get; set; } = string.Empty;

// oder

[JsonPropertyName("alter")]
[Column]
public string Alter { get; set; } = string.Empty;
```

REFLECTION UND ATTRIBUTE

- Mithilfe der Klasse Attribute kann zur Laufzeit abgefragt werden, ob ein bestimmter Typ ein Attribut hat

```
Attribute.IsDefined(typeof(Person), typeof(FooAttribute));
```

- Mittels der Attribute Klasse oder der Methoden der Type Klasse können die einzelnen Attribute abgefragt werden

```
Type type = typeof(Person);

IEnumerable<FooAttribute> attributes = type.GetCustomAttributes<FooAttribute>();

foreach(FooAttribute attribute in attributes)
{
    Console.WriteLine(attribute.Id);
}
```

ATTRIBUTE

- Um ein eigenes Attribut zu erstellen, muss von der Attribut Klasse geerbt werden
- Der Name sollte auf Attribute enden
- Bei der Verwendung wird der Name jedoch ohne Attribute benutzt
- Über Konstruktoren und Eigenschaften kann das Attribut wie jede andere Klasse gestaltet werden

```
public class FooAttribute : Attribute
{
    public int Id { get; }

    public FooAttribute(int id)
    {
        Id = id;
    }
}
```

```
public class Person
{
    [Foo(5)]
    public int Age { get; set; }
}
```

ATTRIBUTE USAGE

- Das Attribut „AttributeUsage“ wird verwendet, um einzuschränken, wo ein Attribut verwendet werden kann
- Hiermit können Attribute definiert werden, die z. B. nur für Eigenschaften und Felder gültig sind

```
[AttributeUsage(AttributeTargets.Property | AttributeTargets.Field)]
public class FooAttribute : Attribute
{
    public int Id { get; }

    public FooAttribute(int id)
    {
        Id = id;
    }
}
```

MEHRFACHVERWENDUNG VON ATTRIBUTEN

- AttributeUsage bietet einen weiteren Parameter namens AllowMultiple
- Dieser Parameter kann verwendet werden, um eine Mehrfachverwendung desselben Attributs zu ermöglichen

```
[AttributeUsage(AttributeTargets.Property, AllowMultiple = true)]
public class FooAttribute : Attribute
{
    public int Id { get; }

    public FooAttribute(int id)
    {
        Id = id;
    }
}
```

```
public class Person
{
    [Foo(5)]
    [Foo(6)]
    [Foo(7)]
    public int Age { get; set; }
}
```