

Codekonventionen für C#

Allgemein:

- Englischer Code.
- Kommentare Deutsch und/oder Englisch (egal).
- Ausgaben in der Landessprache (hier: Deutsch).
- Code der länger als 120 Zeichen ist "umbrechen" (in der nächsten Zeile weiterschreiben).
- Jede Datei bekommt einen Headerkommentar, in dem Sinn und Zweck der Datei beschrieben wird.
- In der History wird aufgeführt wer, was, wann geändert hat.
- Beispiel:

```
/******  
* Project: CMN5201GPR-0322-Game  
* File   : MyScript.cs  
* Date    : 09.04.2022  
* Author  : Ralf Hüwe (RH)  
*  
* These coded instructions, statements, and computer programs contain  
* proprietary information of the author and are protected by Federal  
* copyright law. They may not be disclosed to third parties or copied  
* or duplicated in any form, in whole or in part, without the prior  
* written consent of the author.  
*  
* History:  
* 09.04.22    RH    Created  
******/
```

Verwendet Pascal Casing für Klassennamen und Methodennamen.

```
public class ClientActivity  
{  
    public void ClearStatistics()  
    {  
        //...  
    }  
    public void CalculateStatistics()  
    {  
        //...  
    }  
}
```

Verwendet Pascal Casing für structs.

```
public struct MySettings  
{  
    //...  
}
```

Verwendet camel Casing für Methodenargumente und lokale Variablen.

```
public class UserLog
{
    public void Add(LogEvent logEvent)
    {
        int itemCount = logEvent.Items.Count;
        // ...
    }
}
```

Verwendet Screaming Caps für Konstanten.

```
public const string SHIPPINGTYPE = "DropShip";
```

Verwendet Pascal Casing für public Variablen und Property's, vermeidet unterstriche in den Bezeichnungen.

```
public DateTime ClientAppointment;
public TimeSpan TimeLeft;
// Property
public int MyProperty { get; set; }

private float _health = 100f;
public float Health
{
    get => _health;
    set => _health = value;
}

// Avoid
public DateTime client_Appointment;
public TimeSpan time_Left;
```

Verwendet camel Casing für private Variablen und stellt das Präfix "_" voran.

```
private string _myString = "Hello";
```

Verwendet Pascal Casing für Events und stellt das Präfix "On" voran.

```
public event Action OnPlayerMove;
```

Verwendet camel Casing für statische Felder und stellt das Präfix "s_" voran.

```
private static float s_moveSpeed = 5.0;
```

Verwendet Pascal Casing für enums.

```
public enum Color
{
    Red,
    Green,
    Blue,
    Yellow,
    Magenta,
    Cyan
}
```

Verwendet Pascal Casing für Interfaces und stellt das Präfix "I" voran.

```
public interface IShape
{
}
public interface IShapeCollection
{
}
public interface IGroupable
{
}
```